



Katedra informatiky, FPV, Univerzita Mateja Bela  
MPC, Regionálne pracovisko v Banskej Bystrici  
UIPŠ, Školské výpočtové stredisko Banská Bystrica

# **DidInfo 2015**

21. ROČNÍK NÁRODNEJ KONFERENCIE

BANSKÁ BYSTRICA  
2015

**Predseda programového výboru/ General Chair:**

doc. Ing. Ľudovít Trajtel', PhD., Univerzita Mateja Bela, Banská Bystrica, SK

**Programový výbor konferencie/Programm Committee:**

doc. RNDr. Gabriela Andrejková, CSc., Univerzita Pavla Jozefa Šafárika, SK  
PaedDr. Ivan Brodenec, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
Mgr. Martin Cápay, PhD., Univerzita Konštantína Filozofa, Nitra, SK  
doc. RNDr. Miroslava Černochovej, CSc., Univerzita Karlova, Praha, CZ  
Ing. Dana Horváthová, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
Ing. Jana Jacková, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
prof. RNDr. Ivan Kalaš, PhD., Univerzita Komenského, Bratislava, SK  
doc. RNDr. Zuzana Kubincová, PhD., Univerzita Komenského, Bratislava, SK  
doc. RNDr. Gabriela Lovászová, PhD., Univerzita Konštantína Filozofa, Nitra, SK  
Ing. Jana Majherová, PhD., Katolícka Univerzita, Ružomberok, SK  
RNDr. Pavel Pešat, PhD., Technická Univerzita, Liberec, CZ  
doc. RNDr. Tomáš Pitner, PhD., Masarykova Univerzita, Brno, CZ  
prof. Ing. Veronika Stoffová, CSc., Univerzita J. Selyeho, Komárno, SK  
RNDr. Ľubomír Šnajder, PhD., Univerzita Pavla Jozefa Šafárika, Košice, SK  
doc. RNDr. Pavel Töpfer, CSc., Univerzita Karlova, Praha, CZ  
doc. PaedDr. Jiří Vaníček, PhD., Jihočeská Univerzita, České Budějovice, CZ

**Organizačný výbor konferencie/Organizing Committee:**

Ing. Dana Horváthová, PhD.  
PaedDr. Ivan Brodenec, PhD.  
Katarína Gavaldová  
Ing. Jana Jacková, PhD.  
RNDr. Alžbeta Michalíková, PhD.  
RNDr. Miroslav Melicherčík, PhD.  
Mgr. Stanislav Slačka  
Mgr. Peter Trhan, PhD.  
Mgr. Michal Vagač, PhD.  
PaedDr. Darina Výbohová, PhD.

**Recenzenti/Reviewers:**

doc. RNDr. Gabriela Andrejková, CSc., Univerzita Pavla Jozefa Šafárika, Košice, SK  
PaedDr. Ján Beňačka, PhD., Univerzita Konštantína Filozofa, Nitra, SK  
PaedDr. Ivan Brodenec, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
PaedDr. Ján Guniš, PhD., Univerzita Pavla Jozefa Šafárika, Košice, SK  
Ing. Dana Horváthová, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
Ing. Jana Jacková, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
RNDr. Ľudmila Jašková, PhD., Univerzita Komenského, Bratislava, SK  
doc. RNDr. Zuzana Kubincová, PhD., Univerzita Komenského, Bratislava, SK  
RNDr. Gabriela Lovászová, PhD., Univerzita Konštantína Filozofa, Nitra, SK  
RNDr. Miroslav Melicherčík, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
PaedDr. Viera Michaličková, PhD., Univerzita Konštantína Filozofa, Nitra, SK  
RNDr. Ľubomír Salanci, PhD., Univerzita Komenského, Bratislava, SK  
prof. Ing. Veronika Stoffová, CSc., Univerzita J. Selyeho, Komárno, SK  
RNDr. Ľubomír Šnajder, PhD., Univerzita Pavla Jozefa Šafárika, Košice, SK  
doc. RNDr. Monika Tomcsányiová, PhD., Univerzita Komenského, Bratislava, SK  
RNDr. Peter Tomcsányi, PhD., Univerzita Komenského, Bratislava, SK  
doc. RNDr. Pavel Töpfer, CSc., Univerzita Karlova, Praha, CZ  
doc. Ing. Ľudovít Trajtel', PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
Mgr. Michal Vagač, PhD., Univerzita Mateja Bela, Banská Bystrica, SK  
doc. PaedDr. Jiří Vaníček, PhD., Jihočeská Univerzita, České Budějovice, CZ

Editor: doc. Ing. Ľudovít Trajtel', PhD.

Technický editor: PaedDr. Ivan Brodenec, PhD., Ing. Dana Horváthová, PhD., Ing. Jana Jacková, PhD.

Copyright © 2015 autori

ISBN: 978-80-557-0852-2

Recenzovaný zborník

# Predhovor

Milí účastníci konferencie!

Zase raz vedci hľadali vzťah medzi vzdelávaním sa a demenciou.

Najprv zistili, že vzdelávanie sa oddiaľuje nástup demencie s každým rokom v škole asi o dva a pol mesiaca. Neskôr však prišli s tým, že každý rok vzdelávania sa navyše znamená o približne 4 % rýchlejšiu stratu pamäti.

Našťastie (pre nás informatikov), existujú aj takí vedci, ktorí hovoria, že na informatické vzdelávanie sa sa to druhé zistenie nevzťahuje.

Lebo širším poslaním „informatiky“ je aj pomáhať pri vzdelávaní sa a pri výchove k poznaniu; zaslúži si preto, aby mala prirodzené a popredné miesto vo všeobecnom vzdelávaní sa každého jednotlivca tejto spoločnosti, a to bez ohľadu na jeho vek a odbornosť (i prípadné riziko demencie).

Najmä v poslednom období však začíname vnímať akýsi „šum“ v ponuke predmetov s informatickým obsahom a, do istej miery, nedoceňovanie aj/najmä teoretického významu informatiky; najmä na stredných (a viac na tých odborných) školách. Znepokojuje ma, že čoraz častejšie i hlasnejšie počuť názory o, na jednej strane, dnešnom vraj príliš teoretickom vzdelávaní budúcich informatikov i ich učiteľov a, na druhej strane, o praxi, ktorej vraj stačí oveľa menej, resp. treba jej vraj čosi iné.

Pravdou je, dobre si na to spomínam, že ani v prvých rokoch konferencie DidInfo to s presadzovaním „viac teoretickej“ koncepcie informatickej výučby nebolo bez problémov. Už vtedy nás, nadšených informatikov, stálo veľmi veľa energie presviedčanie, že snaha naučiť žiakov/študentov algoritmicky myslieť, tvoriť a programovať je vhodnejšia ako učenie len o používaní počítača. Ako naozajstní informatici sme veľmi rýchlo pochopili, že najväčšou devízou kvalitného informatika pre prax je práve jeho schopnosť abstraktného pohľadu na problém.

Avšak, práve istá nevyváženosť pohľadu na obsah a formu informatického vzdelávania je, logicky, motorom i základom úspechu našej konferencie DidInfo, ktorej ťažiskové témy sa každoročne dotýkajú pálčivých informatických problémov, aby nás vďaka tomu táto významná akcia nielen spájala, ale aj hnala k lepším informatickým zajtrajškom. Jej organizovaním sa snažíme pre každého z účastníkov vytvárať rámec pre vyjadrenie jeho vlastného pedagogického postoja, rámec pre informovanie iných, všetkých na konferencii prítomných, rámec pre odborný dialóg s priateľmi a kolegami. Jednoducho všetci chceme, aby práve/aj informatické vzdelávanie na základných, stredných i vysokých školách rozvíjalo kritické vnímanie, myslenie, vyššie poznávacie funkcie, flexibilné rozhodovanie a zvládanie každodenných praktických i teoretických problémov a, často, veľmi náročných situácií. Napokon, život v dnešnej modernej informačnej spoločnosti a efektívna účasť každého pri riešení problémov praxe si to priamo vyžaduje.

Teší ma preto veľmi, že sa nás tu aj tento rok opäť stretá veľa takých, ktorí máme skutočný záujem vstúpiť do otvoreného dialógu, máme chuť referovať o svojich výsledkoch, chceme sa poučiť zo skúseností kolegov a priateľov. Teším sa z účasti pozvaných prednášateľov, všetkých ďalších aktívne vystupujúcich, všetkých prítomných učiteľov a rovnako ma teší aj rozhodnutie zástupcov z vedecko-výskumných i obchodných pracovísk alebo zástupcov z iných príbuzných odborov zúčastniť sa na našej konferencii.

A rád by som ešte zdôraznil, že aj v tomto roku s radosťou prijímame záštitu Ministerstva školstva SR nad našou konferenciou. A rovnako si vážime podporu všetkých sponzorov i mesta Banská Bystrica.

Nech sa nám teda konferencia DidInfo 2015 k spokojnosti všetkých zúčastnených vydarí!

Ďakujem Vám za Vašu pozornosť!

Želám Vám všetkým krásny prvý konferenčný deň i krásne všetky nasledujúce konferenčné dni.

Apríl 2015

doc. Ing. Ľudovít Trajtel', PhD.  
predseda programového výboru

# Obsah

## Pozvané prednášky

Miroslava Černochová, Hanka Šandová	
<b>První ohlédnutí za výukou základů programování ve Scratch na ZŠ aneb Čím nás žáci překvapili i zaskočili, co musíme příště dělat jinak.....</b>	<b>7</b>
Zuzana Kubincová	
<b>IT Fitness test 2014.....</b>	<b>12</b>
Lubomír Šnajder	
<b>Metodika výučby programovania mobilných zariadení.....</b>	<b>18</b>
Tatiana Hajdúková, Ivan Bacigál	
<b>Ochrana detí pred nebezpečenstvom, ktoré im hrozí z prostredia internetu.....</b>	<b>28</b>

## Krátke prednášky

Ján Beňačka	
<b>Odvozenie vzorca pre obsah elipsy a jeho overenie metódou Monte Carlo v Exceli na gymnáziu.....</b>	<b>34</b>
Vladimír Bolek, Anikó Törökóvá	
<b>Proces zmeny vzdelávacích metód vplyvom učebných štýlov.....</b>	<b>37</b>
Róbert Borbély, Gabriel Stach	
<b>Interaktívne testovanie žiakov v prostredí Samsung School a Socrative.....</b>	<b>42</b>
Veronika Dropčová, Zuzana Kubincová	
<b>Peer review pri hodnotení projektov.....</b>	<b>47</b>
Tomáš Dudík, Erika Lettrichová	
<b>ITfactor 2014.....</b>	<b>51</b>
Rostislav Fojtík	
<b>Vzdělávání budoucích učitelů informatiky.....</b>	<b>53</b>
Roman Horváth, Veronika Stoffová	
<b>Nový spôsob hodnotenia domácich prác z programovania.....</b>	<b>56</b>
Dana Horváthová	
<b>Kurz Multimédiá v LMS Moodle.....</b>	<b>60</b>
Roman Hrušecký, Andrea Hrušecká, Monika Tomcsányiová, Ivan Kalaš	
<b>Tvorivá informatika – Prezentácie.....</b>	<b>65</b>
Milan Hudec	
<b>Automatizovaná príprava hlasového korpusu.....</b>	<b>69</b>
Zdeněk Hájek, Peter Vasaráb	
<b>Skúsenosti s programovaním na základnej škole.....</b>	<b>72</b>
Ľudmila Jašková, Natália Kováčová	
<b>Prvé skúsenosti s realizáciou súťaže iBobor pre nevidiacich žiakov druhého stupňa ZŠ.....</b>	<b>78</b>
Jan Jára	
<b>Informatika v bicím sekvenceru ukrytá.....</b>	<b>84</b>
Alžbeta Kanáliková	
<b>Programovanie, alebo hra?.....</b>	<b>89</b>
Ján Krejčí, Jiří Šretr	
<b>Použití mechatronických systémů jako motivačního prvku nejen ve výuce informatiky.....</b>	<b>93</b>
Daniel Lessner	
<b>Tvorba kurikula pro výuku informatiky jako vědy pro česká gymnázia.....</b>	<b>96</b>
Gabriela Lovászová, Viera Michaličková	
<b>Hlasovacie systémy - interaktívna technológia na podporu aktívneho učenia sa.....</b>	<b>102</b>



Eva Mészárosová, Monika Tomcsányiová	
<b>Vyučovanie základov programovania v jazyku Python na niektorých SŠ a VŠ.....</b>	<b>108</b>
Václav Nádvorník	
<b>Videonávody na školní webové stránky.....</b>	<b>112</b>
Rudolf Pecinovský	
<b>Metodika Architecture First .....</b>	<b>118</b>
Pavel Pešat	
<b>E-Skills for Jobs in Education na Univerzitě J. E. Purkyně.....</b>	<b>123</b>
Peter Potocký	
<b>Využívanie disponibilných hodín k aktualizovaniu školského vzdelávacieho programu.....</b>	<b>125</b>
Ľubomír Salanci	
<b>Analýza informatiky v novom štátnom vzdelávacom programe.....</b>	<b>129</b>
Mária Spišáková, Ľubomír Salanci	
<b>Meranie motivácie žiakov ku výkonu.....</b>	<b>139</b>
Mária Stankovičová, Mária Kaliaková, Zdenka Vitálišová	
<b>Informatické kompetencie žiakov ZŠ so zrakovým postihnutím a ich využitie na vyučovaní matematiky a fyziky používaním matematického editora LAMBDA.....</b>	<b>144</b>
Veronika Stoffová, Kristína Czaková	
<b>Semantický diferenciál v pedagogickom výskume.....</b>	<b>150</b>
Peter Tomcsányi, Andrej Blaho	
<b>Programovací jazyk Python vo výučbe predmetu Algoritmy a dátové štruktúry.....</b>	<b>155</b>
Monika Tomcsányiová, Peter Tomcsányi	
<b>Vyučovanie programovania vývojom androidových aplikácií v ApplInventore.....</b>	<b>160</b>
Peter Trhan, Svetlana Gáperová, Ján Gáper	
<b>Elektronická učebnica pre iPad.....</b>	<b>164</b>
Jiří Vaníček	
<b>Programování ve Scratch badatelsky orientovaným přístupem.....</b>	<b>169</b>
Michaela Veselovská, Karolína Mayerová	
<b>Aktivity s LEGO WeDo na 2. stupni ZŠ.....</b>	<b>175</b>
Dana Šuníková, Zuzana Kubincová	
<b>Čo sú Open Badges a ako ich použiť?.....</b>	<b>181</b>

## **Postery**

Brodeneč, Horváthová, Klimeš, Škrinárová	
<b>Desaťročie výučby informatiky na Univerzite tretieho veku na UMB v Banskej Bystrici.....</b>	<b>188</b>
Gašperanová	
<b>Implementácia informačných a komunikačných technológií do vzdelávania na národohospodárskej fakulte ekonomickej univerzity v Bratislave.....</b>	<b>189</b>
Jacková, Trajtel'	
<b>Medzipredmetové vzťahy v predmete semestrálny projekt.....</b>	<b>190</b>
Majherová	
<b>Inovácie kurikula v informatike na ZŠ.....</b>	<b>191</b>

## **Pracovné semináre**

## **Organizátori**

## **Sponzori a mediálni partneri**

*Pozvané přednášky*

*Invited Papers*

# První ohlédnutí za výukou základů programování ve Scratch na ZŠ aneb Čím nás žáci překvapili i zaskočili, co musíme příště dělat jinak

Miroslava Černochová  
Univerzita Karlova v Praze, Pedagogická fakulta  
Magdalény Rettigové 4  
116 39 Praha 1  
Česká republika  
Miroslava.cernochova@pedf.cuni.cz

Hanka Šandová  
Gymnázium Jana Keplera  
Parléřova 2  
169 00 Praha 6  
Česká republika  
sandova@gjk.cz

## ABSTRAKT

Příspěvek seznamuje se zkušenostmi získanými ve výuce základů programování ve SCRATCH se dvěma různými skupinami žáků (ve věku 11-12 let) jako součásti povinné výuky v rámci vzdělávací oblasti ICT na dvou pražských školách. V týmové spolupráci žáci programovali ve Scratch své vlastní příběhy. Teoretickým východiskem pro metodické přístupy k výuce programování byl princip tvorby jako způsobu poznávání ([10]). Pomocí rozhovorů se žáky, s použitím dotazníkového šetření a na základě průběžného hodnocení a analýzy výsledků žákovských prací jsme zjišťovali dopad tvůrčího procesu na představy žáků o počítači a na osvojení několika základních pojmů. Velký důraz byl kladen na mezipředmětovou spolupráci, která se rozhodujícím způsobem podílí na osvojování různých gramotností žáků (gramotnosti čtenářské, matematické, gramotnosti literární a vizuální). Tento výukový projekt „Vyprávěj příběh a zkus ho naprogramovat“ navazuje na výsledky mezinárodní spolupráce v projektu Literacy from Scratch ([5]) (<http://www.literacyfromscratch.org.uk/>).

## ABSTRACT

In this article the authors present some experiences from 20 lessons teaching programming in SCRATCH with two different groups of 11-12 years' old pupils in two schools in Prague as part of compulsory ICT. Working in teams and based on their original graphics, pupils programmed their own stories in Scratch. The underpinning theory for such a pedagogical approach was based on that of creativity as a cognitive process ([10]). Using interviews, questionnaires, evaluation of pupils' products and synchronous assessment, the authors analysed the effect of the creative process on pupil's images, ideas and thinking about computer functions and related concepts. During the teaching, a particular emphasis was put on interdisciplinarity, which significantly contributes to development of different forms of literacies (e.g. reading, mathematical, writing, digital and visual literacy). The work forms part of an international initiative, 'Literacy from Scratch' ([5]) (<http://www.literacyfromscratch.org.uk/>).

## Klíčové slová

program, programování, story-telling, Scratch, představa žáka, počítač, gramotnost

## Keywords

Program, programming, story-telling, Scratch, pupil's idea, computer, literacy

## 1. ÚVOD

Ve světě se teď hodně diskutuje o návratu k programování do školního vzdělávání, o potřebě rozvíjet informatické myšlení (computational thinking) „jako způsobu *„řešení problémů, navrhování systémů a porozumění lidskému chování s použitím klíčových konceptů čerpajících z počítačových věd a informatiky“* ([11]). ACM ([1]) chápe informatické myšlení jako „*metodologii pro řešení problémů, které jsou úzce provázány s počítačovými vědami/informatikou a dalšími obory, poskytující specifické prostředky pro analýzu a hledání řešení problémů, které mohou být řešeny automatickým výpočtem*“<sup>1</sup>. Bort a Brylow ([2]) zdůrazňují, že „*informatické myšlení je vlastně metoda pro řešení problémů využívající abstrakci a analytické myšlení k dosažení nejlepšího řešení. Nejde o použití počítače pro vyřešení problému, ale o myšlení založené na algoritmizaci.*“

Řada zemí (Anglie, Austrálie, Litva aj.) na rozdíl od ČR zavedla předmět informatika nebo začala do školního vzdělávání implementovat koncept informatického myšlení. V souvislosti s dokumentem Strategie digitálního vzdělávání do 2020 ([7]), který schválila vláda ČR v listopadu 2014, však lze očekávat, že v brzké době už i v ČR začnou školy přemýšlet o jiných přístupech k výuce vzdělávací oblasti ICT v RVP. Některé školy inspirovány některými výsledky ze zahraničí už zkoušejí vlastní cestu k výuce ICT a navrhuji metodické postupy pro utváření algoritmického myšlení žáků. Jedním z takových pokusů je i projekt „Vyprávěj příběh a zkus ho naprogramovat“ realizovaný jako součást mezinárodní iniciativy Literacy from Scratch.

## 2. VÝCHODISKA

Východiskem pro přípravu projektu „Vyprávěj příběh ...“ se stalo několik skutečností:

- (1) Postavení tématu algoritmizace a programování v kurikulárním dokumentu RVP ZV

Cílové zaměření vzdělávací oblasti ICT v kurikulárním dokumentu RVP ZV ([9], s. 32) nezmiňuje potřebu rozvíjet algoritmické myšlení žáků či věnovat pozornost programování. Tyto dva koncepty se neobjevují ani v učivu a výstupech dílčích tématických částí Vzdělávací oblasti ICT ([9], s. 33-34), ani v návrhu standardu pro ICT na ZŠ ([3]). Vzdělávací oblast ICT se podle ([9]) zaměřuje hlavně na utváření základních uživatelských dovedností žáků pracovat s počítačem, vyhledávat informace na Internetu a prezentovat je za použití různých aplikačních

<sup>1</sup> computationally

programů. Rovněž tematický celek Využití digitálních technologií zařazený do vzdělávací oblasti Člověk a svět práce ([9], s. 84-89) se soustředí na osvojení běžných uživatelských činností nezbytných pro obsluhu digitálních technologií.

## (2) Algoritmizace a programování ve školní praxi na 2. stupni ZŠ

Výzkum provedený v rámci GA ČR ([8]) mezi učiteli ICT na 2. stupni ZŠ odhalilo a současně potvrdilo, že výuka ICT předmětů na 2. stupni ZŠ se v ČR zaměřuje především na práci s Internetem (vyhlédávání informací, e-mail) a na uživatelské dovednosti používat základní aplikace (textový editor, tabulkový procesor, grafický editor a zpracování grafiky). Jen zcela výjimečně se učitelé ICT věnují rozvoji algoritmického myšlení žáků a tématům z programování ([8], [4]). Potvrzují to i rozhovory se studenty dvouletého navazujícího magisterského kombinovaného jednooborového kombinovaného studia ICT, kteří využívají ICT předměty na ZŠ. Základy programování zařazují do ŠVP jen ti učitelé, kteří absolvovali bakalářské/magisterské studium na technických vysokých školách nebo při studiu informatiky, případně nějakou dobu pracovali jako programátoři.

## (3) Didaktické přístupy k učení

Současná teoretická didaktická východiska k učení jako ke specifickému případu poznávání stále víc a víc zdůrazňují význam aktivní činnosti žáka, zejména pak tvůrčí činnosti, v jejímž průběhu vznikají různé typy artefaktů včetně artefaktů mentálních; s tvorbou artefaktu zcela úzce souvisí i obsah poznávací činnosti, objevování struktury obsahu poznání a osvojování významu používaných jazykových výrazů a gramatik ([10]). Edukační programování pak lze chápat jako proces tvorby vedoucí ke specifickým artefaktům, během níž dochází k poznávání, které probíhá v mentálním prostoru, v němž se organizují subjektivní zkušenosti.

A právě tvorba počítačových programů ve školách s využitím vhodných edukačních programovacích prostředí může sehrát významnou roli v učení žáků a v osvojování různých gramotností (matematické, čtenářské, jazykové, vizuální aj.) včetně gramotnosti digitální.

Zkušenosti z 2012/13 ([5]) ukázaly, jak důležité je, aby žáci různými formami (ústně, písemně, graficky, strukturovaně) vyprávěli, sdělovali, přemýšleli o svých projektech, o svých záměrech.

## (4) Vládní strategie pro vzdělávání v ČR

Ačkoliv se v ČR zatím o potřebě zařadit témata z informatiky do ICT v kurikulárním dokumentu pro ZŠ nehovoří, tak už se podařilo do vládního dokumentu Strategie digitálního vzdělávání do 2020 ([7], s. 46) schváleného vládou ČR 12.11.2014 začlenit požadavek rozvíjet digitální kompetence a informatické myšlení žáků a učitelů. Je to důležitý signál, že se v ČR budeme v brzké době zabývat otázkou zavedení některých klíčových pojmů z informatiky (včetně metodiky rozvoje algoritmického myšlení a základů tvorby počítačových programů) do kurikula školního vzdělávání a do vysokoškolské přípravy a do dalšího vzdělávání učitelů. Potéšitelné je, že také MPSV ve spolupráci s MŠMT začíná věnovat pozornost otázce digitální gramotnosti ([6]).

## 3. PROJEKT „VYPRÁVĚJ PŘÍBĚH ...“

Jedním ze způsobů, jak můžeme učitelům ZŠ pomáhat při hledání nových přístupů ke vzdělávací oblasti ICT, je připravit s budoucími ICT učiteli konkrétní výukový projekt, seznámit

s ním vedení vybrané školy, vyškolit její učitele a výukový projekt ve spolupráci s učiteli školy se žáky ověřovat. Tento didaktický experiment přispívá k utváření profesních dovedností budoucích ICT učitelů, kteří svůj výukový projekt v podmínkách konkrétní školy realizují, k vyškolení učitelů školy, kteří po skončení projektu pokračují v podobných aktivitách se svými žáky, k vytvoření metodických materiálů pro učitele a učebních materiálů pro žáky, k získání zkušeností, které můžeme využít v další vysokoškolské přípravě ICT učitelů, a k získání poznatků o tom, do jaké míry je současná škola připravena na nové přístupy k výuce ICT. Tento didaktický model jsme již úspěšně vyzkoušeli v 2012/13 ([5]) a byl aplikován i v projektu „Vyprávěj příběh ...“, který se uskutečnil se dvěma různými skupinami žáků ve věku 11-12 let od září 2014 do konce ledna 2015.

Projekt „Vyprávěj příběh ...“ se realizoval v rámci povinné výuky pravidelně 1h/týdně v 6. ročníku ZŠ a v primě osmiletého gymnázia pod vedením stejné učitelky ICT předmětů s tím, že zatímco žáky z primy na gymnáziu učitelka velmi dobře znala, tak se žáky 6. ročníku ZŠ se setkávala jen a pouze na jejich povinné výuce jako hostující učitel. Přípravu na výuku pro žáky ZŠ konzultovala učitelka s ředitelem školy, za jehož přítomnosti také výuka na ZŠ probíhala. Přípravy na hodinu a materiály pro žáky jsou zveřejněny na webových stránkách <http://scratch.sandofky.cz>. Na obou školách probíhala výuka v příjemné atmosféře v počítačové učebně s dataprojekcí.

### 3.1 Cíl projektu

Cílem výukového projektu „Vyprávěj příběh ...“ bylo nejen navrhnout a ověřit metodiku pro výuku programování ve SCRATCH pro žáky ve věku 11-12 let, ale také pokusit se zjistit dopad aktivit spojených s programováním ve Scratch na představy žáků o počítači a na chápání některých základních pojmů. Cílem tohoto výukového projektu bylo rozvíjet komplex základních gramotností: čtenářskou, výtvarnou, digitální, jazykovou, matematickou, hudební a vizuální s důrazem na mezipředmětové vazby. Projekt v žádném případě neusiloval o srovnávání ICT znalostí a dovedností a výsledků prací žáků ZŠ s výsledky žáků osmiletého gymnázia.

### 3.2 Cílová skupina

Do projektu byli zapojeni žáci 6. ročníku ZŠ a primy osmiletého gymnázia ve věku 11-12 let.

Skupinu žáků ZŠ tvořilo celkem 22 žáků (10 chlapců, 12 děvčat). ŠVP, podle něhož byli žáci ZŠ vzděláváni, zdůrazňuje kreativitu žáků a tvůrčí přístupy učitelů k výuce a učení. Žáci této školy jsou zvyklí přistupovat k úkolům tvůrčím způsobem, a to nejen v umělecky zaměřených předmětech. Pokud žáci během výuky něčemu nerozumějí, dávají to najevo svým chováním. Žáci jsou velice spontánní, rádi o problémech diskutují. Jsou zvyklí dělat věci podle svých představ. ŠVP této ZŠ obecně do výuky Informatiky nezařazuje problematiku algoritmizace či programování, nicméně žáci 6. ročníku se s jednoduchými základy práce ve Scratch seznámili již ve školním roce 2013/14. Informatika se na této ZŠ zaměřuje hlavně na tvůrčí práci s vybranými aplikacemi, nikoliv na technické principy konstrukce a funkce počítače.

Gymnaziální skupinu tvořilo celkem 14 žáků z primy (12 chlapců, 2 dívky). Tito žáci byli přijati na gymnázium na základě přijímacího řízení. V hodinách povinného předmětu Informatika jsou zvyklí používat jednoduché algoritmické postupy, s prostředím Scratch se seznamovali až v projektu „Vyprávěj příběh...“. ŠVP tohoto gymnázia věnuje velkou pozornost tématu hardware počítače a principu fungování počítače.

### 3.3 Výzkumná otázka

Při realizaci projektu „Vyprávěj příběh ...“ nás zajímalo, s jakými představami o počítači žáci pracují? Potřebují vůbec žáci nějakou představu o počítači mít? Změní se představy žáků o tom, co umí počítač a proč počítač umí právě to, co umí, když si žáci vyzkouší nějaké programování? Ovlivní první programovací zkušenosti žáků představu o tom, co rozhoduje o tom, že počítač člověku umožňuje provádět na počítači různé činnosti, služby, úkony?

### 3.4 Výzkumné metody

Pro nalezení odpovědi na výzkumnou otázku byly použity především kvalitativní metody kombinované s běžnými kvantitativními metodami, konkrétně se jednalo o metodu soustavného pozorování, metodu rozhovoru se žáky, metody pro analýzu a evaluaci žákovských prací, dotazníkové šetření, metodu testování s přiřazovací testovou úlohou a s testovou úlohou typu volná odpověď (esej).

**Tabulka 1 Použité výzkumné metody**

Časová etapa	Použitá metoda
Na začátku projektu	dotazníkové šetření (DOT1)
V průběhu projektu	metoda soustavného pozorování rozhovor se žáky metody pro analýzu a průběžnou evaluaci žákovských prací test s přiřazovací testovou úlohou (TEST1) zpětná vazba: co jsi se naučil/a?
Na závěr projektu	test s přiřazovací testovou úlohou (TEST2) test s testovými úlohami typu volná odpověď (ESEJ) dotazníkové šetření (DOT2)

### 3.5 Metodický postup

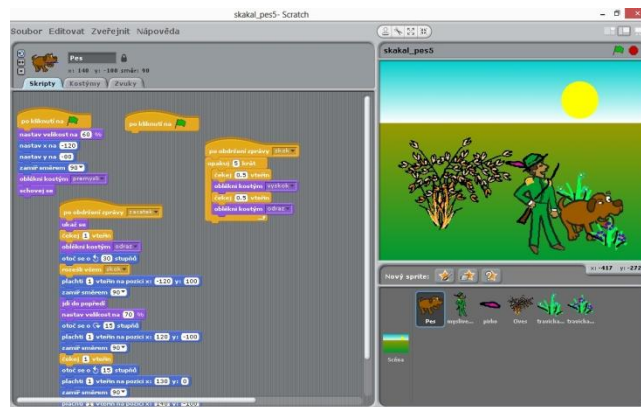
Žáci pracovali ve dvoučlenných týmech, každý tým si pro svůj tým vymyslel název (např. Lemuři). Každý žák mohl pracovat u svého počítače, každý tým tak měl k dispozici dvě počítačové stanice. Kladli jsme důraz na to, aby každý člen týmu cítil odpovědnost za práci na projektu, aby se podílel na řešení, aby si členové týmu práci na projektu rozumně mezi sebou rozdělili a pak dílčí výsledky spojovali. Výuka byla rozdělena do pěti fází:

#### *Fáze 1: Analýza hotové ukázky a její následná tvorba (reprodukce)*

Na hotové ukázce příběhu, vycházejícího z lidové písničky „Skákal pes přes oves“, se žáci seznámili s některými základními pojmy a příkazy Scratch (Obrázek 1) a se strukturou konkrétního programu. Žáci měli k dispozici pracovní list s úkoly, do něhož zakreslovali jednotlivé části programu a zkoumali strukturu programu a funkce konkrétních příkazů.

Tato ukázka sloužila i jako návod pro žáky, jak postupovat, až budou mít připravený scénář svého příběhu, čím začít, o čem je zapotřebí přemýšlet. Každý žákovský tým pak ve Scratch sestavoval na základě pracovního listu s pokyny a hotových spritů a pozadí pro scény program „Skákal pes přes oves“. Svá řešení pak týmy porovnávaly s ukázkou, kterou jim učitelka zpřístupnila.

Ukázka pak ve Fázi 4 sloužila při vysvětlování některých základních pojmů (proměnná, cyklus, algoritmus, aj.).



**Obrázek 1 Ukázka písničky Skákal pes**

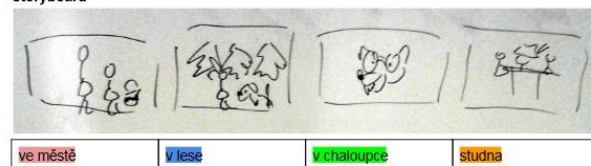
#### *Fáze 2: Tvorba scénáře vlastního příběhu (týmová práce)*

Tvorbě scénáře byla věnována velká pozornost. Žáci nejprve diskutovali o tom, o čem by měli vyprávět, jak by měl příběh vypadat, jaké příběhy nejsou vhodné a proč. Každý příběh se měl postupně odehrávat na čtyřech scénách. Každý tým nejprve verbálně vyprávěl několika větami hlavní linii svého příběhu. Poté ztvárnil tento příběh v podobě slohového útvaru vyprávění; v textu pak barevně vyznačil čtyři časoprostorové úseky odpovídající čtyřem scénám. Potom žáci zakreslili graficky v podobě zjednodušeného „komiksu“ čtyři po sobě jdoucí scény příběhu, přičemž každý úsek (scénu) bylo zapotřebí pojmenovat. Názvy všech scén doplnili v přehledné tabulce seznamem postavíček, které na scéně vystupují. Nakonec žáci popsali podrobně každou scénu (Obrázek 2).

#### **Cervená Karkulka**

Maminka vyslala Karkulku poprát babičce k svátku. Cestou v lese se Karkulka zdržela trhaním květin pro babičku a potkala vika. Vik Karkulce ukázal delší cestu a Karkulku předběhl. Když Karkulka přišla k babiččině chaloupce, ležel v posteli vik, který před tím snědl babičku. Karkulce se to nezdálo, tak se vyptávala a vik jí také sežral. Naštěstí šel okolo myslivec. Dovtlipil se, co se stalo a zachránil babičku i Karkulku. Do rozpáraného vikova břicha zašil kamení. Vikovi, když se probudil, bylo těžko. Šel se napít a přepadl do studny.

#### **Storyboard**



#### **Seznam sprite**

Maminka	Karkulka	Karkulka	Vik
Karkulka	Vik	Vik	Myslivec
Košík	Květiny		
Babovka	Stromy		
Sirup			

#### **Podrobný scénář**

##### **scéna 1: ve městě**

Maminka dává věci pro babičku do košíku a podává ho Karkulce. Karkulka bere košík a odchází. Mávali si s maminkou.

**Obrázek 2 Návrh scénáře**

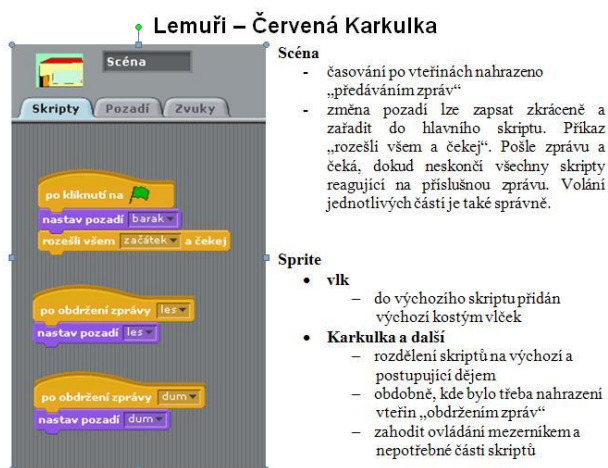
#### *Fáze 3: Týmová práce na vlastním projektu (originální žákovská tvorba)*

Práce týmů byla po celou dobu průběžně sledována a hodnocena. Týmy dostávaly zpětnou vazbu písemně v podobě tištěného dokumentu, který obsahoval kopie obrazovek žákovských řešení doplněné komentářem učitelky a doporučeními, jak má tým dál



postupovat, co má udělat jinak, na co si má dávat pozor (Obrázek 3).

Kontrola příběhů verze z 15. prosince 2014



Obrázek 3 Ukázka hodnocení práce týmu Lemuři

#### Fáze 4: Základní pojmy

S využitím hotové ukázky „Skákal pes přes oves“ se žáci seznámili s těmito základními pojmy: algoritmus, program, sekvence příkazů, vstup, výstup, proměnná, cyklus, větvení, ladění (Obrázek 4). Po vysvětlení a procvičení významu těchto pojmů žáci řešili testovou přiřazovací úlohu (TEST1), na níž měli předvést míru porozumění těmto pojmům.



Obrázek 4 K pojmu větvení

#### Fáze 5: Předvedení výsledků týmové práce ve Scratch

Na poslední lekci každý tým předvedl svůj výsledný projekt. Některé projekty byly hotové, některé se nepodařilo některým týmům dokončit (nemoc, nesoustředěnost, podcenění spolupráce, nevyužití týmové práce, neodpovědný přístup k práci, ...).

#### Fáze 6: Zpětná vazba o tom, co se žáci naučili (dotazník, test, samostatná úloha)

Na závěr projektu každý žák dostal dotazník (DOT2) se třemi otevřenými otázkami, v němž se pokoušeli popsat svou vlastní představu o počítači, jak funguje a co je s ním možné provádět, a testovou přiřazovací úlohu (TEST2, Obrázek 5).

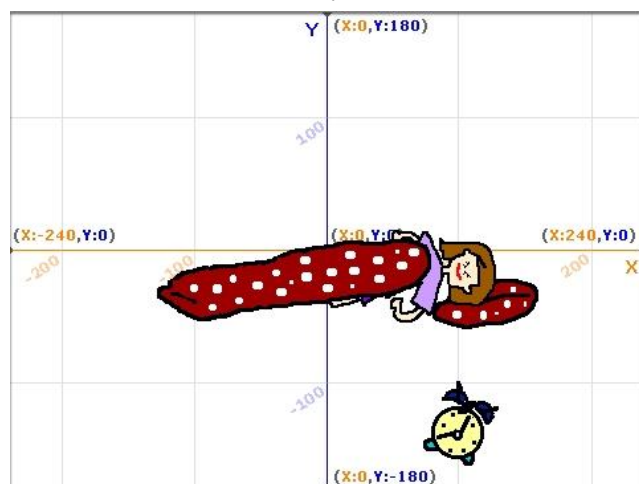
V týmu pak řešili úlohu, v níž měli žáci ukázat, že dovedou číst a interpretovat na tiskárně vytištěný výpis programu ve Scratch. Žáci dostali z papíru vystříhané postavičky, které na základě údajů vyčtených z výpisu programu umísťovali do pracovního listu se souřadným systémem (Obrázek 6). Vlastními slovy pak žáci

popisovali, jaký se děje po spuštění programu, jaký příběh byl ve Scratch naprogramován.

Ke každému pojmu najdi definici, která jej podle Tebe nejlépe vystihuje ... Odpovídající písmeno u příslušné definice v pravém sloupci запиš do čtverečku u pojmu v levém sloupci.

	Program	A – Obecný popis řešení problému krok za krokem.
	Sekvence příkazů	B – Hledání a náprava chyb v počítačovém programu.
	Větvení	C – Data která vkládáme do počítače.
	Vstup	D – Počítačem vrácená zpracovaná data.
	Proměnná	E – Seznam pokynů, které říkájí počítači co má přesně dělat.
	Cyklus	F – Opakování nějakého příkazu více než jedenkrát.

Obrázek 5 Ukázka testové úlohy TEST2



Obrázek 6 Umístění postaviček do souřadného systému

### 3.6 První výsledky k zamyšlení

Na začátku projektu byl žákům zadán dotazník (DOT1) s cílem zjistit jejich počáteční představy o počítačích. Z dotazníků vyplynulo, že žáci se učí ovládat počítač především od svých sourozenců, rodičů a dalších rodinných příslušníků, teprve pak ve škole nebo sami bez jakékoliv další pomoci. Přibližně polovina rodičů dotázaných žáků pracuje s počítačem, na němž rodiče používají Facebook, vyhledávají informace, tisknou a evidují faktury nebo navrhuji „architektonické věci“. Přestože žáci u počítače a podobných digitálních zařízení tráví nemalý čas, dotazník neprokázal, že by žáci měli nějaké hlubší povědomí o počítačích a že by měli nějaké jasné představy o tom, jak počítač funguje, pracuje.

Řada žáků ve věku 11-12 let se s programováním počítače mimo školu nesetkává. Tyto děti nemají představu o tom, proč mohou pomocí počítače provádět různé činnosti (hrát hry, komunikovat aj.). Na této skutečnosti se bohužel podílí i zkušenosti dětí s tablety; práce s tabletem se přece jen liší od práce s běžným počítačem. Žáci vnímají počítač jako zařízení s hotovými aplikacemi, jako zařízení na stahování programů nebo prohlížení fotek, na komunikaci na Facebooku a na vyhledávání informací na Internetu.

#### 4. ZÁVĚR

Práce se žáky ZŠ a gymnázia přinesla řadu překvapení. Na otázku, čemu se žáci ve výuce se Scratch naučili, tak většina žáků ZŠ odpovídala, že se v hodinách naučili především to, jak zanačňovat zvuk pro své aplikace ve Scratch; žáci měli radost, že se jejich postavičky mohou mezi sebou dorozumívat prostřednictvím namluvených dialogů nebo že mohou vydávat různé zvuky. Na tom, že žáci pozitivně ocenili tuto zkušenost se záznamem zvuku, se zřejmě nejvíce podílel fakt, že počítačová učebna nebyla sluchátky s mikrofony běžně vybavena, práce s mikrofonom nepatřila mezi běžné každodenní činnosti dětí ve škole.

Na práci se žáky ZŠ měl velký vliv to, že se někteří žáci neuměli dlouhodobě soustředit na práci, že něco začali a rychle od započatého díla odbíhali. Velkým problémem byly i sociální vztahy uvnitř některých týmů; stalo se, že uprostřed rozdělané práce si žáci v týmu přestali rozumět, takže jsme museli tým rozdělit a každý člen týmu pracoval sám – obvykle nechtěl pokračovat v příběhu, který si v týmu s kamarádem připravil. Dívčí týmy neměly problémy se spoluprací a vztahy uvnitř týmu, zatímco několik chlapeckých týmů se „rozkmotilo“.

Ne vždy se žáci ZŠ drželi striktně svých připravených scénářů; v závislosti na tom, čemu se postupně se Scratchem naučili, scénář měnili. Dost času zbytečně také žáci ztráceli nad zbytečnými diskusemi o dalším vývoji jejich příběhu, ačkoliv již práce na scénáři byla ukončena.

Žáci gymnázia byli připraveni pracovat samostatně a práce v týmech jim nečinila problémy. Týmová práce jim pomáhala upevňovat kamarádské vztahy na nové škole (žáci primy). Řada z nich pokračovala na svém příběhu i po skončení výuky a stihla doplnit příběh jednoduchou hrou. Ukázalo se, že výuka Scratch celé jedno pololetí s takovými žáky už není záživná, někteří žáci dávali najevo, že už je to nudí. Zřejmě to bylo tím, že někteří žáci navštěvují nepovinný předmět „Programování v Pythonu“, a tak se jim Scratch „nezdál dost dobrý“. Jedná se převážně o chlapeckou třídu (ze 14 žáků 2 dívky), polovina chlapců se o programování zajímá aktivně nad rámec předmětu Informatika, takže své příběhy doplňovali jednoduchými hrami nebo pomáhali/radili svým spolužákům, popř. Se Scratchem experimentovali a vytvářeli další projekty. Velmi silná a viditelná byla spolupráce všech žáků, navzájem si pomáhali a povzbuzovali se, věděli, koho se na co zeptat, po úvodních hodinách pracovali stále více a více samostatněji.

Zatímco na ZŠ se dařila mezipředmětová spolupráce (český jazyk, výtvarná výchova, hudební výchova) podporovaná ředitelem školy, na gymnáziu projekty vznikaly jen v rámci výuky Informatiky. Nicméně i v případě ZŠ bude zapotřebí věnovat v projektech typu „Vyprávěj příběhy ...“ velkou pozornost dopracování metodiky mezipředmětové spolupráce při rozvíjení dílčích gramotností žáků a způsobu hodnocení jejich práce.

#### 5. BIBLIOGRAFICKÉ ODKAZY

- [1] ACM (2011) *Curricula Recommendations, CSTA K-12 Computer Science Standards, Revised 2011*. Retrieved August 31, 2012. Available on <http://www.acm.org/education/curricula-recommendations>.
- [2] BORT, H., BRYLOW, B. (2013) CS4Impact: Measuring Computational Thinking Concepts Present in CS4HS Participant Lesson Plans. In: *SIGCSE '13 Proceeding of the 44th ACM technical symposium on Computer science education*, 2013. Available on

<http://db.grinnell.edu/sigcse/sigcse2013/Program/viewAcceptEdProposal.pdf?sessionType=paper&sessionNumber=262>.

- [3] BRDIČKA, B. et al. (2013) *Standardy pro základní vzdělávání. Informační a komunikační technologie*. Pracovní verze z 30. 4. 2013. Zpracováno dle upraveného RVP ZV platného od 1. 9. 2013
- [4] ČERNOCHOVÁ, M. (2010) Teaching practice from the perspective of ICT student teachers at the Faculty of Education, Charles University in Prague. In *Key Competencies in the Knowledge Society. IFIP TC 3 International Conference, KCKS 2010. Held Part of WCC 2010 Brisbane, Australia, September 2010, Proceedings*. (Eds.) Nicholas Reynolds, Márta Turcsányi-Szabó. IFIP AICT 324. ISSN 1868-4238. Springer, Printed in Germany. pp. 44-55.
- [5] ČERNOCHOVÁ, M., KOMRSKA, T. (2013) Scratch v hodinách výtvarné výchovy a ICT na 1. stupni ZŠ aneb žáci 1. stupně ZŠ vyprávějí příběhy. In: *DIDINFO 2013*. Ed. L. Trajtel. Banská Bystrica, 2013. s. 21-26.
- [6] MPSV, MŠMT (2015) *Strategie digitální gramotnosti ČR na období 2015 až 2020*. Praha, leden 2015.
- [7] MŠMT (2014) *Strategie vzdělávací politiky České republiky do roku 2020*. MŠMT : Praha, 2014.
- [8] RAMBOUSEK, V. et al. (2013) *Rozvoj informačně technologických kompetencí na základních školách*. Praha: Česká technika.
- [9] RVP ZV (2013) *Úplné znění upraveného RVP ZV. Verze platná k 1.9.2013*. Příloha č. 2 k Opatření ministra školství, mládeže a tělovýchovy, kterým se mění Rámcový vzdělávací program pro základní vzdělávání, č.j. MSMT-2647/2013-210. Praha, 2013.
- [10] SLAVÍK, J. et. al. (2014) *Tvorba jako proces poznání*. Karolinum : Praha, 2014.
- [11] WING, J. (2006) Computational thinking. In: *Communications of the ACM*, 49:33-35, 2006.

# IT Fitness test 2014

Zuzana Kubincová

FMFI UK

Mynská dolina

84248 Bratislava

Štát

kubincova@fmph.uniba.sk

## ABSTRAKT

Testovanie IT zručností prostredníctvom IT Fitness testu sa v roku 2014 konalo na Slovensku už po štvrtý raz. Hlavnú cieľovú skupinu tvorili už tradične mladí ľudia zodpovedajúci vekovej kategórii študentov stredných a vysokých škôl. Tento rok bol po prvýkrát do testovania začlenený aj špeciálny test určený žiakom základných škôl. V prednáške prinesieme analýzu testu pre študentov, ako aj výsledky výskumu dát získaných v informačnej časti testovania. Zameriame sa aj na úspešné a neúspešné oblasti testovania a porovnáme výsledky študentov z rozdielnych typov škôl.

## ABSTRACT

In 2014, the IT Fitness test was carried out for the fourth time in Slovakia. As usual, the main target group consisted of young people whose age corresponded to the age of the secondary school students and university students. This year, for the first time, the testing included a special test for primary school pupils. In this paper we communicate the test analysis as well as the results of the research conducted on the information part of the testing. We also focus on successful and unsuccessful topics of the testing and compare the results of students from various school types.

## Kľúčové slová

IT Fitness test, digitálne technológie, testovanie, IT zručnosti

## Keywords

IT Fitness test, digital technologies, testing, IT skills

## 1. ÚVOD

Rastúci dopyt po odborníkoch z oblasti digitálnych technológií, ktorých je aj napriek pomerne vysokej miere nezamestnanosti v Európe stále nedostatok, ako aj nízke povedomie o potrebe zlepšenia zručností a vedomostí v oblasti digitálnych technológií kvôli lepšej šanci zamestnať sa, boli v roku 2010 podnetom pre rozsiahlu kampaň Európskej komisie pod názvom *eSkills week*. Jej cieľom bolo motivovať mladých ľudí, najmä študentov stredných škôl a univerzít študovať alebo pracovať v oblasti digitálnych technológií a tiež podnietiť záujem širokej verejnosti v zvyšovaní úrovne vlastných vedomostí z zručností v tejto oblasti. Na kampani rôznymi aktivitami participovalo viac ako 20 krajín z Európy, medzi nimi aj Slovensko. Hlavnou aktivitou u nás bol IT Fitness test [2], do ktorého sa vtedy zapojilo viac ako 55 tisíc respondentov.

Po úspechu tejto kampane sa Európska komisia rozhodla zopakovať ju v roku 2012 a taktiež v roku 2014, kde sa už niesla pod heslom *eSkills for jobs*. Po skúsenostiach s obrovskou zapojenosťou mladých ľudí do prvého IT Fitness testovania, a taktiež vzhľadom k veľmi pozitívnemu hodnoteniu tejto aktivity a jej výstupov v správe Európskej komisie [1] sme zorganizovali IT Fitness test aj počas nasledujúcich eSkills kampaní [3,4,5].

Okrem toho v rámci národnej iniciatívy prebiehalo toto testovanie aj v roku 2013. Keďže aj iné krajiny (napr. Česká republika, Poľsko, Rusko) prejavili záujem o test, od roku 2012 sa náš test prekladá do angličtiny a češtiny.

Aj napriek tomu, že cieľovou skupinou testovania boli mladí ľudia vekovo zodpovedajúci študentom stredných a vysokých škôl, vždy sa doňho zapájali aj iné vekové kategórie. Preto sme od roku 2014 zaviedli pri testovaní aj ďalšiu cieľovú skupinu – žiakov základných škôl, pre ktorých bol pripravený samostatný test.

Vo výskumoch, ktorých čiastkové výstupy uvádzame v tomto článku, sme sa zamerali na preskúmanie výsledkov základnej cieľovej skupiny pri riešení testu určenému najmä študentom stredných a vysokých škôl.

## 2. IT FITNESS TEST PRE ŠTUDENTOV

### 2.1 Cieľ testovania

Tak ako aj v predchádzajúcich rokoch bolo cieľom testovania zistiť úroveň základných i pokročilejších znalostí a zručností účastníkov v oblasti digitálnych technológií. Základnú cieľovú skupinu tvorili mladí ľudia medzi 15 a 30, prevažne študenti stredných a vysokých škôl.

### 2.2 Organizácia testovania

IT Fitness test 2014 bol organizovaný pod záštitou ministerstva školstva a bola mu venovaná pomerne veľká publicita. Testovanie prebiehalo on-line na portáli [www.itfitness.sk](http://www.itfitness.sk) a bolo prístupné od polovice mája do konca júla 2014.

Celá testovacia aktivita bola rozdelená do troch častí:

*Profil* – respondenti tu vyplňali základné osobné údaje. V rámci testovania boli vyberaní najúspešnejší riešitelia vo viacerých kategóriách. Údaje uvedené v profilovej časti sa využívali na identifikáciu víťazov.

*Informačná časť* – v nej respondenti odpovedali na niekoľko otázok súvisiacich s ich využívaním digitálnych technológií.

*Testovacia časť* – samotné testovanie znalostí a zručností v oblasti vyhľadávania, vytvárania, spracovania i prezentovania informácií prostredníctvom digitálnych technológií.

Test pozostával z 24 uzavretých otázok s viacerými možnosťami odpovede, z ktorých len jediná bola správna. Test pre každého respondenta sa generoval zvlášť, náhodným výberom každej otázky spomedzi jej 4 vopred pripravených variantov.

Počas vyplňania testu sa každému respondentovi priebežne vyhodnocovali jeho odpovede a po ukončení testu sa mu jeho výsledky okamžite zobrazili formou percentuálnej úspešnosti. Okrem toho bol zaradený do jednej z piatich úrovní znalostí



a zručností v oblasti DT, ku ktorej dostal aj slovnú charakteristiku.

## 2.3 Charakteristika testu a pokryté oblasti

S ohľadom na cieľovú skupinu bol test vytvorený na základe platných stredoškolských štandardov pre informatiku, niektorých iných štandardov – napr. ECDL, ako aj na základe našich skúseností z predchádzajúcich testovaní.

Test pokrýval nasledujúce oblasti:

**A. Základy.** Otázky zaradené do tejto oblasti testovali základné znalosti o používaní počítačov a základného softvéru.

**B. Formáty dát.** Táto oblasť testovala znalosti o dátových formátoch vhodných na konkrétny zadaný účel, ich charakteristiky a pod.

**C. Hardvér.** V otázkach z tejto oblasti sme sa pýtali respondentov na počítače, ich komponenty, vstupné a výstupné zariadenia, mobilné zariadenia a pod.

**D. Softvér.** Táto oblasť testovala znalosti respondentov o typoch softvéru a ich zručnosti pri výbere a použití vhodného softvéru na riešenie daného problému.

**E. Kancelársky softvér.** Otázky pokrývali znalosti z oblasti práce s textovým procesorom a tabuľkovým kalkulátorom.

**F. Internet.** Táto oblasť zahŕňala prevažne praktické úlohy testujúce zručnosti a znalosti respondenta o internete ako informačnom zdroji a komunikačnom médiu. Aby respondenti mohli zodpovedať otázky, museli najprv nájsť na internete relevantné informácie.

**G. Sociálne, právne a bezpečnostné aspekty.** Tieto otázky pokrývali oblasti ako podvodné e-maily, netiketa, softvérové licencie, copyright, informačná bezpečnosť a pod.

Niektoré z otázok pokrývali aj viacero z uvedených oblastí a nedajú sa jednoznačne zaradiť len do jednej z nich.

## 3. VÝSLEDKY

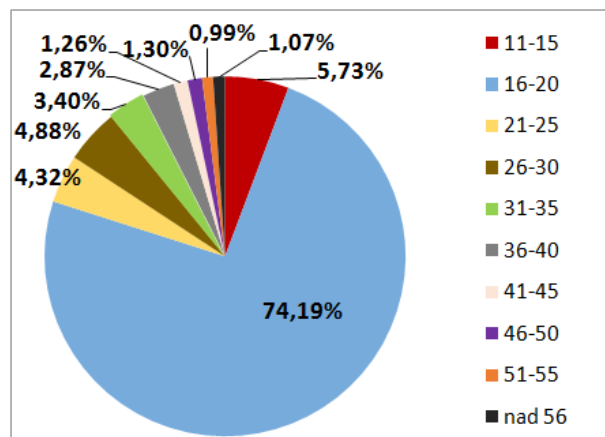
### 3.1 Demografické údaje

Z celého korpusu dát získaných od všetkých respondentov počas testovania sme odfiltrovali neúplné a nezmyselné dáta, čím sme získali údaje celkovo od 14 540 respondentov, z čoho bolo 10 585 takých, ktorí vyplnili aspoň jeden z testov. Z týchto dát sme odseparovali dáta riešiteľov testu pre základnú školu a zostali nám dáta od 6 091 riešiteľov testu pre študentov.

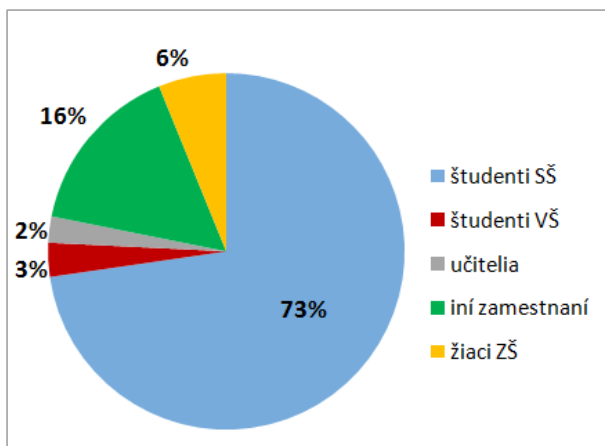
Keďže test bol verejne prístupný a otestovať sa mohol ktokoľvek, získané dáta zahŕňali nielen hlavnú cieľovú skupinu (vek 15-30 rokov), ale aj iné vekové kategórie. Ako vidno z nasledujúceho grafu (Obrázok 1), najsilnejšiu vekovú kategóriu tvorili respondenti vo veku 16-20 rokov, čo vekovo zodpovedá študentom stredných škôl, prípadne čerstvým absolventom stredných škôl. Do vekovej kategórie zodpovedajúcej hlavnej cieľovej skupine patrilo viac ako 83 % všetkých testovaných.

Respondentov testu sme kategorizovali aj podľa zamestnania. Hlavné kategórie zamestnania, ktoré sme sledovali, boli: žiaci základných škôl, študenti stredných a vysokých škôl a učitelia. Respondenti, ktorí nespádali do žiadnej z týchto kategórií boli označení ako 'ostatní'. Zastúpenie jednotlivých kategórií

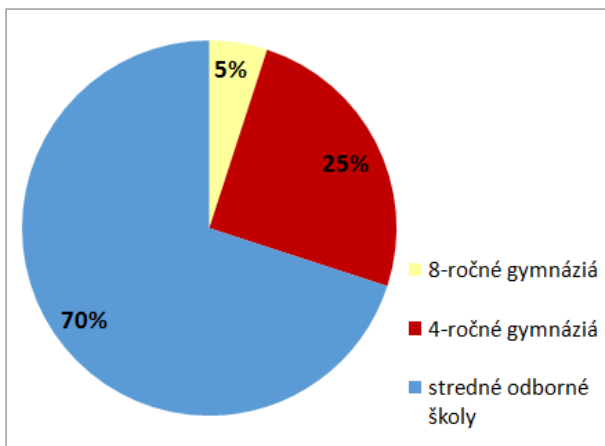
zamestnania je zachytené na grafe na obrázku 2. Študenti stredných škôl a vysokých škôl tvorili približne 76% zo všetkých respondentov.



Obrázok 1. Respondenti testu pre študentov podľa veku



Obrázok 2. Respondenti testu podľa zamestnania



Obrázok 3. Respondenti podľa typu strednej školy

V našom výskume sme sa zaoberali aj porovnaním výsledkov študentov z jednotlivých typov stredných škôl [4]. Percentuálne zastúpenie respondentov zo stredných škôl vzhľadom k typu školy ukazuje graf na obrázku 3.

Testu pre študentov sa zúčastnilo necelých 74% respondentov mužského pohlavia a len asi 26% respondentov ženského pohlavia. Tento výsledok spolu s výsledkami z predchádzajúcich rokov poukazuje na trend klesajúcej účasti respondentiek-žien v testovaní.

### 3.2 Výsledky informačnej časti

V tejto časti uvádzame výsledky otázok z informačnej časti testu relevantných pre danú cieľovú skupinu.

Zodpovedanie všetkých otázok informačnej časti nebolo striktné povinné. Pri vyjadrovaní mnohých odpovedí preto percentuálne prepočítavame počty jednotlivých alternatív vzhľadom k počtu respondentov, ktorí na danú otázku odpovedali. V prípadoch, keď je to vhodné vyjadrujeme výsledky s ohľadom na všetkých respondentov testu.

**Tabuľka 1. Používanie počítača od predškolského veku**

	študenti SŠ	študenti VŠ
áno	37,71%	55,47%
nie	62,29%	44,53%

Na otázku, či **počítač využívajú už od predškolského veku**, neodpovedalo až 43,61 % stredoškôľakov a 55,52 % vysokoškôľakov (Tabuľka 1). Z tých, čo odpovedali, sa kladne vyjadrilo prekvapivo značne vyššie percento študentov VŠ (55,47 %) ako študentov SŠ (37,71 %).

**Tabuľka 2. Miesto používania počítača**

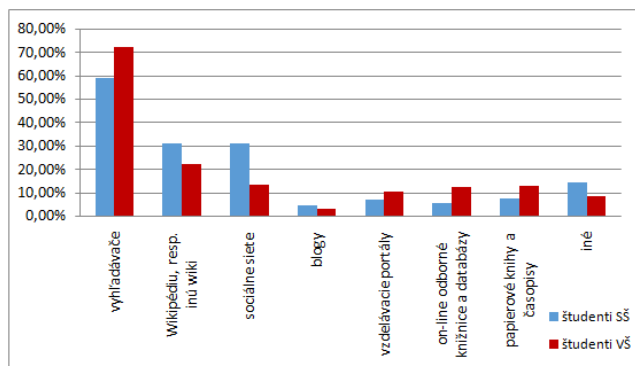
	študenti SŠ	študenti VŠ
v škole/ zamestnaní	70,58%	89,94%
doma/ na internáte	93,47%	94,81%
na verejných miestach (knížnica, internetová kaviareň a pod.)	19,18%	44,16%

Zaujímalo nás aj **miesto, kde študenti počítač používajú** (Tabuľka 2). Percentuálny podiel respondentov používajúcich počítač doma je v oboch skupinách študentov vysoký a takmer rovnaký. Prekvapujúci je však takmer 20 % rozdiel v počte študentov, ktorí používajú počítač v škole v prospech študentov VŠ.

**Tabuľka 3. Využitie informácií z internetu v škole**

	študenti SŠ	študenti VŠ
veľmi často	62,35%	74,03%
často	36,35%	25,97%
málokedy	0,00%	0,00%
vôbec nevyužívam	1,30%	0,00%

Ďalšia otázka bola zameraná na využívanie informácií z internetu v škole (Tabuľka 3). Porovnanie odpovedí študentov a všetkých respondentov ukazuje, že študenti využívajú v škole informácie získané na internete viac, ako ich na pracovné, resp. študijné účely využíva priemerný respondent. Z tých, ktorí odpovedali na otázku, len 1,3 % študentov SŠ odpovedalo „vôbec nevyužívam“, študenti VŠ využívajú na 100 %, z toho 74 % veľmi často.



**Obrázok 4. Najčastejšie využívané informačné zdroje**

Pokiaľ ide o **najpreferovanejšie zdroje informácií** (Obrázok 4), starší študenti oproti mladším vo väčšej miere samostatne vyhľadávajú informácie pomocou vyhľadávačov. Na druhej strane, vysokoškôľáci využívajú ako svoj informačný zdroj sociálne siete v menej ako polovičnej miere oproti stredoškôľakom (13,77 % ku 31,55 %).

**Tabuľka 4. Sebahodnotenie študentov**

	študenti SŠ		študenti VŠ	
expert v oblasti IT	0,00%	0,00%	0,00%	0,00%
pokročilý používateľ IT	44,79%	24,67%	52,73%	28,25%
bežný používateľ IT	37,67%	20,75%	45,45%	24,35%
začínajúci používateľ IT	17,54%	9,66%	1,82%	0,97%
neodpovedalo		44,92%		46,43%

Výsledky prieskumu **sebahodnotenia** nezodpovedajú výsledkom testu. V priemere viac ako 25 % všetkých študentov (a viac, ako 45 % tých, ktorí odpovedali na otázku), sa pokladá za pokročilého používateľa IT, zatiaľ čo podľa výsledkov testu tejto kategórii zodpovedá len okolo 17 % respondentov (ak za pokročilých považujeme tých, ktorí správne zodpovedali aspoň 2/3 otázok).

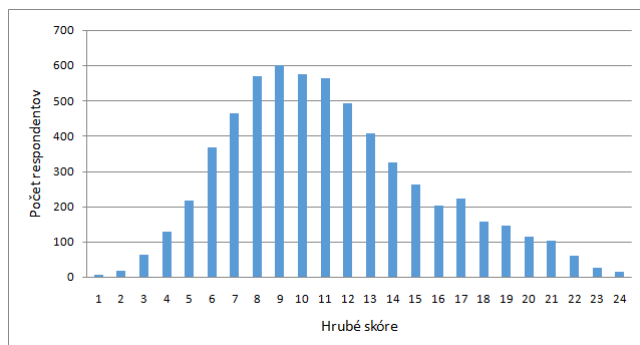
### 3.3 Analýza testu

Celková priemerná úspešnosť testu bola 46,57 %. Výsledok študentov bol ešte nižší, dosiahli priemernú úspešnosť 42,83 %. Pozrime sa podrobnejšie na niektoré parametre testu.

#### 3.3.1 Hrubé skóre.

Hrubé skóre vyjadruje, koľko respondentov dosiahlo v teste jednotlivé počty bodov.

Ako vidno z grafu na obrázku 5, rozloženie hrubých skóre testu pre SŠ dosť dobre odpovedá Gaussovej krivke, jej maximum je však mierne posunuté vľavo. Vypovedá to o skutočnosti, že test bol pre respondentov trochu obťažnejší. Toto zistenie korešponduje aj s priemerným hrubým skóre 11,18 a priemernou úspešnosťou testu 46,57 %.



Obrázok 5. Rozdelenie hrubých skóre testu

### 3.3.2 Úspešnosť jednotlivých úloh

Najvyššiu priemernú úspešnosť dosiahli úlohy z oblasti bezpečnosti (76,9 %), mobilných zariadení (70,2 %) a vyhľadávania na internete (kurzová kalkulačka – 63,6 %, texty – 60,4 % a GPS a mapy – 60,1 %) (Obrázok 6).

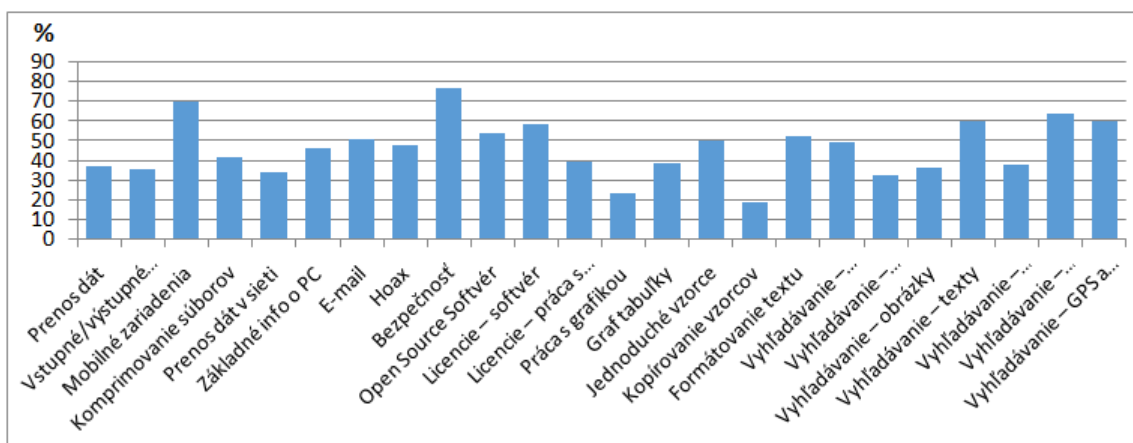
Najnižšiu priemernú úspešnosť mali úlohy z oblasti tabuľkového kalkulatora (kopírovanie vzorcov – 18,9 %), práce s grafikou

(23,5 %) a trochu paradoxne aj z oblasti vyhľadávania na internete (údaje o konkrétnej konferencii).

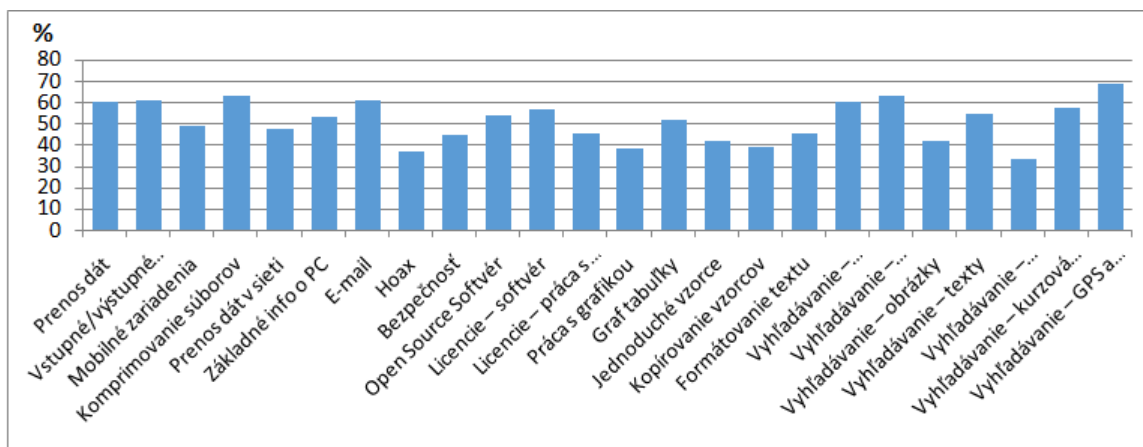
### 3.3.3 Citlivosť úloh

Dôležitým parametrom testových úloh je ich citlivosť, teda to, ako dobre vedia odlíšiť respondentov s dobrou úrovňou vedomostí, resp. zručností od slabých respondentov. Citlivosť úlohy sa počíta ako rozdiel priemernej percentuálnej úspešnosti najúspešnejšej pätiny testovaných a najmenej úspešnej pätiny testovaných. Ak má mať test dobrú rozlišovaciu schopnosť, mal by obsahovať čo najviac úloh s vysokou citlivosťou. Za dobrú citlivosť úlohy považujeme citlivosť nad 30 %.

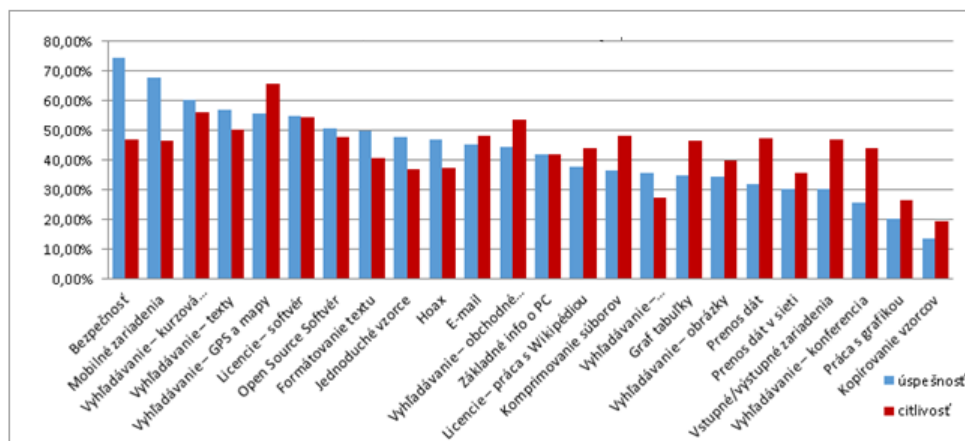
Ako možno vidieť na grafe z obrázka 7, všetky úlohy v teste mali dobrú citlivosť. Najslabšiu citlivosť (32,9 %) preukázala úloha o vyhľadávaní na portáli Ministerstva školstva, vedy, výskumu a športu SR, ktorej priemerná úspešnosť bola 37,8 %, a ktorú ani najúspešnejších 20 % respondentov nevyriešilo lepšie, ako s priemernou úspešnosťou 56,6 %. Podľa tohto parametra teda náš test bol schopný rozlíšiť dobrých a slabých respondentov.



Obrázok 6. Úspešnosť jednotlivých otázok



Obrázok 7. Citlivosť jednotlivých úloh



Obrázok 8. Úspešnosť a citlivosť úloh vzhľadom k výsledkom študentov (usporiadané podľa úspešnosti)

### 3.3.4 Výsledky študentov vs. výsledky ostatných

V porovnaní s ostatnými kategóriami riešiteľov testu študenti dosiahli najslabší výsledok – 42,83 % (Tabuľka 5). Najlepšie dopadli učitelia, ktorých úspešnosť bola oproti študentom viac ako 1,5-násobne vyššia.

Tabuľka 5. Priemerné výsledky študentov, učiteľov a ostatných respondentov

	študenti	učitelia	ostatní
priemerné hrubé skóre	10,28	15,93	15,22
celková priemerná úspešnosť	42,83%	66,38%	63,40%

### 3.3.5 Podrobnejšia analýza výsledkov študentov

V našom výskume sme sa venovali aj podrobnejšej analýze výsledkov študentov. Vyhodnotili sme pre nich nielen úspešnosť a citlivosť jednotlivých úloh testu (Obrázok 8.), ale aj úspešnosť a citlivosť jednotlivých variantov úloh. Analýza z tohto pohľadu ukázala, že niekoľko variantov úloh malo citlivosť nižšiu ako 30 %.

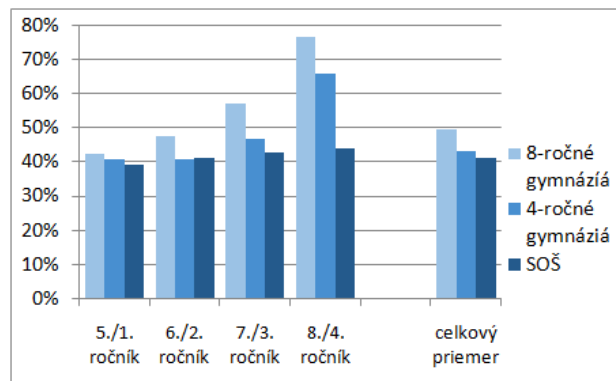
Medzi 5 najmenej úspešných úloh patrili všetky 4 varianty úlohy z oblasti "Tabuľkový kalkulator – kopírovanie vzorcov". Všetky štyri varianty mali navyše veľmi nízku citlivosť, čo svedčí o tom, že táto úloha bola pre študentov mimoriadne ťažká (aj najúspešnejších 20 % študentov v nej dosiahlo priemernú úspešnosť pod 29 %). Úlohy o tabuľkovom kalkulátore už tradične patria medzi úlohy IT Fitness testu s najnižšou úspešnosťou, a to nielen u študentov, ale celkovo. Toto sa potvrdilo aj tento rok. Avšak aj v tejto oblasti sa našli úlohy, ktoré aj študenti zvládli výrazne lepšie, napr. úloha o zostavení jednoduchého vzorca dosiahla nadpriemerné skóre. Veríme, že to môže byť aj dôsledok toho, že sa aj tejto oblasti v poslednom čase venuje v školách vyššia pozornosť.

V neúspešnej úlohe z oblasti grafiky, ktorá preverovala tak teoretické vedomosti ako aj praktické skúsenosti, bolo treba navyše správne vyhodnotiť tri tvrdenia a z nich skombinovať správnu odpoveď. Tento trochu náročnejší postup zrejme tiež prispel k slabšiemu výsledku. O náročnosti úlohy vypovedá aj jej nižšia citlivosť.

Medzi prvými siedmimi úlohami s najvyššou úspešnosťou sa nachádzali po tri varianty z dvoch úloh (9 – Bezpečnosť a 3 – Mobilné zariadenie). Až na jediný variant, ktorého veľmi nízka citlivosť (27,1 %) a veľmi vysoká úspešnosť (91,3 %) zrejme svedčia o veľmi nízkej náročnosti tejto konkrétnej úlohy, všetky ostatné úspešné úlohy mali veľmi dobrú citlivosť a mali by teda dobre rozlišovať medzi dobrými a slabými študentmi.

### 3.3.6 Porovnanie výsledkov študentov z rôznych typov stredných škôl

Časť nášho výskumu bola zameraná na porovnanie výsledkov študentov podľa typu a stupňa školy, ktorú navštevujú. Viac podrobností z tejto oblasti, ako aj vyhodnotenie výsledkov podľa iných kritérií možno nájsť v našom článku z konferencie ICETA 2014 [4]. Tu uvedieme len porovnanie výsledkov študentov stredných odborných škôl, 4-ročných a 8-ročných gymnázií (Obrázok 9).



Obrázok 9. Celková úspešnosť študentov jednotlivých typov stredných škôl podľa príslušných ročníkov

Študenti 8-ročných gymnázií dosiahli najlepší celkový výsledok (49,4%), študenti 4-ročných gymnázií boli približne o 6,5 % slabší a študent stredných odborných škôl boli v testovaní najmenej úspešní (41,1 %). Pokiaľ ide o výsledky v jednotlivých ročníkoch, študenti 8-ročných gymnázií ukázali, že majú lepšie vedomosti a zručnosti v oblasti digitálnych technológií ako študenti v zodpovedajúcom ročníku iných stredných škôl. Tieto

výsledky potvrdili naše zistenia z predchádzajúceho testovania v roku 2012 a dokonca ukázali aj nový fakt - že pozitívny rozdiel medzi úspešnosťou študentov 8-ročných gymnázií a ostatných stredných škôl dokonca so zvyšujúcim sa ročníkom študentov narastá. Toto môže poukazovať na vyššiu efektívnosť vzdelávania a rozvoja zručností a vedomostí z oblasti digitálnych technológií na 8-ročných gymnáziách.

#### 4. Záver

Výsledky posledného IT Fitness testovania boli síce slabšie, ako sme očakávali podľa výsledkov v predchádzajúcich rokoch, toto však nemusí nutne znamenať pokles vo vedomostiach a zručnostiach respondentov. Nie je totiž možné dosť dobre porovnať výsledky dvoch rozdielnych skupín respondentov vyplňajúcich dva rôzne testy. Taktiež treba upozorniť na to, že cieľom autorov testu nebolo vytvoriť test s veľmi vysokou úspešnosťou, ale test, ktorý bude vedieť odlišiť respondentov s dobrými vedomosťami a zručnosťami od respondentov so slabšími vedomosťami a zručnosťami, čo sa podľa nás podarilo.

V našom výskume sme sa zaoberali o.i. aj vývojom zručností a vedomostí v oblasti digitálnych technológií u študentov stredných a vysokých škôl. Výsledky výskumu ukazujú postupný vývoj týchto zručností a vedomostí u stredoškôľakov, avšak u študentov stredných odborných škôl je progres extrémne pomalý v porovnaní so študentmi gymnázií a najmä študentmi 8-ročných gymnázií. Títo vykazujú dokonca vyššiu úspešnosť, ako študenti prvého ročníka vysokých škôl. Zdá sa teda, že spôsob, akým v oblasti digitálnych technológií rozvíjajú svojich študentov 8-ročné gymnáziá, je značne efektívnejší ako na ostatných typoch stredných škôl.

Výsledky nášho testu poukázali na najproblematickejšie oblasti, no taktiež na to, že pre študentov medzi najťažšie úlohy patrili také úlohy, pri ktorých bolo treba použiť istú mieru abstrakcie, skombinovať teoretické poznatky so skúsenosťami nadobudnutými v praxi, prípadne úlohy, pri riešení ktorých bolo treba použiť zložitejšie operácie (napr. niektoré komplikovanejšie úlohy zamerané na vyhľadávanie na internete). Práve z tohto dôvodu by bolo potrebné odporučiť učiteľom, aby prostredníctvom vhodne zvolených vzdelávacích aktivít, komplexnejších úloh a projektov viedli študentov

k prepájaniu poznatkov z viacerých oblastí učiva, k využívaniu tak teoretických vedomostí ako aj praktických zručností, ako aj k zapájaniu vyšších kognitívnych operácií, napr. analyzovať, hodnotiť, tvoriť. Ako potvrdzujú napr. aj výsledky informačnej časti nášho testovania, rozvíjať zručnosti a schopnosti v oblasti digitálnych technológií je dôležité a je potrebné s tým začať podľa možnosti čím skôr.

#### 5. POĎAKOVANIE

Príspevok je súčasťou výskumu v projekte VEGA 1/0948/13.

#### 6. BIBLIOGRAFICKÉ ODKAZY

- [1] European Commission: European e-Skills week 2010: Key facts & figures.  
[http://files.eun.org/eskills/facts\\_figures\\_eSkillsWeek.pdf](http://files.eun.org/eskills/facts_figures_eSkillsWeek.pdf)
- [2] KUBINCOVÁ, Z., HOMOLA, M., KUBINEC, P., HRUŠECKÁ, A., JAKAB, F., BRESTENSKÁ, B., *IT Skills in Slovak schools: Results of the eSkills IT Fitness test 2010*, in ICETA 2010: 8th International Conference on Emerging eLearning Technologies & Applications, 2010 IEEE Košice: elfa, 2010. - pp.339-344, ISBN 978-80-8086-166-7.
- [3] KUBINCOVÁ, Z., HOMOLA, M., KUBINEC, P., VARGA, M., HRUŠECKÁ, A., BLAHO, A., JAKAB, F., BRESTENSKÁ, B., *eSkills IT Fitness Test 2012 in Slovakia*, in ICETA 2012: 10th International Conference on Emerging eLearning Technologies & Applications, 2012 IEEE.
- [4] KUBINCOVÁ, Z., JAKAB, F. *eSkills eSkills for Jobs 2014 in Slovakia: IT Fitness Test for students*, in ICETA 2014: 12th International Conference on Emerging eLearning Technologies & Applications, 2014 IEEE
- [5] *IT Fitness Test 2014*, Správa o výsledkoch IT Fitness Testovania 2014, IT Asociácia Slovenska, Bratislava 2014.

# Metodika výučby programovania mobilných zariadení

Ľubomír Šnajder

UPJŠ v Košiciach, Prírodovedecká

fakulta, Ústav informatiky

Jesenná 5

041 54 Košice

Slovenská republika

lubomir.snajder@upjs.sk

## ABSTRAKT

Vďaka projektom Digiškola a Škola na dotyk sa do našich škôl dostáva čoraz viac dotykových zariadení s množstvom edukačných aplikácií. Tieto technológie využíva vo výučbe aj mimo nej narastajúca sa komunita učiteľov a ich žiakov. Od roku 2012 sme sa začali venovať výučbe programovania mobilných zariadení so zameraním sa na talentovaných žiakov základnej školy. V príspevku popisujeme vývoj metodík výučby programovania mobilných aplikácií v prostredí App Inventor 2, ktoré vytvárame a overujeme v spolupráci s gymnáziami zapojenými do projektu APVV-0715-12 (Výskum efektívnosti metód inovácie výučby matematiky, fyziky a informatiky). Ďalej prezentujeme vytvorené metodiky výučby spolu s pracovnými listami a ukážkami žiackych projektov. V závere príspevku uvádzame ako prezentujeme tieto metodiky učiteľskej verejnosti v rámci stretnutí Klubov učiteľov informatiky a tiež ich integráciu do vysokoškolskej prípravy budúcich učiteľov informatiky v predmetoch Školské programovacie prostredia 2 a Didaktika informatiky.

## ABSTRACT

Thanks to the projects “Digiškola” (Digital School) and “Škola na dotyk” (School to the touch) our schools are getting more and more touch devices with a number of educational applications. These technologies are used in teaching and beyond it by increasing community of teachers and their pupils. Since 2012, we have started teaching of programming mobile devices focusing on the talented pupils in elementary schools. This paper describes the development of methodologies for teaching programming mobile applications in the App Inventor 2 which we develop and test in cooperation with the grammar schools involved in the project APVV-0715-12 (Research on the effectiveness of innovative teaching methods in mathematics, physics and computer science). Further we present the proposed methodologies of teaching together with worksheets and examples of pupils’ projects. In the conclusion of the paper we present the following contribution to the methodology of teaching in the context of the public meetings of the Clubs of Teachers of Computer Science and integrate them into the University training of future teachers of computer science in the subjects Educational Programming Environments 2 and Didactics of Computer Science.

## Kľúčové slová

metodika výučby, programovanie, mobilné zariadenia, App Inventor 2, príprava učiteľov informatiky.

## Keywords

Teaching methodology, programming, mobile devices, App Inventor 2, pre-service teachers training.

## ÚVOD

Projekty *Elektronizácia vzdelávacieho systému regionálneho školstva (Digiškola)* [1] a *Škola na dotyk* [2] významne napomáhajú šíreniu dotykových technológií do našich škôl. Popri zabezpečení nevyhnutnej technologickej infraštruktúry (tabletov, interaktívnych tabúl, smerovačov, notebookov, farebných tlačiarň), poskytujú školám digitálny vzdelávací obsah pre vybrané predmety a softvérové nástroje pre podporu a riadenie výučby. Veľmi dôležité v týchto projektoch je vzdelávanie učiteľov formou metodických seminárov a konferencií. Vďaka týmto projektom vznikla a postupne sa rozširuje komunita otvorených a aktívnych učiteľov, ktorí vytvárajú kolekcie on-line vzdelávacích materiálov (tzv. tuuly), ktoré sú k dispozícii aj pre ostatných učiteľov.

Dotykové technológie, obzvlášť mobilné zariadenia ako tablety a smartfóny, majú navyše potenciál vytvárať užitočné aplikácie samotnými žiakmi, ktorí sa týmto zároveň učia programovať a riešiť algoritmické problémy. Okrem textových programovacích jazykov (napr. Java, Python), sú dostupné aj alternatívy vhodné pre žiakov základných, prípadne stredných škôl. Pre programovanie hier založených na polohe, konkrétne pre softvérovú platformu Wherigo, sa používa vizuálne vývojové prostredie Urwigo, ktoré popisujú v svojom príspevku [3] Gabriela Lovászová a Viera Palmarová. Wolfgang Slaný v [4] predstavuje pomerne jednoduché prostredie Catrobat Pocket Code, umožňujúce vytvárať kód skladaním blokov priamo v mobilnom zariadení.

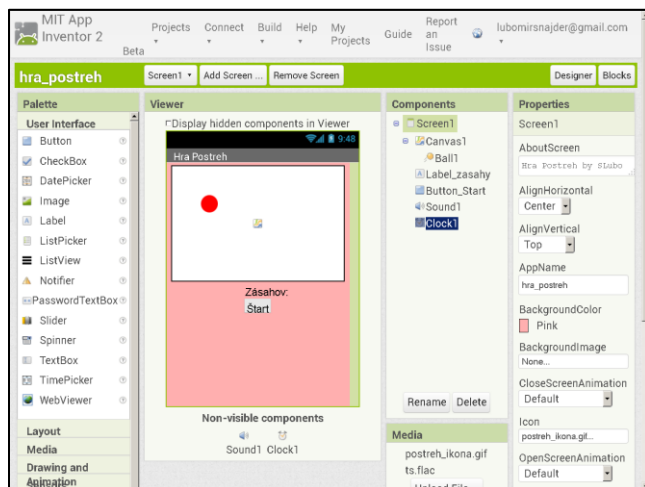
Veľkú popularitu medzi študenti a žiakmi rôznych vekových kategórií získava cloudové programovacie prostredie MIT App Inventor 2 (ďalej AI2), umožňujúce intuitívnym spôsobom programovať mobilné aplikácie určené pre otvorenú platformu Android. Podľa [5] bolo k októbru 2014 registrovaných na cloude AI2 1,9 milióna používateľov zo 194 krajín, ktorí vytvorili, prípadne začali vytvárať spolu 4,7 milióna aplikácií. AI2 používajú v niektorých amerických vysokých školách v úvodných kurzoch programovania pre neinformatické odbory po vzore Davida Wolbera z University of San Francisco, ktorý vyučuje AI2 od začiatku jeho vývoja (od roku 2009) a je tvorcom metodických materiálov The App Inventor Course-in-a-Box dostupných na webe [6] a spoluautorom významnej publikácie o AI2 [7].

U nás sa metodikou výučby programovania v prostredí v AI2 zaoberá Monika Tomcsányová, ktorá na svojich stránkach [8] prezentuje viaceré zaujímavé projekty a metodické materiály k programovaniu vybraných projektov v AI2. Postupne sa objavujú základné a stredné školy, ktoré sa venujú programovaniu v AI2, napr. na ZŠ Benkova v Nitre [9], na Gymnáziu Alejová a Gymnáziu Poštová v Košiciach.



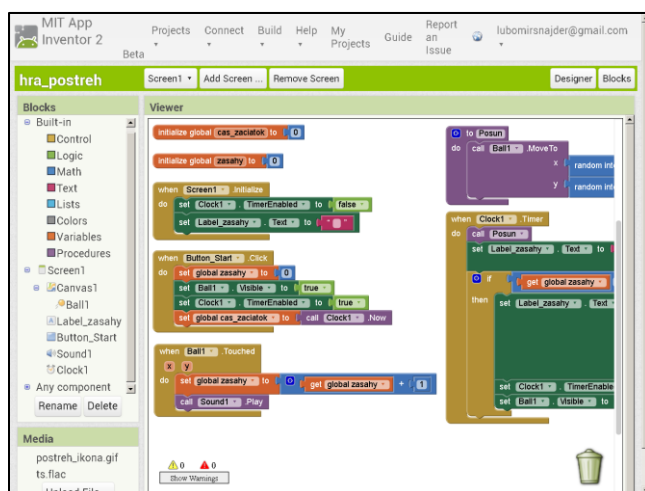
## 1. VÝVOJOVÉ PROSTREDIE AI2

Na vytváranie mobilných aplikácií v AI2 potrebujeme počítač so štandardným operačným systémom (Windows, Ubuntu, Debian, Mac OS) s webovým prehliadačom. Do cloudového vývojového prostredia AI2 [10] sa prihlasujeme prostredníctvom Google účtu. Prostredie AI2 umožňuje vytvárať a spravovať zdrojové súbory (\*.aia), kompilovať a zabiť ich do cieľového distribučného formátu (\*.apk). Používateľské rozhranie aplikácie, navrhujeme v module *Designer*. Na Obrázku 1 je uvedený rozhranie aplikácie obsahujúce viditeľné komponenty – *Canvas*, *Ball*, *Label a Button* a neviditeľné komponenty – *Clock*, *Sound*.



**Obrázok 1.** Modul Designer pozostáva z: Palety komponentov, Vzhľadu aplikácie, Hierarchie použitých komponentov, Zoznamu multimediálnych súborov a Vlastností aktuálneho komponentu.

Programový kód aplikácie vytvárame v module *Blocks*. Na Obrázku 2 sú uvedené bloky ovládačov udalostí (*Button.Click*, *Screen.Initialize*, *Clock.Timer*, *Ball.Touched*), blok vlastnej procedúry a inicializačné bloky globálnych premenných.



**Obrázok 2.** V module Blocks vytvárame program ťahaním blokov príkazov z ľavej časti obrazovky do plochy a ich skladaním vedľa seba, resp. do seba ako dieliky stavebnice.

Nakoniec program necháme skompilovať a vytvoriť jeho distribučný balíček (\*.apk), ktorý je možné a uložiť na cloude alebo na počítači. Balíčkový súbor z cloudu stiahneme do mobilného zariadenia oskenovaním QR-kódu s webovou

adresou jeho umiestnenia na cloude. Pre inštaláciu balíčkového súboru na mobilné zariadenie potrebujeme na ňom nastaviť povolenie inštalovať tzv. *non-market applications*.

Ak máme počítač aj mobilné zariadenie pripojené na rovnakú wifi-sieť, odporúčame pri vývoji mobilnej aplikácie nadviazať živé spojenie medzi počítačom a mobilným zariadením, na ktorom môžeme testovať funkcionality napísaného kódu aplikácie. Pre uskutočnenie takého živého spojenia potrebujeme na mobilné zariadenie nainštalovať program *MIT AI2 Companion*. Ak nemáme poruke mobilné zariadenie, mobilnú aplikáciu môžeme testovať aj pomocou emulátora – programu *aiStarter* spusteného na počítači. Túto možnosť odporúčame len v krajnom prípade, keď nemáme k dispozícii žiadne mobilné zariadenie.

Aplikácie naprogramované v AI2 môžu využiť nasledovné možnosti mobilného zariadenia:

- dotykovú obrazovku,
- prehrávanie a záznam zvuku/video/fotografií,
- senzor zrýchlenia,
- senzor orientácie,
- senzor polohy (GPS),
- rozpoznávanie reči,
- syntézu reči,
- bluetooth spojenie,
- NFC spojenie,
- skener čiarových kódov,
- prijímanie a odosielanie SMS správ,
- uskutočnenie telefonických hovorov,
- komunikáciu na Twitteri,
- ukladanie/načítanie údajov na/z mobilného zariadenia, resp. z webu,
- ovládanie robotických modelov Lego Mindstorms NXT,
- spustenie ďalších aplikácií, atď.

Práve vďaka týmto širokým možnostiam mobilného zariadenia a jednoduchosti vývojového prostredia je AI2 veľmi atraktívnym nástrojom na vývoj veľmi pútavých a užitočných aplikácií, a tiež pre výučbu programovania a základných princípov informatiky.

## 2. METODICKÝ PRÍSTUP VÝUČBY PROGRAMOVANIA V AI2

David Wolber na svojej webovej stránke The Course In a Box [6] odporúča pri výučbe programovania v AI2 nasledovný model:

*Zostaviť – Konceptualizovať – Prispôbiť – (Vytvoriť)*

Najprv študenti zostavia zaujímavú aplikáciu podľa podrobného tutoriálu. Po dokončení aplikácie pripraví učiteľ konceptuálne otázky týkajúce sa jednotlivých prvkov vytvorenej aplikácie, ktoré nechá študentom prediskutovať a vysvetliť. V ďalšej fáze študenti prispôbujú a rozširujú danú aplikáciu. Po dvoch-troch takýchto iteráciách *Zostaviť – Konceptualizovať – Prispôbiť* by mali byť študenti pripravení naprogramovať vlastnú mobilnú aplikáciu podľa svojich osobných záujmov.

Ďalšími užitočnými pomôckami pri štúdiu, resp. samoštúdiu, uvedenými na stránke [6], sú tutoriály a videotutoriály a sekcia "how do you ...?" s návodmi na riešenie vybraných problémov.

David Wolber odporúča, aby si študenti vytvárali na webových stránkach portfólio obsahujúce ich vlastné užitočné mobilné aplikácie, ktoré by mali ukázať svojim priateľom a rodine.

Pre publikovanie, zdieľanie a komentovanie projektov vytvorených v prostredí MIT App Inventor 1 zriadili tri americké univerzity (USF, Umass-Lowell, MIT) portál App Inventor Community Gallery [11], ktorý slúži hlavne začínajúcim vývojárom ako on-line štúdio podporujúce ich kolaboráciu a učenie sa programovania.

Náš prístup vyučovania programovania v AI2 je inšpirovaný uvedeným prístupom Davida Wolbera a je založený na nasledovných myšlienkach:

- výučba programovania v AI2 sa realizuje výlučne programovaním pútavých a užitočných aplikácií, ktoré sú špecifické pre mobilné zariadenia, tieto aplikácie tvoria portfólio študenta,
- pri tvorbe aplikácií sa najprv predstavia senzory až neskôr postupne jednotlivé programové konštrukcie (cykly, vetvenia, procedúry, atď.),
- študenti postupujú pri tvorbe aplikácie podľa pracovného listu obsahujúceho prvky formatívneho hodnotenia, pričom konceptualizujeme preberané poznatky už od začiatku vývoja aplikácie,
- učiteľom k výučbe programovania v AI2 sú poskytnuté metodické materiály k pracovným listom (s riešeniami úloh, diskusnými otázkami, námetmi na rozšírenia aplikácie, skúsenosťami z výučby), ktoré sa iteratívne vylepšujú na základe spätnej väzby od aktívnych učiteľov, vyplnených pracovných listov žiakov a vytvorených žiackych aplikácií.

### 3. VÝVOJ METODÍK VYUČBY PROGRAMOVANIA V AI2

Na jar 2012 sme začali zbierať prvé skúsenosti s výučbou programovania v prostredí MIT App Inventor 1 (Classic). Na informatickom krúžku sme naprogramovali spolu so žiakmi niekoľko aplikácií (Hello Purr, Paint Pot, Mole Mash) podľa tutoriálov dostupných na webe [12]. Prvú metodiku k programovaniu hry Postreh (inšpirovanej hrou Mole Mash) sme vytvorili na základe týchto skúseností z informatického krúžku a publikovali v [13]. Táto metodika obsahovala časti:

- Zameranie (časová dotácia, veková skupina, zoznam kognitívnych cieľov).
- Priebeh (popis priebehu výučby, skúsenosti z krúžku, odkazy na tutoriály, referenčné príručky, diskusné fóra k AI1).
- Pracovný list pre žiaka (zadanie úlohy, postupnosť čiastkových problémov doplnená krátkymi pomocnými informáciami o AI1, dotazníkové otázky k zaujímavosti a náročnosti tejto aktivity).
- Pomôcky (riešenie hry Postreh – grafické prostredie s popisom a parametrami komponentov, programový kód s komentárom k významu jednotlivých blokov príkazov).

Začiatkom roku 2014 sme prešli na súčasnú (stabilnejšiu, prívetivejšiu a výrazovo bohatšiu) verziu AI2, ktorú sme

využívali na informatickom krúžku aj v rámci jedného stretnutia v predmete Detské programovacie prostredia. V rámci projektu APVV-0715-12 Výskum efektívnosti metód inovácie výučby matematiky, fyziky a informatiky (ďalej VEMIV) sme vytvorili metodiky, pracovné listy a učebný text k dvom mobilným aplikáciám Kresliaci editor a Hra Postreh, ktoré overujeme s učiteľmi spolupracujúcich partnerských gymnázií. Aktuálne aj staršie verzie metodík a pracovných listov spolu s hodnotiacim formulárom sú umiestnené v Moodle kurze projektu VEMIV [14].

Štruktúra aktuálnych metodík je nasledovná:




- Základné informácie (autori, recenzenti, verzia, téma, cieľová skupina, trvanie, vstupné vedomosti a spôsobilosti, kognitívne a afektívne ciele, spôsobilosti vedeckej práce, didaktický problém, materiálne didaktické prostriedky, vyučovacie metódy a organizačné formy).
- Príprava na výučbu (potrebný hardvér, softvér, pracovné listy, učebné texty, internetové zdroje).
- Priebeh výučby (postupnosť zadání úloh, usmerňujúce otázky, nápovede a riešenia úloh).
- Hodnotenie výučby (metodický komentár k formatívne a sumatívne hodnoteniu žiakov).
- Skúsenosti z výučby (postrehy z vlastnej výučby).
- Alternatívy metodiky (alternatívne prístupy pri menšom, resp. väčšom časovom rozsahu výučby).

Aktuálne pracovné listy pozostávajú z viacerých úloh, každá z nich má nasledovnú štruktúru:

- Zadanie úlohy.
- Analýza a návrh riešenia úlohy.
- Postup riešenia úlohy, resp. nápovede k riešeniam úlohy.
- Zhodnotenie úlohy.

#### 3.1 Kresliaci editor

V rámci prvého stretnutia programovania mobilných aplikácií v AI2 by sme mali:

- zistiť predchádzajúce poznatky žiakov o mobilných zariadeniach (aké zariadenie majú a ktoré mobilné aplikácie na ňom používajú) a ich očakávania od programovania mobilných aplikácií,
- nechať žiakov zistiť parametre mobilného zariadenia (verziu OS, kapacity pamäte zariadenia a SD karty) a tiež prítomné senzory, napr. pomocou aplikácie *Sensor Box for Android* od Imoblife, Inc. 
- nechať žiakov overiť, či majú na mobilnom zariadení aplikáciu na čítanie čiarových kódov, ak nie, tak by mohli z Google play stiahnuť a nainštalovať, napr. *Barcode Scanner* od ZXing Team, 
- nechať žiakov overiť pripojenie zariadenia do wifi-siete a nainštalovať aplikáciu pre živý vývoj aplikácií *MIT AI2 Companion* od MIT Center for Mobile Learning, 
- poskytnúť žiakom základnú predstavu o mobilných zariadeniach pracujúcich pod OS Android, o prostredí AI2, o procese tvorby a inštalácie mobilnej aplikácie.

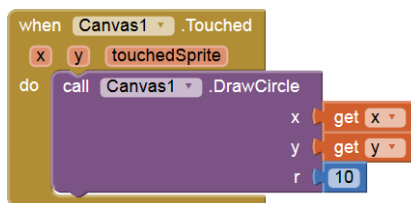


- nechať žiakov vytvoriť jednoduchú kresliacu aplikáciu v prostredí AI2 a inštalovať ju na mobilné zariadenie.

Najjednoduchšia verzia kresliaceho editora je taká, ktorá na každom mieste dotyku zobrazí na plátne kruh. Žiakov od začiatku vedieme, aby pri vývoji aplikácie najprv premysleli grafické rozhranie s viditeľnými a neviditeľnými komponentmi a potom premysleli správanie aplikácie a zapísali ho do tabuľky ako trojice *komponent – udalosť – odpoveď*.

**Tabuľka 1.** Správanie Kresliaceho editora 1 zapísané v tabuľke.

komponent	udalosť	odpoveď
plátno	dotyk	na plátno nakreslí kruh so stredom v mieste kliknutia a polomerom 10

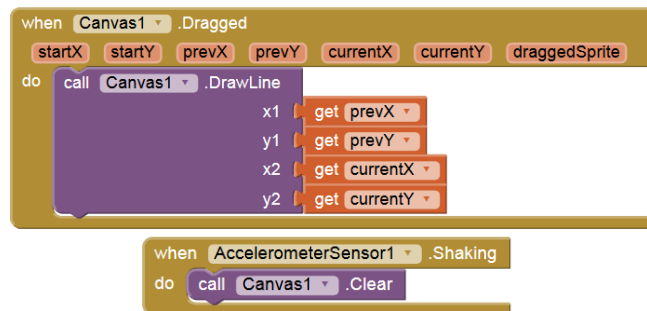


**Obrázok 3.** Programový kód Kresliaceho editora 1.

Mierne rozšírená verzia kresliaceho editora navyše umožňuje kresliť čiary pri ťahaní na plátno a zmazať plátno potrasením mobilného zariadenia.

**Tabuľka 2.** Správanie Kresliaceho editora 2 zapísané v tabuľke.

komponent	udalosť	odpoveď
plátno	dotyk	na plátno nakreslí kruh so stredom v mieste kliknutia a polomerom 10
plátno	ťahanie	na plátno nakreslí úsečku od predchádzajúcej do aktuálnej pozície
akcelerometer	trasenie	zmazanie plátna



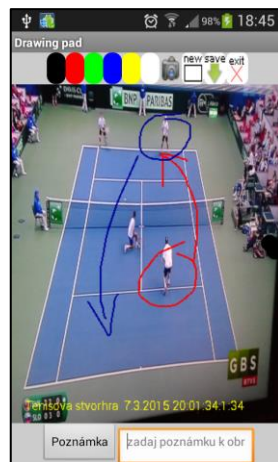
**Obrázok 4.** Doplnený programový kód Kresliaceho editora 2.

Napokon kresliaci editor môžeme doplniť o ďalšie funkcionality:

- zmena farby pera (pomocou tlačidiel pre 6 farieb),
- zmena farby celého plátna na bielu farbu,
- nastavenie obsahu plátna na aktuálnu fotografiu získanú vstavaným fotoaparátom,
- uloženie aktuálneho obsahu plátna do mobilného zariadenia a jeho opätovné načítanie pri ďalšom spustení aplikácie,

- doplnenie textovej poznámky s dátumom a časom,
- vyvolanie ukončenia aplikácie,
- vytvorenie a nastavenie vlastnej ikony aplikácie.

Na Obrázku 5 je zobrazená nainštalovaná aplikácia Kresliaci editor s uvedenými funkcionalitami.



**Obrázok 5.** Nainštalovaná aplikácia Kresliaci editor 3.

Uvedenú aplikáciu môžu využiť žiaci v role investigatívneho novinára, ktorý vstavaným fotoaparátom na smartfóne zaznamená dôležitú situáciu, ktorú okomentuje a opatrí časovou značkou.

Pri načítavaní fotografie uloženej v súbore je v starších verziách OS Android začiatok cesty `/mnt/sdcard/` a v novších verziách `/storage/emulated/0/`.

Miera rozšírenia funkcionalít kresliaceho editora je v rukách učiteľa, ktorý zváži, či ostane pri verzii 2 alebo nechá žiakom dokončiť verziu 3. V každom prípade po dokončení tejto aplikácie by mal učiteľ zopakovať a zhrnúť prebrané učivo, napr. formou otázok:

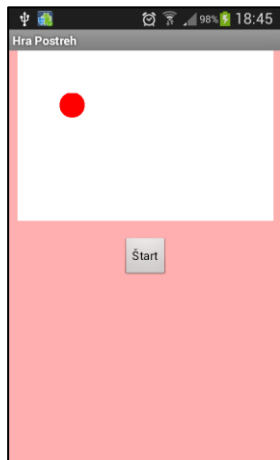
- Aký hardvér a softvér potrebujeme pre programovanie mobilnej aplikácie v cloud prostredí AI2?
- Ako budeme postupovať pri programovaní mobilnej aplikácie v cloud prostredí AI2?
- Aký je rozdiel medzi modulmi Designer a Blocks prostredia AI2?
- Čo obsahuje mobilná aplikácia vytváraná v prostredí AI2 z pohľadu udalostíami riadeného programovania?
- Aký je rozdiel medzi súborom s príponou AIA a príponou APK?
- Zosumarizujte, ktoré komponenty, udalosti, metódy, nastavenia ste použili v aplikácii Kresliaci editor.
- Aký význam má udalosť `Screen.Initialize`?
- Vysvetlite rozdiel medzi metódou `Camera.TakePicture` a udalosťou `Camera.AfterPicture`.
- Vysvetlite akým spôsobom ste zmenili farbu pozadia plátna na bielu.
- Vysvetlite akým spôsobom ste uložili/načítali pozadie plátna na/z mobilného zariadenia.

### 3.2 Hra Postreh

Zadanie základnej verzie hry môže vyzeráť nasledovne:

Vytvorte hru *Postreh*, ktorá bude v pravidelných časových intervaloch zobrazovať červenú loptu na náhodné miesta obrazovky. Úlohou hráča je čo najskôr sa dotknúť tejto lopty. Po dotyku lopty sa ozve krátky zvuk.

Aplikácia obsahuje viditeľné komponenty plátno a na ňom umiestnenú loptu a neviditeľné komponenty hodiny a zvuk (pozri Obrázok 6). Súčasťou aplikácie sú multimediálne súbory – zvukový súbor a ikona aplikácie, ktoré nahráme v module Designer a nastavíme ku komponentom Screen a Sound.

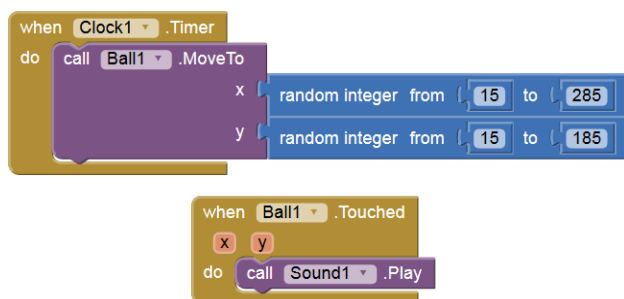


Obrázok 6. Grafický vzhľad hry Postreh.

Správanie aplikácie vo forme trojíc *komponent – udalosť – odpoveď* zachycuje Tabuľka 3 a výsledný programový kód na Obrázku 7.

Tabuľka 3. Správanie hry Postreh 1 zapísané v tabuľke.

komponent	udalosť	odpoveď
hodiny	časovač	presunie loptu na náhodnú pozíciu na plátno (tak aby bola zobrazená celá)
lopta	dotyk	prehrá sa zvuk



Obrázok 7. Programový kód hry Postreh 1.

V tejto aplikácii je použitý komponent *Hodiny* a jeho udalosť *Časovač*, ktorá je veľmi dôležitá a využiteľná v mnohých ďalších aplikáciach.

K ďalším námetom na rozšírenie hry *Postreh* môžeme dospieť, napr. formou brainstormingu. Námety prediskutované so žiakmi na informatickom krúžku sú zachytené na Obrázku 8.

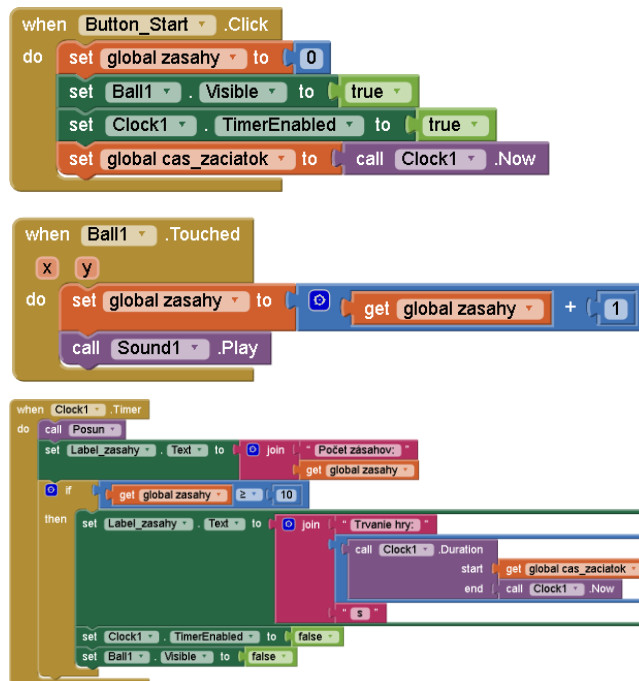


Obrázok 8. Výsledok brainstormingu k rozšíreniam hry Postreh.

Príklady námetov na ďalšie rozšírenie:

1. Registrovanie úspešných dotykov – po každom úspešnom dotyku sa zaznamená a zobrazí počet úspešných dotykov a doplnenie tlačidla na vynulovanie počtu úspešných dotykov.
2. Registrovanie aj neúspešných dotykov – výsledné skóre bude rozdielom úspešných a neúspešných pokusov.
3. Zmena farby lopty pri úspešnom aj neúspešnom zásahu so zvukovým efektom.
4. Ukončenie hry po 10 úspešných dotykoch.
5. Znemožnenie registrácie dotykov na loptu po ukončení hry.
6. Zobrazenie času trvania hry po jej ukončení.
7. Hra sa ukončí po uplynutí 30 sekundového intervalu.
8. Po určitom počte úspešných zásahov sa zrýchli zobrazovanie lopty, resp. aj zmenší veľkosť lopty.
9. Zaznamenanie ostatného skóre hry po opätovnom spustení hry, resp. 3 najlepších hodnôt skóre hry.

Riešenie aplikácie s rozšíreniami 1, 4, 5, 6 je uvedené na Obrázku 9.



Obrázok 9. Časť programového kódu hry Postreh 2.

Úroveň porozumenia vybraným programovým konceptom AI2 môžeme zistiť pomocou didaktického testu. Pri jeho zostavovaní by sme mali zohľadniť, aby jeho jednotlivé položky mapovali prebrané učivo a aby vyžadovali od žiakov zapojenie vyšších poznávacích funkcií. Odporúčame nasledovné typy formulácií testovacích položiek:

- Opíšte, aký problém rieši uvedený programový kód.
- Vyberte, ktorý z uvedených (2-3) programových kódov rieši zadaný problém.
- Vyberte, ktorý z uvedených (2-3) čiastkových programových kódov treba doplniť do uvedeného programového kódu, aby riešil zadaný problém.
- Nájdite a opravte chybu v uvedenom programovom kóde, ktorý má riešiť zadaný problém.

Výsledné žiacke produkty môžeme hodnotiť pomocou pozorovacieho hodnotiaceho hárku, v ktorom môžeme sledovať – funkčnosť aplikácie, množstvo pridaných funkcionalít, programátorskú náročnosť, použitie multimediálnych súborov, využiteľnosť aplikácie v praxi, atď.

V Tabuľke 4 uvádzame, ktoré komponenty, udalosti, metódy, vlastností a programové koncepty sme pokryli aplikáciami Kresliaci editor 1-3 a hra Postreh 1-2.

**Tabuľka 4.** Pokrytie komponentov, udalostí, metód, vlastností a programových konceptov uvedenými aplikáciami

Aplikácia	Komponenty (udalosti)	Metódy, vlastnosti, programové koncepty
Kresliaci editor 1	Canvas (Touched)	Canvas (DrawCircle)
Kresliaci editor 2	Canvas (Touched, Dragged), AccelerometerSensor (Shaking)	Canvas (DrawCircle, DrawLine, Clear)
Kresliaci editor 3	Canvas (Touched, Dragged), AccelerometerSensor (Shaking), Button (Click), TextBox (Text), Camera (AfterPicture), Clock, Screen (Initialize)	Canvas (DrawCircle, DrawLine, Clear, DrawText, Height, Width, PaintColor, LineWidth, BackgroundImage, SaveAs), TextBox (Text), Camera (TakePicture), Clock (Now, FormatDateTime), aritmetické operácie ( $\times$ , $-$ ), join, close application
Hra Postreh 1	Clock (Timer), Ball (Touched)	Ball (MoveTo), Sound (Play), generovanie náhodného čísla
Hra Postreh 2	Clock (Timer), Ball (Touched), Screen (Initialize), Label	Ball (MoveTo, Visible), Sound (Play), Clock (Now, Duration, TimerEnabled), Label (Text), generovanie náhodného čísla, globálna premenná (inicializácia, priradenie, inkrementácia), aritmetické operácie ( $/$ , $+$ ), logické hodnoty true/false, vlastná procedúra, join, if-then

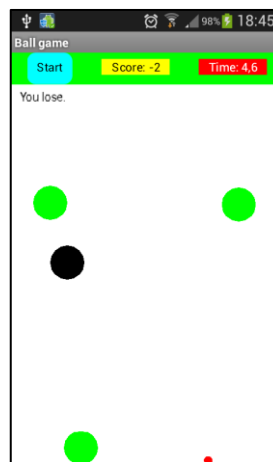
## 4. NÁMETY ĎALŠÍCH APLIKÁCIÍ V AI2

V tejto časti predkladáme námety ďalších zaujímavých a v praxi využiteľných aplikácií, ktorých riešenia uvádzame na webovej stránke [15].

### 4.1 Hra Guľôčka

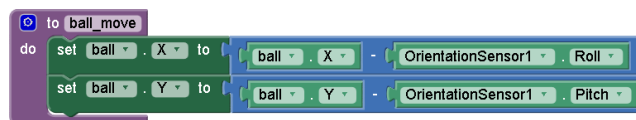
Vytvorte hru Guľôčka, v ktorej nakláňaním mobilného zariadenia ovládáte malú červenú guľku tak, aby sa čo najskôr dotkla troch zelených kruhov a vyhla sa čiernemu kruhu predstavujúcemu jed. Pri dotyku guľôčky so zeleným kruhom, kruh zmizne a získavate za neho 10 bodov, pri dotyku s okrajom plátna strácate 2 body a pri dotyku s jedom (čiernym kruhom) strácate 40 bodov. Hra končí po dotknutí sa troch zelených kruhov alebo pri získaní záporného skóre hry.

Na Obrázku 10 je zobrazený vzhľad hry Guľôčka – tlačidlo Štart, texty so skóre a časom, plátno s jednou aktívnou a štyrmi statickými loptami.



**Obrázok 10.** Nainštalovaná aplikácia hra Guľôčka.

Pri programovaní tejto aplikácie použijeme *orientačný senzor*, ktorého vlastnosti *Roll* (naklonenie zľava-doprava) a *Pitch* (naklonenie dopredu-dozadu) použijeme pri zmene súradníc guľôčky. Na Obrázku 11 je uvedená časť kódu súvisiaca so zmenou súradníc guľôčky podľa nakláňania mobilného zariadenia.



**Obrázok 11.** Časť programového kódu hry Guľôčka.

Obmenou tejto hry, ktorá využíva orientačný senzor na pohyby guľôčky, je hra Bludisko, v ktorej testovanie voľnej dráhy môžeme urobiť pomocou funkcie *Canvas.GetPixelColor*.

### 4.2 Hovoriaci kompas

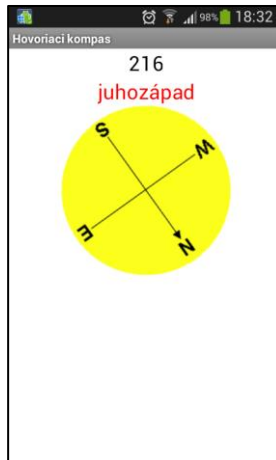
Vytvorte aplikáciu pre slabozrakých, či zaneprázdnených, ktorá okrem zobrazenia azimutu a svetovej strany na kompas bude hovoriť aktuálnu svetovú stranu syntetickým hlasom.

Pri programovaní tejto aplikácie použijeme *orientačný senzor*, a jeho vlastnosť *Azimuth*, ktorú využijeme pre natočenia obrázka s kompasom. Pre uloženie a sprístupnenie názvov svetových strán použijeme údajový typ *zoznam*. Na prečítanie svetovej strany syntetickým hlasom použijeme metódu *TextToSpeech.Speak*.

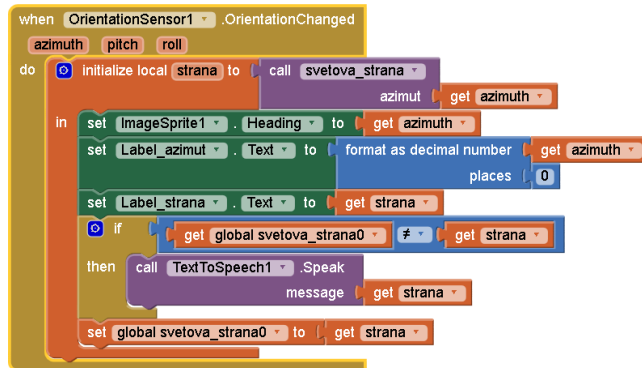
Ak inštalovaná aplikácia na syntézu reči nepodporuje slovenčinu, dá sa nainštalovať z Google play aplikácia eSpeak TTS od Eyes-Free Project.



Na Obrázku 12 je zobrazená nainštalovaná aplikácia Hovoriaci kompas a na Obrázku 13 časť programového kódu tejto aplikácie.



Obrázok 12. Nainštalovaná aplikácia Hovoriaci kompas.

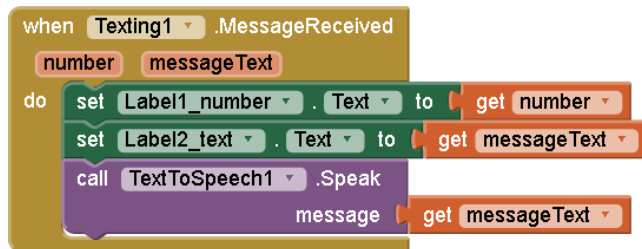


Obrázok 13. Časť programového kódu aplikácie Hovoriaci kompas.

### 4.3 Čítačka SMS

Vytvorte aplikáciu pre slabozrakých, či zaneprázdnených, ktorá prečíta nahlas prichádzajúcu SMS správu syntetickým hlasom.

Pri programovaní tejto aplikácie okrem syntezátora reči použijeme neviditeľný komponent *Texting* (určený pre prijímanie a odosielanie SMS správ) a udalosť *Texting.MessageReceived*. Na Obrázku 14 je uvedený kompletný kód tejto aplikácie.

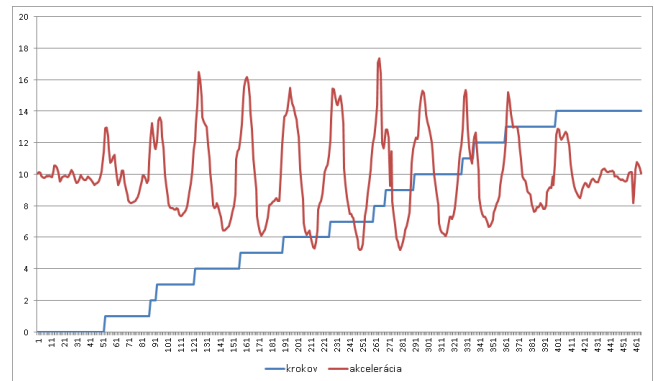
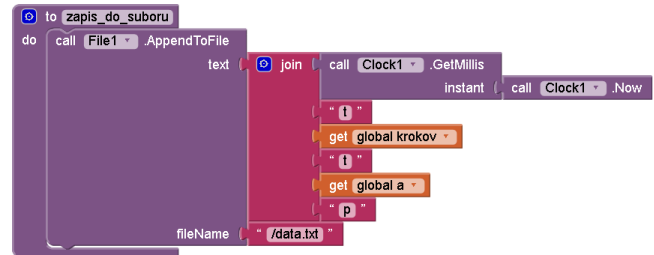


Obrázok 14. Programového kód aplikácie Čítačka SMS.

### 4.4 Krokomer

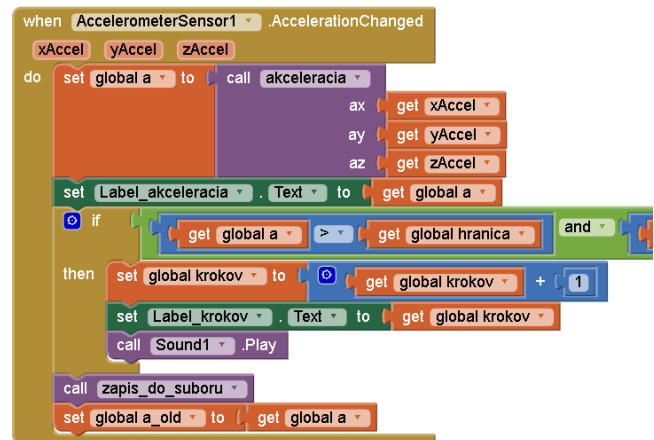
Vytvorte mobilnú aplikáciu, ktorá bude merať počet prejdenných krokov.

Počet krokov budeme merať pomocou senzora akcelerácie. Krok zarátame len vtedy, ak bude aktuálna hodnota akcelerácie väčšia ako posúvačom nastavená medzná hodnota a zároveň predchádzajúca hodnota akcelerácie bude menšia ako medzná hodnota. Pre lepšiu predstavu o priebehu akcelerácie pri rôznych aktivitách (krokoch, drepoch, dvíhaní rúk, páde) môžeme zaznamenať priebeh akcelerácie uložením do textového súboru.



Obrázok 15. Programový kód zaznamenávajúci priebeh hodnôt senzora akcelerácie pri krokaní a jeho vizualizácia.

Na Obrázku 16 je uvedená časť programového kódu počítajúca kroky so sprievodným zvukom s nezobrazenou druhou časťou podmienky



Obrázok 16. Časť programového kódu, ktorá počíta kroky.

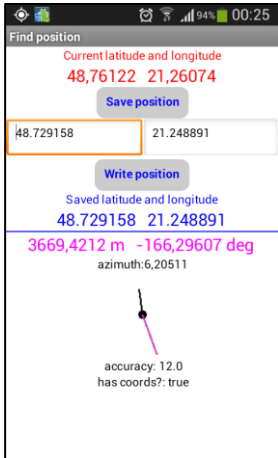
Okrem počítania krokov, drepov, zdvihov rúk, ... vieme upraviť aplikáciu tak, aby zaregistrovala pád seniora a zaslala SMS správu s jeho GPS polohou a azimutom na vybrané telefónne číslo.



### 4.5 Hľadaj uloženú GPS pozíciu

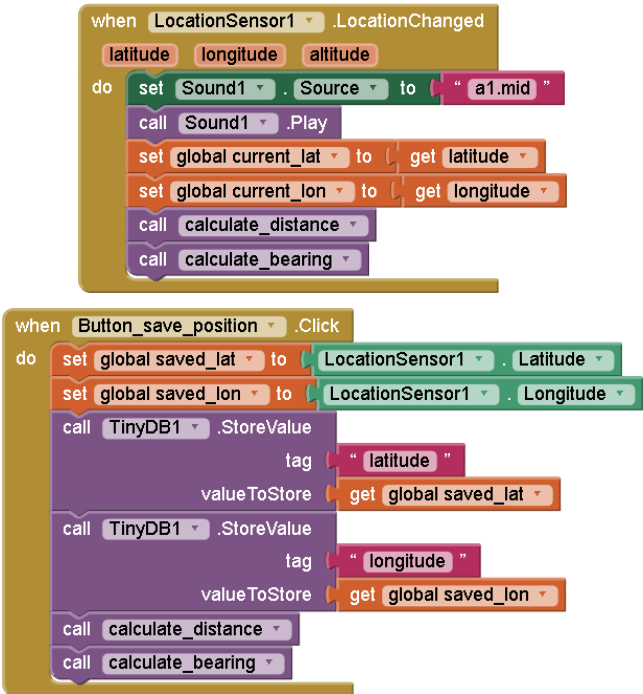
Vytvorte mobilnú aplikáciu, ktorá umožní uložiť GPS pozíciu miesta a neskôr sa na to miesto vrátiť na základe vypočítania vzdialenosti a smeru od aktuálnej GPS pozície k uloženej GPS pozícii.

Na obrázku 17 je zobrazená nainštalovaná aplikácia s aktuálnou a uloženou GPS pozíciou a vypočítanou vzdialenosťou a smerom od aktuálnej k uloženej GPS pozícii.



Obrázok 17. Nainštalovaná aplikácia Hľadaj GPS pozíciu.

Pre načítanie GPS polohy použijeme *LocationSensor* a jeho vlastnosti *Latitude* a *Longitude*. Pre uloženie a načítanie uloženej GPS pozície použijeme komponent *TinyDB* a jeho metódy *StoreValue* a *GetValue*.

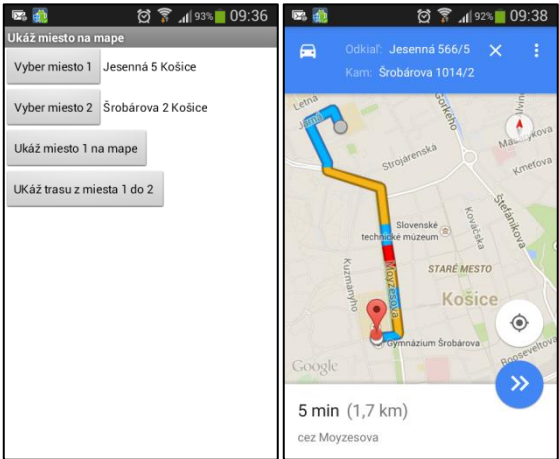


Obrázok 18. Časť kódu aplikácie Hľadaj GPS pozíciu.

### 4.6 Ukáž miesto/trasu na mape

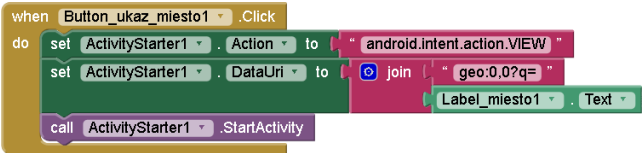
Vytvorte mobilnú aplikáciu, ktorá zo zoznamu adries umožní vybrať adresu/adresy a ukázať miesto/trasu medzi miestami pomocou aplikácie Google maps.

Na Obrázku 19 je zobrazená nainštalovaná aplikácia Ukáž miesto/trasu na mape – vľavo tlačidlá na výber miest, vpravo zobrazená trasa na Google maps medzi zadanými miestami.



Obrázok 19. Nainštalovaná aplikácia Ukáž miesto/trasu na mape.

Na Obrázku 20 je uvedený programový kód aplikácie spúšťajúcej inú aplikáciu (Google maps) na zobrazenie miesta s uvedenou adresou.



Obrázok 20. Časť kódu aplikácie Ukáž miesto/trasu na mape.

### 4.7 Ďalšie aplikácie

Vytvorili sme rad ďalších aplikácií, napr. Generátor náhodných viet, Stopky s hodinami, Klavír s prehrávačom skladieb. Tieto aplikácie okrem dotyku nevyužívajú veľmi špecifické funkcionality mobilného zariadenia, preto ich motivačný potenciál je menší ako v aplikáciách uvedených v podkapitolách 4.1 – 4.6. Na druhej strane majú aj tieto aplikácie veľký význam, hlavne pri precvičovaní rôznych programových konceptov.

Tabuľka 5. Pokrytie komponentov, udalostí, metód, vlastností a programových konceptov uvedenými aplikáciami 4.1 – 4.6.

Aplikácia	Komponenty (udalosti)	Metódy, vlastnosti, programové koncepty
Hra Gul'ôčka	Screen (Initialize), Canvas, Ball (CollidedWith), Clock (Timer), Sound, OrientationSensor	Clock (TimerEnabled), Sound (Play, Source), Label (Text), Ball (X, Y, Radius, Visible, EdgeReached), Canvas (DrawText, Clear, Width, Height), OrientationSensor (Roll, Pitch), generovanie náhod. čísla, globálna premenná (inicializácia, priradenie, inkrementácia), aritmetické operácie (/ , +),

		logické hodnoty true/false, vlastná procedúra, join, if-then
Hovoriaci kompas	Canvas, ImageSprite, Label, OrientationSensor (OrientationChanged), TextToSpeech,	Label (Text), ImageSprite (Heading), TextToSpeech (Speak), vlastná procedúra, údajový typ zoznam (vytvorenie, výber prvku), globálna a lokálna premenná (inicializácia, priradenie, porovnanie), funkcie modulo, floor, format as decimal number, aritmetické operácie (+, -, ×, /), if-then
Čítačka SMS	Label, Texting (MessageReceived), TextToSpeech	Label (Text), TextToSpeech (Speak)
Krokomer	Label, Button (Click), Slider (PositionChanged), Screen (Initialize), Accelerometer (Ac- celerationChanged), Clock, Sound, File	Label (Text), Clock (Now, GetMillis), Sound (Play), File (AppendToFile, Delete), vlastná procedúra a funkcia, globálna premenná (inicializácia, priradenie, porovnanie), zložená podmienka (and), aritmetické operácie (+, ^), sqrt, join, if-then
Hľadáč uloženú GPS pozíciu	Screen (Initialize), Canvas, Label, Button (Click), TextBox, LocationSensor (LocationChanged), OrientationSensor, Clock (Timer), TinyDB, Sound	Canvas (Clear, PaintColor, DrawCircle, DrawLine), Label (Text, TextColor), TextBox (Text), Sound (Play, Source), LocationSensor (Latitude, Longitude, Accuracy, HasLongitudeLatitude), OrientationSensor (Azimuth), TinyDB (GetValue, StoreValue), globálna a lokálna premenná (inicializácia, priradenie), vlastná procedúra, funkcie sin, cos, atan2, sqrt, convert degrees to radians, aritmetické operácie (+, -, ×), join, if-then, if-then-else
Ukáž miesto/trasu na mape	Screen (Initialize), Label, ListPicker (BeforePicking, AfterPicking), Button (Click), ActivityStarter	Label (Text), ActivityStarter (StartActivity, Action, DataUri), globálna premenná (priradenie), join

## 5. ĎALŠÍ VÝVOJ METODIKY VÝUČBY PROGRAMOVANIA V AI2 A JEJ ŠÍRENIE

Po vzore metodík a pracovných listov k výučbe aplikácií Kresliaci editor a hra Postreh vytvárame aj k ďalším uvedeným aplikáciám metodiky s pracovnými listami, ktoré budeme overovať v rámci

stretnutí informatického krúžku, vo výučbe na vybraných gymnáziách a tiež v rámci vysokoškolskej výučby.

Žiaci na informatickom krúžku sa každoročne obmieňajú. Od októbra 2014 do polovice februára stihli žiaci naprogramovať Kresliaci editor, hru Postreh a hru Gulfôčka. Do júna 2015 plánujeme naprogramovať aplikácie Hovoriaci kompas, Krokomer, Čítačku SMS, Hľadáčku GPS pozície a Zobrazenie trasy na mape.

V rámci vysokoškolskej prípravy učiteľov informatiky vyučujeme nový predmet Školské programovacie prostredia, v rámci ktorého získajú študenti prehľad školských programovacích prostredí a naprogramujú viaceré projekty v prostrediach Scratch a AI2 (Kresliaci editor, hru Postreh a vlastnú kreatívnu aplikáciu). Na Obrázku 21 sú zobrazené študentské aplikácie Kresliaci editor.



**Obrázok 21.** Výsledné aplikácie Kresliaci editor na konci druhého cvičenia predmetu Školské programovacie prostredia.

V predmete Didaktika informatiky v letnom semestri 2014/2015 budú študenti v rámci jedného priebežného zadania vytvárať pracovný list k výučbe vybranej aplikácie v prostredí AI2.

Naše skúsenosti s výučbou programovania v prostredí AI2 spolu s metodikami a pracovnými listami sme prezentovali učiteľom informatiky základných a stredných škôl na dvoch stretnutiach Klubu učiteľov informatiky (26. 6. 2014 a 28. 10. 2014) a tiež na letnej škole projektu VEMIV (1. 7. 2014).



**Obrázok 22.** Stretnutie Klubu učiteľov informatiky (26. 6. 2014) venované programovaniu v AI2 s 12 prezenčnými účastníkmi a trojicou vzdialených účastníkov z Nitry (na Skype).

## 6. ZÁVER

Výučba programovania v AI2 je veľmi atraktívna, dynamicky sa rozvíjajúca, obsahovo aj rozsahovo veľmi náročná oblasť. Učitelia, ktorých programovanie v AI2 zaujalo a radi by ho vyučovali majú veľmi náročnú situáciu, lebo okrem zvládnutia samotného programovania v AI2 si potrebujú pripraviť plán výučby, pracovné listy, učebné texty, písomky, hodnotiace škály projektov, atď.

V tomto príspevku sme predstavili aktuálny stav nášho skúmania problematiky výučby mobilných aplikácií v AI2. V najbližšom období plánujeme na základe vlastnej výučby a tiež spolupráce s učiteľmi gymnázií vylepšiť existujúce a doplniť nové metodiky a pracovné listy pre programovanie zaujímavých a užitočných mobilných aplikácií.

Veríme, že týmto príspevkom sme inšpirovali a naštartovali ďalších učiteľov v problematike programovania mobilných aplikácií v prostredí AI2 a hlavne jeho výučby na našich základných a stredných školách.

## 7. POĎAKOVANIE

Táto práca bola podporovaná Agentúrou na podporu výskumu a vývoja na základe Zmluvy č. APVV-0715-12 (Výskum efektívnosti metód inovácie výučby matematiky, fyziky a informatiky).

## 8. BIBLIOGRAFICKÉ ODKAZY

- [1] *Projekt Elektronizácia vzdelávacieho systému regionálneho školstva (Digiškola)* [online]. [cit. 2015-02-09]. Dostupné na: <<http://www.digiskola.sk/>>.
- [2] *Škola na dotyk* [online]. [cit. 2015-02-09]. Dostupné na: <<http://akademia.edulab.sk/>>.
- [3] LOVÁSZOVÁ, Gabriela, PALMÁROVÁ, Viera. Edukačné hry pre mobilné zariadenia s navigáciou. In: *DidInfo 2013 : 19. ročník konferencie s medzinárodnou účasťou o vyučovaní informatiky na všetkých typoch škôl* (Banská Bystrica : Univerzita Mateja Bela v Banskej Bystrici, 10. - 12. 4. 2013). ISBN 978-80-557-0527-9, s. 160-166.
- [4] SLANY, Wolfgang. Tinkering with Pocket Code, a Scratch-like programming app for your smartphone. In: *Constructionism and Creativity : Proceedings of the 3rd International Constructionism Conference* (Vienna :

Osterreichische Computer Gesellschaft, 19. - 23. 8. 2014) ISBN 9783854033011. p. 83-92.

- [5] WOLBER, David, ABELSON, Harold, FRIEDMAN, Mark. Democratizing Computing with App Inventor. In: *ACM SIGMOBILE Mobile Computing and Communications Review archive*. Volume 18, Issue 4, October 2014. ACM, New York, NY, 2014, p. 53-58. DOI= <<http://doi.acm.org/10.1145/2721914.2721935>>.
- [6] WOLBER, David. *The App Inventor Course-in-a-Box*. [online]. [cit. 2015-02-09]. Dostupné na: <<http://www.appinventor.org/content/CourseInABox/Intro/courseinabox>>
- [7] WOLBER, David, ABELSON, Hal, SPERTUS, Ellen, LOONEY, Liz. *App Inventor 2: Create your own Android Apps*. O'Reilly Media, 2014. ISBN 978-1491906842
- [8] TOMCSÁNYOVÁ, Monika. *Metodické materiály a aplikácie pre Android vytvorené v Appinventore*. [online]. [cit. 2015-02-09]. Dostupné na: <<http://edi.fmph.uniba.sk/~tomcsanyiova/>>.
- [9] GALBAVÁ, Ľudmila. Využitie MIT App Inventora v PC a na tabletoch (video). [online]. [cit. 2015-02-09]. Dostupné na: <http://youtu.be/ifiYJq00GKsw>
- [10] *Vývojové prostredie MIT App Inventor 2*. [online]. Dostupné na: <<http://ai2.appinventor.mit.edu/>>
- [11] *App Inventor Community Gallery*. [online]. Dostupné na: <<http://gallery.appinventor.mit.edu/>>
- [12] *Tutorials for App Inventor 2*. [online]. Dostupné na: <<http://explore.appinventor.mit.edu/ai2/tutorials>>
- [13] ŠNAJDER, Ľubomír, DANESHJOOVÁ, Danka, GONDOVÁ Valéria. *Informatické pracovné listy s bádateľskými aktivitami*. 1. vydanie. Košice : Ústav Experimentálnej fyziky SAV, 2012. 66 s. ISBN 9788097077976. [online]. Dostupné na: <<http://susy.saske.sk/rok-2012/iplba/>>
- [14] *Moodle kurz projektu Výskum efektívnosti metód inovácie výučby informatiky, fyziky a informatiky*. [online]. Dostupné na: <<http://skoly.upjs.sk/aktivity/mod/folder/view.php?id=959>>
- [15] ŠNAJDER, Ľubomír. *Výučba programovania v AI2*. [online]. Dostupné na: <<http://ics.upjs.sk/~snajder/ai2/>>

# Ochrana detí pred nebezpečenstvom, ktoré im hrozí z prostredia internetu

Tatiana Hajdúková  
Akadémia Policajného zboru  
Sklabinská 1  
835 17 Bratislava  
Slovakia  
tatiana.hajdukova@minv.sk

Ivan Bacigál  
Odbor počítačovej kriminality  
Úrad kriminálnej polície Prezídia PZ  
Pribinova 2  
812 72 Bratislava  
Slovakia  
ivan.bacigal@minv.sk

## ABSTRAKT

O problematike zneužívania verejnej siete internetu na trestnú alebo inú protispoločenskú činnosť sa najmä v zahraničí rozpráva už niekoľko rokov. Zvýšené riziko hrozí najmä deťom a mládeži, ktorí na sociálnych sieťach predstavujú viac ako 60% spomedzi všetkých užívateľov. Napriek veľkej popularite komunikácie detí a mládeže online, nepodarilo sa dostatočne presvedčiť osoby zodpovedné a podieľajúce sa na výchove a vzdelávaní detí na Slovensku, že tejto problematike nie je venovaná dostatočná pozornosť. K potenciálnym rizikám ohrozujúcich deti patrí populárny fenomén súčasnosti, internetové známosti, ktoré sa nie vždy končia šťastne a bez následkov. Ak je dieťa zaujaté prácou na počítači a v tichosti zavreté v detskej izbe domova ešte neznamená, že nie je v ohrození. Odvracať tvár a nepočúvať varovné signály sa dá len do momentu, kým sa kľbko problémov nezačne bezprostredne dotýkať konkrétneho dieťaťa, kedy už obvykle býva neskoro. Spomienky a následky na psychike dieťaťa sú obvykle trvalé a ani intervencia psychológa nedokáže vymazať bolestivú minulosť z pamäte. Prečo to tak je? Aké sú dôsledky dokončených i nedokončených trestných činov? Akú úlohu má a akú by mohlo mať doplnenie obsahu vzdelávania na informatike a intenzívnejšia prevencia a v oblasti bezpečnejšieho a zodpovednejšieho správania sa detí na internete je predmetom nasledovného príspevku.

## ABSTRACT

The issue of abuse of public Internet network to a penalty or other antisocial activities mainly abroad talks for several years. Increased risk of, in particular children and young people who are on social networks represent more than 60 % among all users. In spite of the popularity of communication for children and young people online, failed to convince enough people responsible and involved in the education of children in Slovakia that this issue is not given sufficient attention. The potential risks for children is now a popular phenomenon, online reputation, which is not always end happily and without negative consequences. If the child is interested in computing and closed in the nursery home does not mean that it is not in danger. Avert face and not to listen to the warning signs can only be in the moment, until the ball begins to issues of direct concern to a particular child, which has usually lives too late. Memories and effects on the psyche of the children are usually permanent and intervention or psychologist can not erase painful memories from the past. Why is that? What are the consequences by perfect and inchoate offenses? What is the role and what would have to supplement the education in computer science and intensified prevention and in safer and more

responsible behavior of children on the Internet is subject to the following contribution.

## Kľúčové slová

Bezpečnosť, virtuálna komunikácia, neznáma identita, počítačová kriminalita, prevencia, vzdelávanie

## Keywords

Security, Virtual Communication, Identity Unknown, Cybercrime, Prevention, Education

## ÚVOD

Učitelia informatiky sú často nielen učiteľmi informatiky, ale v rámci úsporných opatrení na školách ako i z dôvodu, že potrebujú mať nastavenia potrebné pre výučbu spravené včas a presne podľa svojej potreby, sú často aj počítačovými správcami. Učitelia informatiky a počítačoví správcovia (poradie sa mení od uhla pohľadu) tak musia počas svojej práce skontrolovať veľké množstvo elektronických záznamov. Ako učitelia ich kontrolujú za účelom zlepšovania nastavení a v čase keď sú správcami preto, aby overili stabilitu, bezporuchovosť a bezpečnosť školského, často nimi vytvoreného, výučbového informačného systému. Počas tejto činnosti však často narazia na informácie, ktoré s úlohami pedagóga, či správcu nesúvisia; nepomôžu ani k zlepšeniu výučby a ako také ani neohrozia bezpečnosť či stabilitu systému.

Napriek tomu, že množstvo spracovávaných informácií, ani ich technická dôležitosť neprekročilo bežnú hranicu, ďalšie spracovávanie sa zastavuje a začne svietiť neviditeľná červená kontrolka s nápisom „Pozor“, pretože niečo prekročilo hranicu, ktorá pritom vôbec nesúvisí s predmetom informatika, či informatikou ako takej.

Je mnoho informácií, ktoré nás dokážu vyvieť z miery a jednou z nich je narušenie zaužívaných spoločenských a morálnych noriem. Tie najzávažnejšie formy narušenia sú zároveň porušením zákonných noriem. V spleti zákonov však nie vždy je však ľahké rozlíšiť či a v akej forme porušenia prišlo. Počítačová kriminalita v sebe zahŕňa širokú škálu trestných činov, prečo nie je možné v definícii vyjadriť rôznorodosť jej obsahu. Tieto protiprávne trestné činy však majú zjednocujúci prvok, ktorým je použitie informačných technológií pri jej páchaní. Pre možnosť preskúmania či prišlo k porušeniu zákona ako aj na vyhľadávanie porušení zákona je v rámci polície vyčlenená skupina policajtov všeobecne známych pod názvom Kriminálna polícia. Tá oznámenia od občanov či vlastné zistenia preverí a pokiaľ prišlo k porušeniu,



postúpi informácie o podozrení zo spáchania trestného činu na začatie trestného stíhania.

Počítače sú často krát využité len ako nepriamy prostriedok na páchanie klasickej trestnej činnosti klasifikovanej ako ekonomická, mravnostná, či majetková kriminalita. Okrem úmyselných trestných činov poznáme aj tzv. omisívne trestné činy, teda trestné činy vznikajúce nekonaním, hoci občan mal povinnosť konať. Medzi tieto trestné činy patria trestné činy uvedené v Trestnom zákone a to najmä § 340 Neoznámenie trestného činu a § 341 Neprekazenie trestného činu. Špecifickým trestným činom, ktorého sa môžu dopustiť správcovia informačných systémov svojou činnosťou je trestný čin podľa § 344 Marenie spravodlivosti a trestný čin podľa § 339 Nadržovanie, ktorý môžu spáchať zmenami v informačných systémoch, keď pozmenia alebo zmazia dôkazy alebo zabránia v získaní takéhoto dôkazu.

Učelia informatiky a správcovia informačných systémov škôl sa môžu stretnúť najmä s tými to trestnými činmi: § 171 Nedovolená výroba omamných a psychotropných látok, jedov alebo prekurzorov, ich držanie a obchodovanie s nimi, § 179 Obchodovanie s ľuďmi, § 201 Sexuálne zneužívanie, § 208 Týranie blízkej osoby a zverenej osoby, § 211 Ohrozovanie mravnej výchovy mládeže, § 247 Poškodenie a zneužitie záznamu na nosiči informácií, § 370 Prechovávanie detskej pornografie a účasť na detskom pornografickom predstavení, § 360 Nebezpečné vyhrážanie, § 360a Nebezpečné prenasledovanie, § 367 Kupliarstvo, § 368 Výroba detskej pornografie, § 369 Rozširovanie detskej pornografie, § 371 Ohrozovanie mravnosti a § 373 Ohováranie.

Okrem týchto trestných činov sa môžu stretnúť aj s trestnými činmi technickejšieho charakteru a to s trestnými činmi § 377 Porušenie dôvernosti ústneho prejavu a iného prejavu osobnej povahy a § 247 Poškodenie a zneužitie záznamu na nosiči informácií.

Ďalšou náplňou práce učiteľa informatiky ako počítačového správcu sú počítačové incidenty ako dôsledok počítačových útokov (interných či externých používateľov). Počítačové incidenty sú každodennou realitou, pričom ich riešenie komplikovanosť riešenia sa mení na základe použitých technológií. Rovnako ako v prípade každého bezpečnostného incidentu, musí byť časť práce súvisiaca s odrazením útoku a identifikáciou škôd uskutočnená skôr, ako k útoku vôbec príde. Zodpovedajúce plánovanie prevencie je v tomto prípade nevyhnutné. Akonáhle však nebezpečná situácia nastane, je dôležitá rýchla a inteligentná reakcia, ktorá zabráni ďalším škodám a odhalí informácie vedúce k identifikovaniu útočníka. Na zaistenie relevantných dôkazov uložených na počítačovom systéme (páchateľa či obete) sú okrem odborného vzdelania získaného absolvovaním špecializovaných školení potrebné aj reálne skúsenosti, ktoré sa nedajú ničím iným ako praxou nahradiť. Počítačových expertov, ktorí majú dostatočné dlhodobé skúsenosti potrebné pre včasnú a efektívnu reakciu na útok prichádzajúci zo siete je nedostatok v celosvetovom meradle, nielen na Slovensku. Reakcia na útok zároveň nesmie znehodnotiť žiadne dôležité dôkazy, musí umožňovať získať dostatočné množstvo dát pre odhalenie podstaty útoku a identity útočníka.

## 1. Výchova a vzdelávanie žiakov z informatiky na základných a stredných školách

Výchova a výučba informatiky oprávnené patrí medzi dôležité povinné predmety všeobecného formálneho vzdelávania na slovenských školách. Opodstatnene sa podieľa na vyváženom rozvoji osobnosti žiakov a ich individuálnom rozvíjaní sa pre potreby súčasnej a budúcej spoločnosti. Štátne vzdelávacie programy (ŠVP) definujú rámcový model absolventa, rámcový plán a rámcové učebné osnovy pre všetky stupne základnej a strednej školy okrem iného aj vo vzdelávacích oblastiach Informatická výchova a Informatika pri rozvíjaní IKT zručností. Nie sme kompetentní hodnotiť obsah vzdelávania z informatiky, ktorý sa nám javí ako náročný, napriek tomu považujeme za potrebné upozorniť na nedostatočnú pozornosť venovanú bezpečnostným aspektom používania IKT najmä z hľadiska ohrozenia žiakov.

Súčasťou informatiky podľa ŠVP je výchova k rešpektovaniu právnych a etických zásad používania informačných technológií a produktov najmä z pohľadu rešpektovania intelektuálneho vlastníctva, vlastníctva intelektuálnych produktov, či legálnosť použitia obrázkov a textov z internetu. Dodržiavanie týchto zásad prikazuje žiakom, aby sa sami správali zodpovedne za čo by v prípade nedržania mali znášať následky. Tento prístup však nehovorí nič o obrane, ak sa žiak stane terčom útoku, hoci len etických zásad a cíti sa ponížený.

Ako výkonový štandard sa od žiaka požaduje schopnosť poslať, a teda aj prijať emailovú správu s prílohou. V tejto súvislosti by sa automaticky a súčasne malo hovoriť o škodlivom kóde, ktorý môže príloha, alebo aj samostatná správa obsahovať. Rozvaha pri otváraní prílohy je nevyhnutnosťou, pričom žiakom by malo byť vštepované, aby prílohu od neznámeho odosielateľa principiálne otvárali s maximálnou opatnosťou.

Práca s emailom na komunikáciu s učiteľom bola v poslednej úprave ŠVP posunutá na skoršie obdobie do 3. ročníka. Emailová komunikácia je rovnako využiteľná aj na komunikáciu s neznámymi ľuďmi, ktorí nemajú najlepšie úmysle, čo môže byť pre dieťa nebezpečné.

Podľa aktuálneho ŠVP by sa žiaci v 2.-4. ročníku mali opatrne zoznámiť a neskôr v 5.- 8. ročníku osvojiť si základy bezpečnosti a rizík v oblasti Informačná spoločnosť. Súčasne s oboznámením sa so základmi bezpečnosti a rizík by mali smelo používať, resp. trénovať nadobudnuté zručnosti v práci v počítačovej sieti a na internete, resp. mali by vedieť pracovať s nástrojmi na komunikáciu. Deti si rýchlo osvojujú najmä zručnosti, ktoré vedú zúžitkovať pri prezentovaní sa na verejnosti, napr. v tak obľúbených sociálnych sieťach. Ak sa žiaci naučia získať, vytvoriť a spracovať fotky na referáty, tento postup je aplikovateľný aj pri tvorbe osobných fotografií a posielaní elektronickou poštou „kamarátom“, ktorých nikdy nevideli a vôbec nemusia byť ich rovesníkmi, za ktorých sa vydávajú. Problém nie je v systematickom rozvíjaní zručností žiakov, pokiaľ by bolo možné kontrolovať ich používanie, alebo by boli žiaci mentálne a osobnostne vyspelí a pripravení reagovať na možné zneužitie tohto nástroja.

**Tabuľka 1. Výkonový štandard a jeho možné použitie/zneužitie v informatickej výchove/informatike**

Výkonový štandard	obdobie	Podľa ŠVP	Rizikové využitie (nedostatok) zručnosti
Digitálnym fotoaparátom odfotiť spolužiaka a fotku za pomoci učiteľa uložiť do priečinku	ISCED 1	spolužiaka	Zverejnenie fotiek na sieti
Pracovať s elektronickou poštou, vie používať niektorý z nástrojov na interaktívnu komunikáciu (rozhovor (chat), okamžitá správa (ICQ))	ISCED 1	s učiteľom, so spolužiakmi	s neznámou osobou
Vyhľadávať informácie na webe pomocou vyhľadávacích strojov a katalógov, poznať základné postupy pri využívaní internetu, vyhľadať tituly v elektronickej knižnici	ISCED 2	na tvorbu prezentácií, projektov, učenie sa	vyhľadávať nevhodný obsah
Poznať a dodržiavať pravidlá NETIKETY	ISCED 3	Zásady má dodržiavať a dodržiavať žiak	Absentuje obrana, keď sa žiak stal obeťou NETIKETY

V rámci ŠVP je vytvorený istý priestor aj pre počítačovú bezpečnosť, avšak s obsahom definovaným pomerne neurčito a doširoka. Populárne rozprávania *Kto, prečo a ako pácha počítačovú kriminalitu, čo môže spôsobiť* [2, s. 7] od osoby, ktorá sa týmto problémom zapodieva len z povinnosti môže mať nádherné senzácie a výnimočnosti odpútavajúci pozornosť od vážnejšieho problému naznačenom vyššie, ktorým sú žiaci reálne vystavovaní.

## 2. Nebezpečenstvá komunikácie online

Nepostačuje vnímať výučbu informatiky výlučne ako cestu *osvojovania si postupov a techník používaných pri práci s údajmi a informácií v informačných systémoch* [1, s. 2] pretože žiaci spontánne pokračujú vo využívaní IKT na svoje voľno časové aktivity bez dohľadu učiteľov a rodičov. Okrem prípravy na vyučovanie, kedy dochádza najmä ku interakcii medzi žiakom a počítačom alebo vyhľadávaním na webe, veľa času venujú hraníu počítačovým hier a nemenší záujem prejavujú o online komunikáciu s reálnymi a virtuálnymi kamarátmi prostredníctvom internetovej siete.

Hoci je internet primárne sociálne prostredie, v ktorom sa komunikuje so známymi priateľmi, dochádza v ňom aj k zoznamovaniu s novými osobami. Nadviazanie kontaktu prostredníctvom internetu je rýchle a umožňuje výmenu názorov aj s osobami, ktoré by sme nestretli osobne.

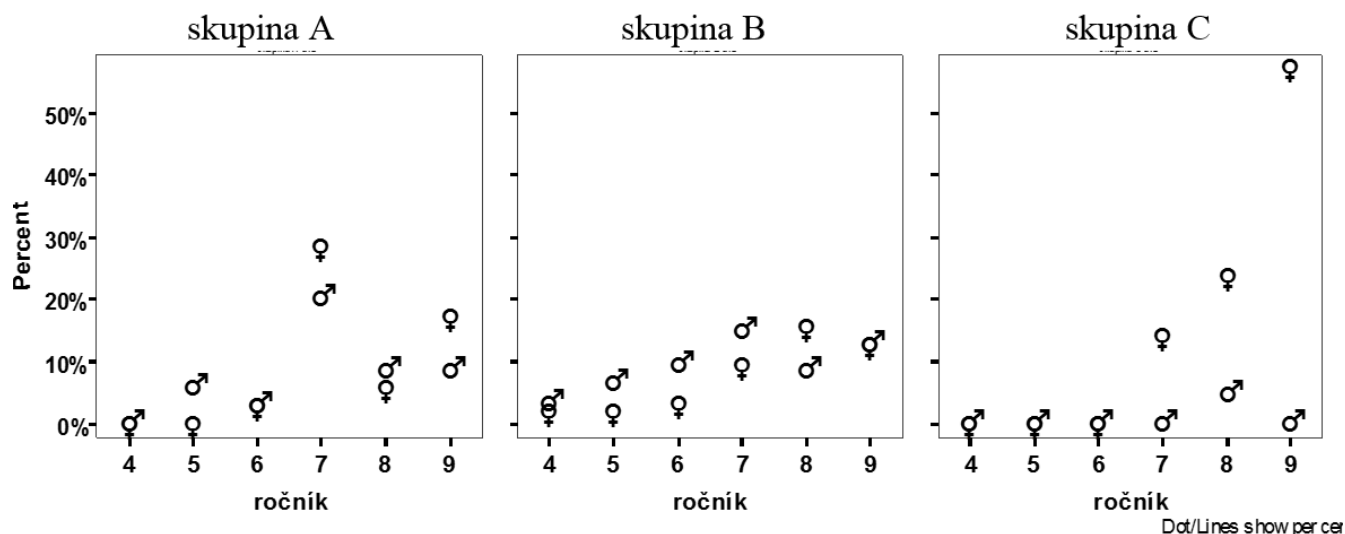
Komunikácia z domova v kombinácii s anonymitou virtuálneho priestoru vytvára najmä u detí zdanlivý pocit bezpečia a povzbudzuje ich, aby sa komunikácia rýchlejšie otvorila a odložili zábrany. Rizikom je, že nie vždy, či dokonca nikdy si nemôžeme byť istí, kto je osoba na druhej strane. Ilúzia dokáže urobiť veľký zmatok s detskou zraniteľnou a dôverčivou myslou s dlhodobými následkami. Mládež nemá zábrany sprostredkovať ani svoje intímne zážitky, čo ľuďom s nie najlepšimi úmyslami dáva do rúk zbrane, ktoré potrebujú na ceste ku svojej potenciálnej obeti. Stačí jediná intímnejšia osobná fotka zverejnená na internete, alebo odoslaná do nesprávnych rúk, čo môže roztočiť dusivú špirálu, z ktorej dieťa bez pomoci dospelých nedokáže vyskočiť. Aby dieťa

nebolo odvážnou fotkou verejne zahanbené, aby sa to nedozvedeli rodičia a ani v škole, treba útočníkovi poslať ďalšie fotky a potom ďalšie fotky. V niektorých prípadoch môže prísť aj ku osobnému stretnutiu, pri ktorom sú dieťaťu ponúkané hračky, mobil, tablet, iné dary, alebo peniaze. Agresori cielene vyhľadávali deti so sociálne slabších skupín, ktoré sa týmto spôsobom môžu dostať k veciam, ktoré boli dovtedy pre nich nedostupné. Samozrejmosťou fungovania tohto obchodného vzťahu je, že o tomto spoločnom „tajomstve“ sa nikto nesmie dozvedieť. Dieťa sa postupne dostane do izolácie, z ktorej sa nedokáže vyslobodiť samé. Komunikovanie, otvorené rozhovory, pozorovanie, zaujímanie sa o dieťa je jediný spôsob, ako mu pomôcť a hľadať riešenie spoločne s dospelými a kompetentnými.

Internetová sieť je len ďalší komunikačný kanál, v ktorom sa prejavujú vlastnosti a správanie ľudí a pozitívne i negatívne javy, ktoré sa vyskytujú medzi ľuďmi aj pri reálnom stretnutí avšak so zhubnejším potenciálom. Dishibitný efekt, vyššia anonymita, široké publikum, ľahká dostupnosť sú pre žiakom povzbudením ku seba prezentovaniu sa a naopak pre agresorov umožnenie vyhľadávať obeť napr. na získavanie ich osobných údajov či intímnych fotiek s následným vydieraním, alebo dohovárania si osobných stretnutí. Žiaci by potrebovali zvýšiť povedomie, že v on-line svete je ťažisko zodpovednosti za vlastnú bezpečnosť na sebe samom a aby sa nestali obeťami mali by sa aktívne podieľať na jej naplňaní. Hovoríme tak na základe skúseností získaných pri preventívnych akciách na základných a stredných školách v školskom roku 2013/14 s 578 účastníkmi v západoslovenskom kraji.

Pri stretnutiach sme zisťovali okrem iného priame skúsenosti žiakov s osobnými stretnutiami s osobami spoznanými výlučne z prostredia internetu. Kategorizácia žiakov do skupín bola robená na základe miesta bydliska nasledovne:

- skupina A – žiaci zo sídliska v meste
- skupina B- žiaci z mesta a okolitých dedín
- skupina C- žiaci z vidieka

**Graf 1: Podiel osobných stretnutí s osobou spoznanou len prostredníctvom internetu podľa miesta bydliska, ročníka a pohlavia žiakov**

Každá z troch sledovaných skupín predstavuje samostatný 100 %-ný celok. Zobrazenie na graf 1 umožňuje sledovať dynamiku podielu stretnutí podľa ročníkov sledovaním x-ovej osi.

Pre všetky sledované skupiny je spoločným znakom zintenzívnenie stretnutí s osobami spoznanými len z prostredia internetu v siedmom ročníku ZŠ. Výraznejšiu diferenciu v správaní sme zaznamenali u dievčat z 9. ročníka bývajúcich na vidieku v skupine C (vpravo), ktoré sa priznali ku skúsenosti z osobného stretnutia s virtuálnou známosťou spoznanou len z internetu s výskytom viac ako 50%. Tieto dievčatá v rozhovoroch pôsobili sebavedomo a mali pocit, že takéto stretnutia sú moderný spôsob nadväzovania nových známostí, v čom nechcú zaostávať. Ku týmto dievčatám sme prišli už neskoro, pretože neboli ochotné nás počúvať a zamyslieť sa nad možnými rizikami, ktorým sa vystavujú. Potenciálne nebezpečenstvá pri stretnutiach s virtuálnymi priateľmi musia žiaci spoznať skôr, ako sa začnú stretávať, čo je pre súčasnú generáciu žiakov približne siedmy ročník ZŠ.

Potrebné je žiakov čo najskôr utvrdzovať v tom, že keď začnú vnímať komunikáciu s virtuálnym priateľom ako nepríjemnú, obťažujúcu, cítia sa bezradní, bezmocní, nevedia ho zastaviť čím skôr ju treba ukončiť a prestať reagovať. Deti treba podporovať, aby s problémom nezostávali sami a obrátili sa na osobu, ktorej dôverujú. V prípade, že takúto osobu nemajú, využiť môžu napr. bezplatnú telefonickú Linku detskej istoty 116111 alebo môžu kontaktovať pomoc priamo prostredníctvom internetovej stránky POMOC.SK (Združená linka pomoci, ktorá vznikla združením Linky detskej istoty Unicef a linky pomoci projektu Zodpovedne.sk). Tejto problematike sa venuje stránka [www.zodpovedne.sk](http://www.zodpovedne.sk) neziskovej organizácie eSlovensko o.z..

## Záver

Kľúčové inštitúcie pre bezpečnosť, poriadok a zákonnosť sú polícia a ďalšie represívne zložky štátu ako napr. justícia, ktoré postihujú až dôsledky sociálne-patologických javov. Trvalejšia náprava, predchádzanie či zabránenie šírenia týchto javov by bolo účinnejšie, ak by sa zlepšilo vzdelávanie a osвета. V úvode článku

sme sa snažili naznačiť zložitosť procesov nevyhnutných pre riešenie incidentov, pri ktorých boli použité informačno-komunikačné technológie. Za posledné roky je zaznamenávaný nárast počítačovej kriminality, ktorý vyplýva z informatizácie spoločnosti ako zámerného cieľavedomého procesu vyspelých štátov sveta. Z tohto dôvodu nie je úmyslom obmedzovať používanie IKT, ale tvoriť systematické bezpečnostné opatrenia legislatívneho, technologického, aplikačného, preventívneho charakteru, ktoré by redukovali trestnú a inú protispoločenskú činnosť prostredníctvom IKT.

V článku sme naznačili potrebu venovať čas otvorenej diskusii o nebezpečenstvách pri komunikácii online. Pre žiakov je bezprostredným nebezpečenstvom sociálno-patologická online komunikácia s nevhodnými, obvykle staršími agresormi, ktorých identitu si nevedia overiť.

Príspevok vznikol v rámci riešenia vedecko-výskumnej úlohy Výsk. 161 Metódy spracovania policajne relevantných informácií na Akadémii Policajného zboru v Bratislave.

## 3. BIBLIOGRAFICKÉ ODKAZY

- [1] Štátny vzdelávací program INFORMATICKÁ VÝCHOVA. Príloha ISCED 1
- [2] Štátny vzdelávací program INFORMATIKA. Príloha ISCED 2
- [3] Štátny vzdelávací program INFORMATIKA. Príloha ISCED 3
- [4] Prevencia kriminality a inej protispoločenskej činnosti v Slovenskej republike na roky 2012-2015
- [5] Hinduja, S., Patchin, J. W. (2009). Bullying beyond the Schoolyard: Preventing and Responding to Cyberbullying. Thousand Oaks, CA: Sage Publications
- [6] Zákon č. 300/2005 Z. z. Trestný zákon v znení neskorších predpisov Zákon č. 583/2008 Z. z. o prevencii kriminality a inej protispoločenskej činnosti a o zmene a doplnení niektorých zákonov
- [7] Habich, L. Průběžný výskum problematiky dětské kybernetické kriminality mezi dětmi z pohledu Policie ČR in

DIDINFO 2015, Univerzita Mateja Bela, Banská Bystrica, 2015.

Právo-Bezpečnost-Informace, Praha: PA ČR, [online].  
Dostupné na: <  
<http://www.teorieib.cz/pbi/?section=articles>> 4/2014, ISSN  
2336-3657

- [8] GREGUSSOVA, M., DROBNÝ, M. *Deti v sieti, Ako chrániť seba a naše deti na internete*. Bratislava : eSlovensko, 2013. 108 s. ISBN 978-80-970676-6-3.
- [9] SMAHEL, D., HELSPER, E., GREEN, L., KALMUS, V., BLINKA, L., ÓLAFSSON, K. Excessive Internet Use Among European Children [online] [vytvorené november 2012] dostupné na  
<[http://www.lse.ac.uk/media%40lse/research/EUKidsOnline/EU%20Kids %20III/Reports/ExcessiveUse.pdf](http://www.lse.ac.uk/media%40lse/research/EUKidsOnline/EU%20Kids%20III/Reports/ExcessiveUse.pdf). 9 p. ISSN  
2045 – 256X>

*Krátke prednášky*

*Short Papers*

# Odvodenie vzorca pre obsah elipsy a jeho overenie metódou Monte Carlo v Exceli na gymnáziu

Ján Beňačka  
KI FPV UKF Nitra  
Tr. A. Hlinku 1  
949 74 Nitra  
Slovenská republika  
jbenacka@ukf.sk

## ABSTRAKT

Článok pojednáva o experimente, v ktorom bola žiakmi gymnázia sformulovaná hypotéza o obsahu elipsy a potom overená metódou Monte Carlo. Vzorec pre obsah bol odvodený z analógie medzi vzorcami pre obsah kruhu, štvorca a obdĺžnika. Monte Carlo experiment bol vykonaný s použitím aplikácie v Exceli, ktorá bola vyvinutá na hodine. Na záver hodiny žiaci vyplnili dotazník. Účelom výskumu bolo preveriť, či modelovanie v tabuľkovom kalkulátore môže urobiť vyučovanie zaujímavým, a či je prínosom pre metodiku vyučovania témy a pre technologické zručnosti a matematické vedomosti žiakov.

## ABSTRACT

The paper gives an account of an experiment in which a hypothesis on the area of ellipse was formulated and then verified by Monte Carlo method by gymnasium students. The area formula was deduced from the analogy between the formulas for the area of circle, square and rectangle. The Monte Carlo experiment was carried out by using a spreadsheet application developed in the lesson. The students answered a questionnaire. The aim of the survey was to find out whether spreadsheet modelling can make lessons interesting and contributing to the mathematical and technological knowledge of the students.

## Kľúčové slová

elipsa, model, štatistické metódy, tabuľkový kalkulátor

## Keywords

ellipse, model, statistical methods, spreadsheets

## 1. ÚVOD

Stredoškólači sa s náhodnými procesmi stretávajú v rámci počtu pravdepodobnosti a štatistiky [1]. Účelom je, aby si žiaci uvedomili, že i pri náhodných procesoch existujú isté ukazovatele, ktoré referujú o očakávateľných, ale nikdy nie istých možnostiach nastania daného javu. Ako príklady náhodných veličín slúžia rôzne dáta, napr. výška žiakov a pod. Ako príklady náhodných javov slúžia rôzne spôsoby vylosovania čísla alebo výberu guľôčky danej farby z osudia a pod. Generátor náhodných čísel je pri štúdiu týchto procesov užitočnou pomôckou.

V článku je uvedená aplikácia, ktorá prezentuje pre laika udivujúcu metódu výpočtu obsahu plošného útvaru náhodnými číslami. Výpočet obsahu plošného útvaru je praktická úloha a je v pozornosti matematikov od staroveku. Integrálny počet umožňuje dospieť k výsledku, ak poznáme primitívnu funkciu k hraničnej krivke útvaru. To nie vždy platí. V takom prípade sa úloha rieši inými metódami. Jednou z nich je metóda Monte Carlo. Nutnou podmienkou použitia metódy Monte Carlo je generátor náhodných čísel, buď fyzický, ako je hádzanie mincou alebo ihlou (známe Buffonove problémy [2]), alebo počítačový [3]. Tabuľkový kalkulátor umožňuje generovať náhodné čísla

jednoduchým spôsobom. K použitiu tabuľkového kalkulátora vo vyučovaní na vyššom strednom stupni viď. napr. [3 – 7].

## 2. TEÓRIA

Útvary, ktorého obsah chceme vypočítať opíšeme obdĺžnik so stranami rovnobežnými so súradnicovými osami. Nech pre súradnice  $[x; y]$  bodov obdĺžnika platí, že  $x \in \langle p, q \rangle$  a  $y \in \langle r, s \rangle$ . Obsah  $S$  obdĺžnika je

$$S = (q - p)(s - r). \quad (1)$$

Generujeme  $n$  náhodných bodov obdĺžnika tak, že generujeme  $n$  dvojíc  $[x; y]$ ,  $x \in \langle p, q \rangle$  a  $y \in \langle r, s \rangle$  (modré body na obrázku 1).

Nech  $m$  z nich sú aj bodmi útvaru, ktorého obsah treba vypočítať (červené body na obrázku 1). Pre obsah  $U$  útvaru potom platí

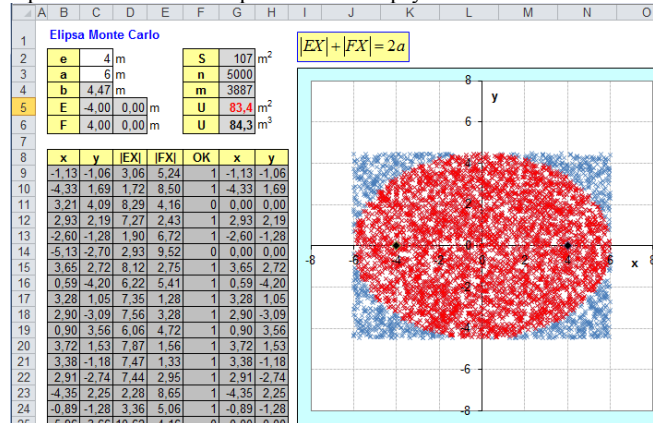
$$U = S \frac{m}{n}. \quad (2)$$

Nech útvárom je elipsa so stredom  $S[0;0]$ , excentricitou  $e$ , ohniskami  $E[-e;0]$ ,  $F[e;0]$  a dĺžkou hlavnej polosi  $a$ . Bod  $P$  je bodom elipsy alebo jej vnútra, ak

$$|EP| + |FP| \leq 2a. \quad (3)$$

## 3. APLIKÁCIA V EXCELI

Aplikácia na obrázku 1 počíta obsah elipsy metódou Monte Carlo.



Obrázok 1. Určenie obsahu elipsy metódou Monte Carlo

Biele bunky obsahujú vstupy, šedé bunky obsahujú vzorce. Excentricita je v bunke C2. Dĺžka hlavnej polosi je v bunke C3. Dĺžka vedľajšej polosi je v bunke C4. Súradnice ohnisk sú v bunkách C5:D6. Obsah opísaného obdĺžnika je v bunke G2 vypočítaný rovnicou (1). Je generovaných 5000 (bunka G3)

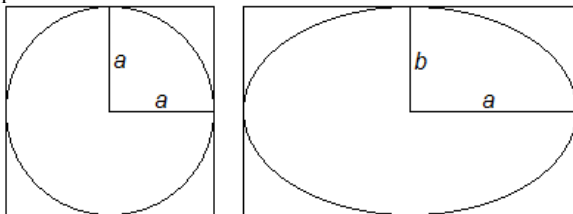
náhodných bodov. Počet bodov, ktoré vyhovujú nerovnici (3) je v bunke G4. Obsah elipsy vypočítaný rovnicou (2) je v bunke G5. Obsah vypočítaný vzorcom  $S = \pi ab$  je v bunke G6. Súradnice  $x$  a  $y$  bodov sú v bunkách B9:C5008. Vzdialenosť bodov od ohnisk  $E$  a  $F$  je v bunkách D9:E5008. Číslo 0 alebo 1 v oblasti F9:F5008 indikuje, že bod nespĺňa alebo spĺňa nerovnosť (3). Bunky G9:H5009 obsahujú súradnice bodov, ktoré spĺňajú nerovnosť (3), inak obsahujú nuly. Modrý graf (bodový graf typu XY) je zostrojený nad oblasťou B9:C5008. Červený graf je zostrojený nad oblasťou G9:H5009.

#### 4. VYUČOVACIA HODINA

Cieľom 90 minútovej vyučovacej hodiny bolo oboznámiť sa s elipsou a s princípom metódy Monte Carlo. Dôvodom pre voľbu elipsy bolo, že pre výpočet jej obsahu existuje jednoduchý vzorec  $S = \pi ab$ , s ktorým žiaci zvyčajne oboznámení nie sú, i keď ho je možné ľahko odvodiť (viď. [8, s. 128]). Žiaci sa ho týmto spôsobom naučia. Metódou Monte Carlo sa dá vzorec ľahko overiť. Ďalším dôvodom pre voľbu elipsy bolo, že jej „záhradnícka“ konštrukcia je jednoduchá a definícia elipsy z nej priamo vyplýva:

- zatlčieme dva kolíky  $E$  a  $F$ , ktoré predstavujú ohniská ;
- na kolíky uviažeme špagát dlhší ako je vzdialenosť kolíkov;
- tretím kolíkom napínáme špagát a kreslíme do zeme krivku smerom doprava a doľava v oboch polrovinách daných priamkou  $EF$ ;
- vyznačíme hlavné vrcholy  $A$  a  $B$ , stred  $S$  a vedľajšie vrcholy  $C$  a  $D$ ;
- definujeme hlavnú polos  $a$ , vedľajšiu polos  $b$  a excentricitu  $e$ ;
- v pozícii, keď je tretí kolík v hlavnom vrchole, ukážeme pomocou špagátu, že  $|EX| + |FX| = 2a$ , kde bod  $X$  je hrot tretieho kolíka, t. j. bod elipsy;
- v pozícii, keď je tretí kolík vo vedľajšom vrchole, ukážeme pomocou špagátu, že  $a^2 = e^2 - b^2$ .

Učiteľ začal hodinu diskusiou o probléme výpočtu obsahu útvarov. Na šikmom reze valca ukázal elipsu. S kusom špagátu predviedol na tabuľu s pomocou dvoch žiakov záhradnícku konštrukciu elipsy a definoval jej prvky. Žiaci na otázky odvodili nerovnosť (3). Učiteľ premietol cez dataprojektor obrázok 2. Na základe analógie medzi obsahom kruhu a opísaného štvorca a elipsy a opísaného obdĺžnika žiaci vyslovili hypotézu, že obsah elipsy je daný vzorcom  $S = \pi ab$ . Učiteľ stanovil za cieľ hodiny overiť platnosť uvedeného vzorca štatistickou metódou.



Štvorec:  $S = (2a)^2 = 4 \cdot a \cdot a$       Obdĺžnik:  $S = (2a) \cdot (2b) = 4 \cdot a \cdot b$   
 Kruh:  $S = \pi a^2 = \pi \cdot a \cdot a$       Elipsa:  $S = \pi \cdot a \cdot b$

Obrázok 2. Vytvorenie hypotézy o obsahu elipsy podľa analógie s obsahom štvorca a kruhu

Učiteľ predviedol cez dataprojektor aplikáciu na obrázku 1. Oboznámil žiakov s princípom metódy Monte Carlo. Potrebu

počtu 5000 generovaných bodov odôvodnil faktom, že pre dosiahnutie dostatočne presného riešenia je nutné, aby opísaný obdĺžnik bol čo najhustejšie pokrytý bodmi. Algoritmus riešenia bol prediskutoval so žiakmi formou otázok a odpovedí. Následne sa pristúpilo k tvorbe aplikácie. Účastníci si stiahli šablónu aplikácie z autorovej stránky (aplikácia na obrázku 1 bez grafu a vzorcov) a s podporou učiteľa aplikáciu vytvorili. Na záver vyplnili dotazník.

#### 5. VÝSKUM

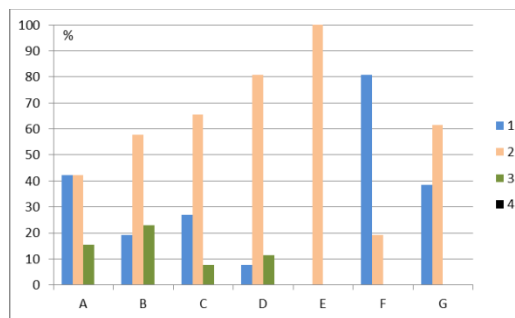
Výskum bol vykonaný s dvomi skupinami žiakov posledného ročníka Gymnázia Šurany v počte 14 a 12 na voliteľnom predmete Aplikovaná informatika. Dotazník obsahoval nasledovné otázky:

- A) *Hodina bola zaujímavá* (1 = veľmi; 2 = dosť; 3 = málo; 4 = nie)  
 B) *Rozumel som* (1 = všetko; 2 = väčšinu; 3 = málo; 4 = nič)  
 C) *Naučil som sa z Excelu* (1 = veľa; 2 = dosť; 3 = málo; 4 = nič)  
 D) *Naučil som sa z matematiky* (1 = veľa; 2 = dosť; 3 = málo; 4 = nič)  
 E) *O metóde Monte Carlo som vedel už predtým* (1 = áno; 2 = nie)  
 F) *Chcem modelovať Poker s kockami* (1 = áno; 2 = nie)  
 G) *Som muž* (1 = áno; 2 = nie)

Počet, aritmetický priemer, smerodajná odchýlka a variačný koeficient odpovedí sú uvedené v tabuľke 1. Relatívny počet v % odpovedí je zobrazený na grafoch 1, 2 a 3.

Tabuľka 1. Počet, aritmetický priemer (AP), smerodajná odchýlka (SO) a variačný koeficient (VK v %) odpovedí žiakov gymnázia

	1	2	3	4	AP	OS	VK
A	11	11	4	0	1,7	0,71	41
B	5	15	6	0	2,0	0,65	32
C	7	17	2	0	1,8	0,56	31
D	2	21	3	0	2,0	0,44	21
E	0	26			2,0	0,00	0
F	21	5			1,2	0,39	33
G	10	16			1,6	0,49	30

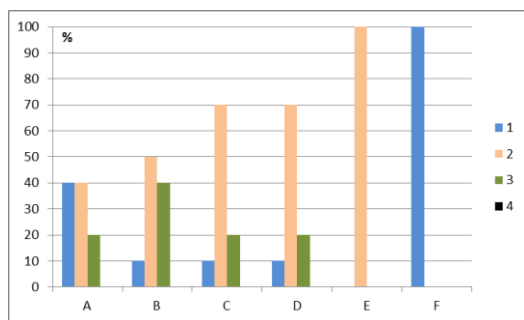


Graf 1. Relatívny počet odpovedí žiakov gymnázia

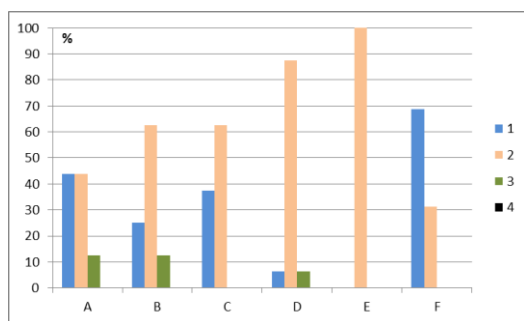
Výsledok je nasledovný: (A) 42 % žiakov považuje hodinu za veľmi zaujímavú, 43 % za dosť zaujímavú a 15 % za málo zaujímavú; (B) 19 % žiakov rozumelo všetko, 58 % väčšinu a 23 % málo; (C) 27 % žiakov má pocit, že sa z Excelu naučili nového veľa, 65 % dosť a 8 % málo; (D) 8 % žiakov sa má pocit, že sa z matematiky naučili veľa, 80 % dosť a 12 % málo; (E) 100 % žiakov o metóde Monte Carlo pred hodinou nevedelo; (F)



81 % chce modelovať v Exceli poker s kockami a 19 % nie; (G) 38 % boli chlapci a 62 % boli dievčatá. Hodina zaujala viac dievčatá, ale Poker s kockami chceli modelovať viac chlapci.



Graf 2. Relatívny počet odpovedí chlapcov



Graf 3. Relatívny počet odpovedí dievčat

Percento kladných odpovedí je nasledovné:

- A) Hodina bola zaujímavá: 85 %
- B) Rozumel som: 77 %
- C) Naučil som sa z Excelu: 92 %
- D) Naučil som sa z matematiky: 88 %
- E) O metóde Monte Carlo som vedel už predtým: 0 %
- F) Chcem modelovať poker s kockami: 81 %
- G) Som muž: 38 %

Výsledok nasvedčuje, že modelovanie náhodných procesov s tabuľkovým kalkulátorom má potenciál v propagácii matematiky a informatiky na úrovni gymnázia, môže urobiť

vyučovacie hodiny zaujímavými a je prínosom pre matematické vedomosti a technologické zručnosti žiakov.

## 6. BLIOGRAFICKÉ ODKAZY

- [1] CALDA, E., DUPAČ, V. *Matematika pro gymnázia - Kombinatorika, pravděpodobnost, statistika*, 5. vyd. Praha: Prometheus, 2008. ISBN 978-80-7196-365-3.
- [2] Buffon's needle. In *Wikipedia, the free encyclopedia* [cit. 2014-12-13]. Dostupné z: <[http://en.wikipedia.org/wiki/Buffon%27s\\_needle](http://en.wikipedia.org/wiki/Buffon%27s_needle)>.
- [3] BEŇAČKA, J., ČERETKOVÁ, S. Excel modelling in upper secondary mathematics – a few tips for learning functions and calculus. In: UBUZ, B., HASER, Ç., Mariotti, M.A. (Eds.). *CERME 8: Eighth Congress of European Research in Mathematics Education*, Manavgat-Side, Antalya, Turkey, 6-10 February 2013: conference proceedings. Ankara, Turkey: Middle East Technical University, 2013, s. 970-979. ISBN 978-975-429-315-9.
- [4] NEURATH, R.A., STEPHENS, L.J. The effect of using Microsoft Excel in a high school algebra class. *International Journal of Mathematical Education in Science and Technology*. 2006, 37(6), 721–756. ISSN 0020–739X.
- [5] FORSTER, P.A. Technologies for teaching and learning trend in bivariate data. *International Journal of Mathematical Education in Science and Technology*. 2007, 38(2), 143–161. ISSN 0020–739X.
- [6] BEŇAČKA, J. Introduction to 3D graphics through Excel. *Informatics in Education* [online]. 2013, 12(2), 221-230 [cit. 2014-06-22]. ISSN 1648-5831. Dostupné z: <[http://www.mii.lt/informatics\\_in\\_education/pdf/INFE226.pdf](http://www.mii.lt/informatics_in_education/pdf/INFE226.pdf)>
- [7] SIVASUBRAMANIAM, P. Distributed cognition, computers and the interpretation of graphs. *Research in Mathematics Education*. 2000, 2(1), 169-190. ISSN 1479-4802.
- [8] VANÍČEK, J. *Počítačové kognitivní technologie ve výuce geometrie*. Praha: Karlova univerzita, Pedagogická fakulta, 2009. ISBN 978-80-7290-394-8.



# Proces zmeny vzdelávacích metód vplyvom učebných štýlov

Anikó Töröková

Ekonomická univerzita v Bratislave  
Fakulta podnikového manažmentu  
Katedra informačného manažmentu  
Dolnozemska cesta 1/b  
852 35 Bratislava  
Slovenská republika  
aniko.torokova@euba.sk

Vladimír Bolek

Ekonomická univerzita v Bratislave  
Fakulta podnikového manažmentu  
Katedra informačného manažmentu  
Dolnozemska cesta 1/b  
852 35 Bratislava  
Slovenská republika  
vladimir.bolek@euba.sk

## ABSTRAKT

Proces vzdelávania, ktorý ovplyvňujú viaceré interné a externé faktory, sa v súčasnosti opiera o moderné vzdelávacie formy a metódy. Medzi aktuálne problémy, na základe ktorých sa tieto formy a metódy aplikujú, je aj štýl učenia sa. Adekvátnosť a vhodnosť aplikovaných vyučovacích metód na základe identifikovania učebných štýlov študentov má priamy vplyv na kvalitu formálneho, ale i neformálneho vzdelávania.

Príspevok sa zameriava na komparáciu definícií učebných štýlov domácich i zahraničných autorov. Vymedzuje hlavné znaky kognitívneho a učebného štýlu na základe charakteru, počtu dimenzií, viazanosti obsahu, zamerania a vnímania. Príspevok prezentuje aj skúsenosti autorov so zmenou vyučovacích metód na základe zistených štýlov učenia sa študentov prvého ročníka prvého stupňa Fakulty podnikového manažmentu Ekonomickej univerzity v Bratislave.

## ABSTRACT

Learning process is influenced by numerous external and internal factors. Currently based on modern education forms and methods. These forms and methods are applied on the basis of the student's learning style. Adequacy and appropriateness of applied teaching methods based on identifying learning styles of students has a direct impact on the quality of formal as well as informal education.

This paper focuses on the comparison of the definitions of learning styles domestic and foreign authors. It sets out the main features of cognitive and learning style based on the nature of dimensions, content commitment, focus and perception. The paper presents the authors' experience with the change of teaching methods based on the identified learning styles of students in the first year of First Level of the Faculty of Business Management, University of Economics in Bratislava.

## Kľúčové slová

učebný štýl, kognitívny štýl, metódy vzdelávania, formy vzdelávania

## Keywords

learning style, cognitive style, learning methods, forms of education

## 1. ÚVOD

Digitálna gramotnosť patrí medzi základné gramotnosti v súčasnom svete, ktorý sa čoraz viac prepája, globalizuje, kybernetizuje, virtualizuje, digitalizuje a jeho komplexita rastie. Zamestnávateľia vyžadujú, aby absolventi škôl boli digitálne

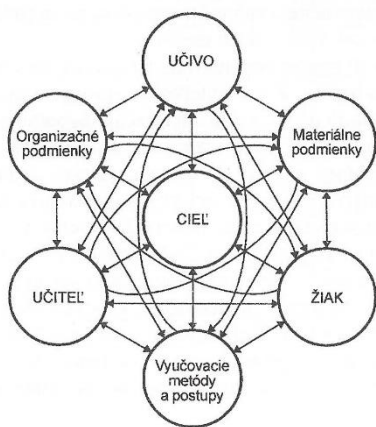
gramotní, vedeli využívať technológie informačnej spoločnosti. Stotožňujeme sa s názorom Koklesa a Romanovej, ktorí konštatujú, že „nielen Európa, ale aj Slovensko, trpia nedostatkom profesionálnych zručností v oblasti IKT a deficitom digitálnej gramotnosti, v dôsledku čoho, je veľa občanov vylúčených z digitálnej spoločnosti i hospodárstva. Dôraz, ktorý je kladený na zastúpenie IKT vo vzdelávacích systémoch, hlavne formálneho charakteru, sa ďalej premieňa do úrovne digitálnej gramotnosti obyvateľov“. [5]

Digitálnu gramotnosť ako jednu z kľúčových kompetencií definuje Odporúčanie Európskeho parlamentu a Rady z 18. decembra 2006 o kľúčových kompetenciách pre celoživotné vzdelávanie (2006/962/ES).

Katedra informačného manažmentu Fakulty podnikového manažmentu Ekonomickej univerzity v Bratislave zabezpečuje výučbu viacerých predmetov spadajúcich do disciplín manažmentu, informačného manažmentu a informatiky v troch stupňoch vysokoškolského štúdia. Poskytuje študentom najnovšie poznatky s možnosťou ich konfrontácie s praxou. Študenti všetkých študijných odborov fakulty v prvom semestri štúdia absolvujú predmet Informatika I, kde získajú ucelený a komplexný prehľad o teórii, rozvoji a súvislostiach informačných technológií, informačných a komunikačných systémov, základné poznatky z oblasti hardvéru, softvéru, komunikačných technológií, podnikovej informatiky, informačnej bezpečnosti a legislatívy, informačnej spoločnosti. V rámci cvičení získavajú pokročilé znalosti a zručnosti v práci s textovým editorom a tabuľkovým kalkulátorom s aplikáciou na riešenie komplexných ekonomických úloh. Študenti prichádzajú na fakultu z rôznych typov stredných škôl, najčastejšie sú z gymnázií a obchodných akadémií. Úroveň ich znalostí a zručností práce s aplikačnými softvérmi je rozdielna. Zväčša študenti majú laxný prístup k Informatike, čo sa často prejaví aj na ich študijných výsledkoch.

Za účelom zvyšovania kvality poskytovaného vzdelávania sme sa rozhodli analyzovať učebné štýly študentov a modifikovať vyučovacie metódy v tomto predmete. Zmenu sme videli v nových vyučovacích metódach, ktoré sme sa rozhodli modifikovať na základe preferovaných učebných štýlov študentov.

Z modelu podľa Hendricha, ktorý analyzoval Petlák [9] možno dedukovať, že celý výchovno-vzdelávací proces, ktorý má viesť k cieľu, je podmienený viacerými faktormi, vplyvujúcimi na samotný proces, ale aj jeho cieľ. Medzi tieto faktory patria: učivo, organizačné podmienky, materiálne podmienky, učiteľ, žiak, vyučovacie metódy, postupy a stanovený cieľ.



Obrázok 1. Výchovno-vzdelávací proces [9]

Cvičenia z predmetu Informatika I prebiehajú 1-krát týždenne v rozsahu 2 vyučovacích hodín (organizačné podmienky) v počítačových učebniach na počítačoch s nainštalovaným aktuálnym operačným systémom (MS Windows 8.1) a kancelárskym balíkom (MS Office 2013). Všetky počítače sú pripojené na intranet a internet. V každej učebni je k dispozícii pevne upevnený dataprojektor (materiálne podmienky).

## 2. CIEĽ PRÍSPEVKU

Problematika učebných štýlov študentov vysokých škôl je u nás málo rozpracovaná a skúmaná. Vychádzajúc z excerptie poznatkov domácej i zahraničnej literatúry a viacerých prieskumov, sme stanovili hlavný cieľ, ktorým je identifikovanie učebného štýlu študentov I. ročníka I. stupňa Fakulty podnikového manažmentu, pre potreby zmeny vyučovacích metód. Poznaním učebných štýlov študentov a aplikáciou vhodných vyučovacích metód možno zvýšiť kvalitu poskytovaného vzdelávania.

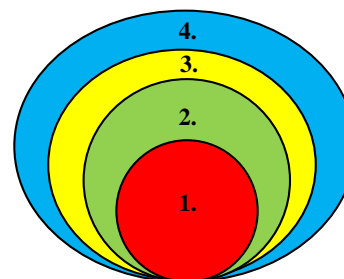
Hlavný cieľ naplňajú viaceré parciálne ciele:

- ❖ analýza dostupných literárnych prameňov zameraných na učebné štýly,
- ❖ zostavenie dostatočne validného a reliabilného dotazníka pre uskutočnenie dotazníkového prieskumu,
- ❖ vyhodnotenie dotazníkového prieskumu,
- ❖ syntéza získaných poznatkov,
- ❖ harmonizácia výstupu.

## 3. UČEBNÉ ŠTÝLY

Definíciou učebných štýlov sa zaoberajú viacerí autori. Mareš definuje štýly učenia ako „svojbytné postupy pri učení (svojbytné svojou štruktúrou, kvalitou, pružnosťou aplikácie), ktoré majú charakter metastrategie učenia. Vznikajú na vrozenom základe (kognitívne štýly) a rozvíjajú sa spolupôsobením vnútorných a vonkajších vplyvov. Dajú sa ovplyvniť a meniť, avšak nie ľahko a rýchlo“. [7] Mareš tvrdí, že kognitívne procesy hrajú iba rolu určitého mediátora učenia. Procesy sa stávajú prostriedkom učenia. Sprostredkujú jeho priebeh (môžu ho uľahčovať alebo komplikovať), sú aktivizované v priebehu celého procesu učenia sa a ovplyvňujú kvalitu jeho výsledku. Nie sú človeku dané v nemennej podobe a učením sa môžu do určitej miery zdokonaľovať. [7] Ako tvrdí Turek, pojem učebný štýl býva spájaný alebo dokonca aj nahrádzaný pojmom kognitívny štýl. Jedna skupina autorov tieto pojmy považuje za synonymá, iná považuje kognitívny štýl za nadradený učebnému štýlu, ďalšia zase naopak tvrdí, že kognitívny štýl je súčasťou učebného štýlu. [14]

Prúcha a kolektív autorov definuje učebný štýl ako „postupy pri učení sa, ktoré jedinec používa v určitom období svojho života vo väčšine situácií pedagogického typu. Sú do istej miery nezávislé na obsahu učenia sa. Vznikajú na vrozenom základe (kognitívny štýl) a rozvíjajú sa spolupôsobením vnútorných a vonkajších vplyvov“. [10]. Učebný štýl možno charakterizovať aj ako svojbytné postupy. Toto ponímanie uvádzajú autorky Müllerová, Šikulová. „Učebné štýly vznikajú na vrozenom základe, majú charakter metastrategie, pri učení sa uplatňujú v určitom období života, čiastočne závisia na učive a teda i na predmete, vedú k výsledkom určitého typu (zapamätanie vedomostí alebo porozumenie so systémom), rozvíjajú sa spolupôsobením vonkajších podmienok, dajú sa do určitej miery ovplyvňovať a meniť“. [8] Ponímanie učebného štýlu podľa Curryovej [1] je vo forme vrstevníc, ktoré predstavujú jednotlivé zložky učebného štýlu.



Obrázok 2. Model učebného štýlu [1]

- 1. Vrozený základ (kognitívny štýl)** - ťažko ovplyvniteľný pedagogickým pôsobením (typ inteligencie, zmyslové preferencie...).
- 2. Spôsob spracovania informácií** - čiastočne ovplyvniteľný Metódy: kritické myslenie, projekty, riešenie problémov, kladenie otázok, učenie objavovaním.
- 3. Sociálne a emocionálne procesy** - ovplyvniteľné sociálnym prostredím. Metódy: školská klíma, medziľudské vzťahy, spolupráca s rodinou, systém podpory...
- 4. Preferencie výučby** - ovplyvniteľné pedagogickým pôsobením. Metódy: projekty, kooperatívne učenie, tímová práca, riadená diskusia, dramatizácia...

Ak analyzujeme klasický cibulový model a jeho hierarchické usporiadanie, konštatujeme, že: jadro predstavuje (1. – červená zóna) rigidný základ, ktorý je pevný a veľmi ťažko ovplyvniteľný vplyvom pôsobenia okolia jedinca. Ďalšie vrstvy otvárajú priestor pre ovplyvniteľnosť jedinca väzbami, metódami a pôsobením okolia na neho. Najvýznamnejšiu zmenu môžu vykonať metódy pedagogického pôsobenia: obsah učiva, metódy, postupy, formy a stratégie, ktoré preferuje jedinec. Z pohľadu na model je zrejmé, že jednotlivé vrstvy sú previazané a existujú medzi nimi vzájomné vzťahy a väzby.

Ak komparujeme viaceré výpovede domácich i zahraničných autorov, nachádzame v nich korporatívne znaky definície učebných štýlov, ale i disparity, predovšetkým v ponímaní kognitívneho štýlu ako súčasti učebného štýlu. Kognitívny štýl možno definovať ako charakteristické spôsoby vnímania, zapamätávania si informácií, myslenia, riešenia problémov, rozhodovania. Sú prejavom individuálnych rozdielov v tom, ako ľudia organizujú a riadia svoje informácie a skúsenosti. [7] Patria do kategórie dispozícií, sú z väčšej časti vrozené, málo ovplyvniteľné, či meniteľné. [10] Z výskumu Simutha vyplýva, že „kognitívny štýl je jednou z premenných, ktorá ovplyvňuje rozhodnutie pre štúdium na vysokej škole a tým aj v mnohých prípadoch kariérnu cestu. Výber vysokej školy má zároveň vplyv na mieru prípravy študentov k

podnikateľstvu. Z charakteru štúdia, predmetov a spôsobu výučby sa dá predpokladať, že najmä študenti manažmentu budú vedení k rozvoju podnikateľského ducha a k podnikaniu. Výsledky výskumu prispievajú k poznaniu, že je vhodné podporovať študentov pri výbere štúdia a kariérnej cesty aj na základe ich osobnostnej charakteristiky – kognitívneho štýlu tak, aby boli so štúdiom, ale najmä budúcou kariérou spokojní a výkonní“. [13]

Hlavné znaky medzi kognitívnym štýlom a učebným štýlom, ktoré vymedzila Hrubíšková, možno zaradiť do nami navrhnutých oblastí:

**Tabuľka 1. Znaký kognitívneho a učebného štýlu [4]**

Oblasť	Kognitívny štýl	Učebný štýl
Charakter	prevažne vrodený	prevažne získaný
Počet dimenzií	2	> 2
Viazanosť obsahu	minimálna	vyššia
Zameranie	vnímanie, myslenie, pamäť, riešenie problémov, rozhodovanie	osobná motivácia, riešenie problémov, autoregulácia, sociálny kontext
Vnímanie	spontánne	spontánne prechádza do vedomého

Stotožňujeme sa s charakteristikou učebných štýlov, ktorú prezentuje Rovňanová: „učebný štýl je súhrn postupov, ktoré jednotlivec v určitom období preferuje pri učení sa. Vyvíja sa z vrodeného základu, ale v priebehu života sa mení a zdokonaľuje. Učebný štýl pomáha jednotlivcovi dosahovať dobré výsledky pri učení sa určitého typu učiva v určitej edukačnej situácii, ale môže mu komplikovať úspešnosť pri učení sa iného typu učiva v inej edukačnej situácii. Zvyčajne si svoj učebný štýl neuvedomuje, systematicky ho neanalyzuje a nehľadá jeho zlepšenie. Učebné štýly sa často jedincovi zdajú samozrejmé. Dajú sa však diagnostikovať a v prípade potreby aj meniť, ale to je náročnejší proces, ktorý si vyžaduje osobnú disciplínu a podporu, pomoc rodičov, spolužiakov i ďalších odborníkov.“ [11]

Na definíciu učebných štýlov priamo nadväzuje ich klasifikácia. Existuje mnoho delení učebných štýlov. Kolb segmentoval jednotlivcov podľa ich preferencií na divergéro, akomodátorov, asimilátorov a konvergéro. [6] Najväčšia prednosť konvergentného učebného štýlu je v riešení problémov, rozhodovaní a praktických aplikáciách vlastných predstáv. Jedinci s týmto štýlom bývajú najlepší v takých situáciách, ako sú bežné inteligenčné testy, v ktorých sa vyžaduje jednoznačné správne riešenie problému. Konvergéri uprednostňujú skôr riešenie technických úloh a problémov než sociálnych situácií. Divergentný učebný štýl má opačné zameranie než konvergentný. Najväčšou prednosťou tejto orientácie je predstavivosť a uvedomenie si významov a hodnôt. Predstavitelia tohto typu myslenia dosahujú vyššie výkony v situáciách, ktoré vyžadujú formulovanie alternatívnych myšlienok, predstavivosť a orientáciu na vlastné prežívanie. Divergéri majú širšie kultúrne záujmy a často sa špecializujú na umenie. Asimilátori uprednostňujú abstraktné pojmy a uvážlivé pozorovanie. Najväčšia výhoda tejto orientácie je v induktívnom myslení, v schopnosti vytvárať teoretické modely a v prispôbovaní rôznorodých pozorovaní do integrovaného celku. Tento učebný štýl sa uplatňuje skôr u jednotlivcov pôsobiach v prírodných vedách a matematike, ako v aplikovaných odvetviach. Jednotlivci s týmto učebným štýlom sa častejšie presadzujú pri výskumných a plánovacích aktivitách. Akomodačný učebný štýl má prednosť v zdôrazňovaní konkrétnych skúseností a v aktívnom experimentovaní. Najväčšia výhoda tejto orientácie je v realistickom vykonávaní

naplánovaných úloh, pričom sa tolerujú aj nové informácie. Adaptívne pôsobenie tejto orientácie je v hľadaní príležitostí riskovaní a v aktívnej činnosti. Ako vyplýva z názvu štýlu, najlepšie sa uplatňuje v tých situáciách, v ktorých sa jednotlivec musí prispôbovať okamžitej zmene situácie. [12]

Pre potreby nášho prieskumu s možnosťou aplikácie v podmienkach vysokej školy sme využili klasifikáciu autorov Honey a Mumford, ktorí definujú štyri skupiny: [3]

**1. Aktivisti** majú radi činnosť, radi sa učia, sú vnímaví, prijímajú názory zo všetkých strán. Nadšene spoznávajú všetko nové, radi vysvetľujú, rozhodujú a vedú ostatných. Majú nižšiu schopnosť sústredenia sa, sú menej dôslední a majú tendenciu konať bez rozmýšľania, bez ohľadu na následky. Pri rovnakej činnosti a pri dlhodobých úlohách strácajú nadšenie a začínajú sa nudiť. Najlepšie im vyhovuje skupinová práca, hry, praktické činnosti, prípadové štúdie, problémové a projektové vyučovanie.

**2. Reflektori** radi rozmýšľajú – ochotne prijímajú, dôsledne zvažujú a prehodnocujú názory ostatných. Sú tolerantní a často súhlasia s väčšinovým názorom, aj keď ich názor je iný. Vyznačujú sa tým, že pri riešení úloh zhromažďujú informácie z rôznych zdrojov. Neradi sú v centre pozornosti, keď majú napríklad prezentovať výsledky skupiny, neradi prežívajú situácie, keď je potrebné reagovať rýchlo, bez prípravy či neplánovane a robiť unáhlené závery alebo rozhodnutia. Najvhodnejším spôsobom vyučovania je pre nich názorné vysvetľovanie, čítanie, sledovanie videa, opakovanie naučeného, pozorovanie spolužiakov pri učení.

**3. Teoretici** uprednostňujú teóriu a fakty. Prijímajú a spracovávajú informácie analyticky-synteticky. Problémy študujú, posudzujú, riešia postupne. Pracujú dôkladne a dlho, používajú fungujúce princípy, postupy, teoretické modely a systémy. Neradi riešia úlohy, pri ktorých musia vyjadrovať emócie a pocity, prípadne príliš jednoduché úlohy. Neradi tolerujú spolupracovníkov v skupine, ktorí sú na nižšej intelektuálnej úrovni ako oni. Najlepšie sa učia prostredníctvom diskusií, rozhovorov a prednášok, ktoré obsahujú abstraktné informácie. Vyhovuje im individuálna samostatná práca, výskum, spracovávanie štúdií.

**4. Pragmatici** radi získané informácie ihneď využívajú v praxi. S nadšením skúmajú a skúšajú nové myšlienky, teórie a techniky. Prekypujú novými nápadmi a myšlienkami a snažia sa o ich priamu aplikáciu. Pracujú rýchlo a sebaisto. Odmietajú nevyužitelné teórie a tiež zbytočné, neefektívne informácie či postupy. Pri vyučovaní im najviac vyhovujú prípadové štúdie, príklady z praxe, praktické cvičenia, projekty, exkurzie, riešenie problémových úloh, hranie rolí.

## 4. METÓDY A METODIKA SKÚMANIA

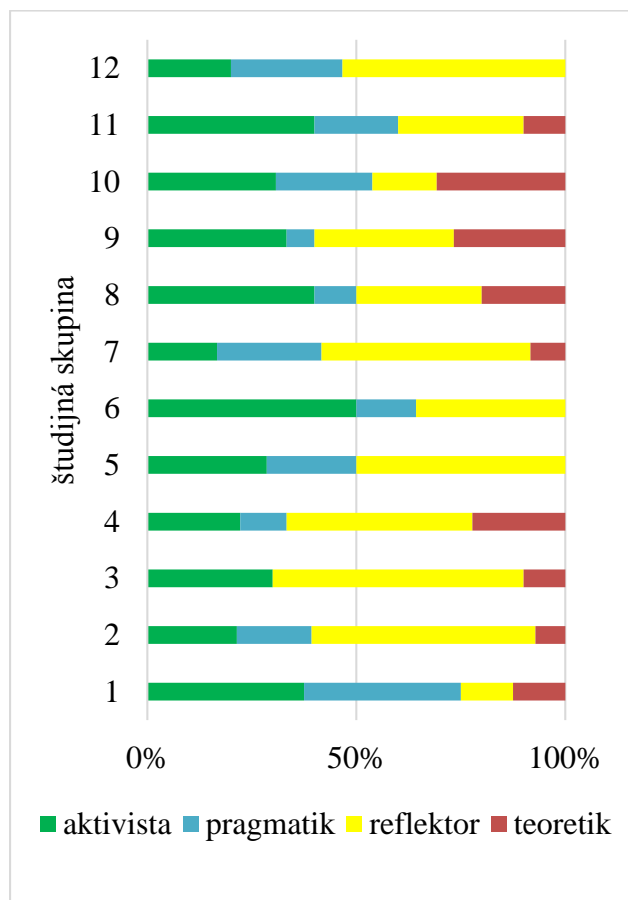
V príspevku bol využitý kvalitatívny výskum, ktorý sme ponímali ako postupy založené na „kvalitatívnej“, resp. „neštatistickej“ analýze, hoci sme ju kombinovali s kvantitatívnymi údajmi a postupmi, napríklad na overenie záverov vyplývajúcich z kvalitatívnej analýzy, či môže sama ilustrovať, alebo vyjašňovať kvantifikované výsledky výskumu. Vyhranené oddelovanie kvantitatívneho a kvalitatívneho výskumu pokladáme za prekonané. Ich spoločným cieľom je zabezpečiť najvyššiu kvalitu výskumu. Fundamentálnym základom analýzy bol dotazníkový prieskum. Pri tomto výskumnom nástroji je veľmi dôležité sledovať jeho validitu a reliabilitu. [2] Pri popise dotazníka sme podrobne špecifikovali proces zostavovania a jednotlivé východiská, vďaka čomu bola odôvodnená konštruktová, obsahová i kritériová validita. Reliabilita, teda presnosť a spoľahlivosť dotazníka, je determinovaná tromi faktormi: početnosťou položiek, homogenitou a zložitnosťou úloh. Každý výskumný nástroj môže byť do určitej miery presný a spoľahlivý, ale žiadny nástroj

nedosahuje dokonalú presnosť a spoľahlivosť. Okrem metódy dotazníkového prieskumu sme aplikovali aj metódu sémantického diferenciálu. Ide o výskumnú metódu, ktorá je zameraná na zisťovanie toho, ako respondenti vnímajú pojmy. Každý pojem má okrem denotatívneho významu aj význam konotatívny. Aplikáciou vyššie uvedených metód sme dosiahli naplnenie stanovených cieľov a formulovanie relevantných záverov aplikovateľných v teoretickej aj praktickej oblasti.

Prieskum učebných štýlov sme uskutočnili medzi študentmi prvého ročníka prvého stupňa Fakulty podnikového manažmentu Ekonomickej univerzity v Bratislave. Do prieskumu bolo zapojených 158 študentov dvoch odborov: Ekonomika a manažment podniku a Finančný manažment podniku. Dotazník pozostával z 80 tvrdení, pri ktorých respondenti vyjadrili svoj súhlas. Žiaduce bolo, aby respondenti odpovedali spontánne bez akejkoľvek prípravy. Zber dát bol vykonaný prostredníctvom dátového úložiska, cloudu. Dáta boli analyzované a vyhodnotené podľa metodiky autorov Honey a Mumford, ktorí definovali tieto skupiny učebných štýlov. U respondentov sme identifikovali učebný štýl na základe danej stupnice získaného celkového skóre kladných odpovedí.

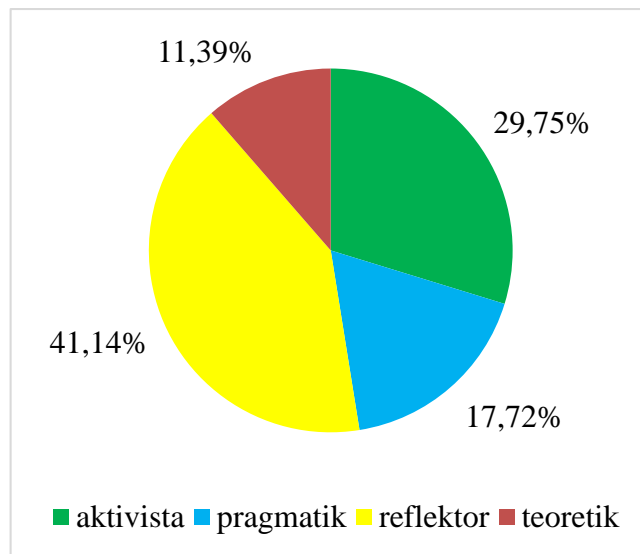
## 5. VÝSLEDKY PRÁCE A DISKUSIA

Prieskumu sa zúčastnili študenti dvanástich študijných skupín. Z grafu možno vidieť, že v každej skupine sú zastúpení študenti s učebným štýlom reflektor, aktivista. Učebný štýl teoretik sa v troch študijných skupinách nevyskytoval. Najviac zastúpený, viac ako 41%, bol učebný štýl reflektor. Pragmatický učebný štýl sa nevyskytoval iba v jednej študijnej skupine.



Graf 1. Učebné štýly v študijných skupinách

Agregované výsledky prieskumu, ktoré boli zohľadnené pri procese modifikácie vyučovacích metód uvádzame v grafe č. 2. Ako vyplýva zo súhrnného grafu, najviac študentov preferuje učebný štýl typu: reflektor – 41%. Títo študenti uprednostňujú názorné vysvetlenie učiva, opakovanie naučeného. Druhou veľkou skupinou sú aktivisti – 30% uprednostňujúci prípadové štúdie, problémové vyučovanie. Pragmatický učebný štýl preferuje 18% študentov. Tento učebný štýl študenti ťažšie aplikujú nakoľko okamžitá aplikácia získaných znalostí a zručností v hospodárskej praxi je obmedzená, praktické zadania sa modelujú na príkladoch z hospodárskej praxe. Najmenej zastúpeným učebným štýlom u študentov je teoretický učebný štýl, ktorý vyžaduje individuálnu prácu študenta. Tento fakt sa potvrdil aj z poznatkov z praxe.



Graf 2. Učebné štýly študentov

Na základe uskutočneného dotazníkového prieskumu možno prijať tento výsledok: študenti I. ročníka I. stupňa Fakulty podnikového manažmentu Ekonomickej univerzity v Bratislave ochotne prijímajú, dôsledne zvažujú a prehodnocujú názory ostatných. Sú tolerantní a často súhlasia s väčšinovým názorom, aj keď ich názor je iný. Vyznačujú sa tým, že pri riešení úloh zhromažďujú informácie z rôznych zdrojov. Neradi sú v centre pozornosti, keď majú napríklad prezentovať výsledky skupiny, neradi prežívajú situácie, keď je potrebné reagovať rýchlo, bez prípravy, či neplánovane a robiť unáhlené závery alebo rozhodnutia.

Najvhodnejším spôsobom vyučovania je pre týchto študentov názorné vysvetľovanie, čítanie, sledovanie videa, opakovanie naučeného, pozorovanie spolužiakov pri učení. Vhodná je priama modifikácia na prípadové štúdie, skupinovú prácu a príklady z hospodárskej praxe. Optimálna je práve kombinácia vyučovacích metód vyplývajúcich z učebných štýlov reflektor a aktivista.

Proces modifikácie vzdelávacích metód vplyvom učebných štýlov bol rozdelený do 3 krokov. 1. krok analýza učebných štýlov formou dotazníkového prieskumu, 2. krok modifikácia predmetu Informatika I. a 3. krok evaluácia. Vychádzajúc z prvého kroku analýzy učebných štýlov študentov a prijatia výsledkov z prieskumu sme do vyučovania predmetu Informatika I. v I. ročníku v I. stupni zaradili praktické modelové úlohy, problémové úlohy, zadania komplexného charakteru – využitie získaných poznatkov a aj riešenie problémov. V súčasnosti využívame skupinovú aj individuálnu prácu študentov. Problémové úlohy boli uvedené spoločným hľadaním riešení (metóda brainstormingu), za účelom

generovania kreatívnej stratégie. Priame uplatnenie brainstormingu prehlbuje u študentov znalosť uplatniteľnosti tejto metódy v praxi. Je jednou v súčasnosti z najvyužívanejších metód pri manažerskom rozhodovaní, čo nadväzuje aj na ich odbornú profiláciu. Uvedením vyššie spomenutých vyučovacích metód sme prispôsobili aj skriptá pre študentov tohto ročníka. Pri získaní spätnej väzby a vykonaní evaluácie sme sledovali úspešnosť študentov pri zápočtových písomkách a dosiahnutie vyššieho skóre pri riešení problémových úloh. Modifikáciou vzdelávacích metód v procese vyučovania sme dosiahli o 7,80% vyššiu úspešnosť pri riešení zápočtových zadanií. Komparácia výsledkov bola pred a po zmene učebných štýlov.

## 6. ZÁVER

Znalosť učebných štýlov prispieva k vytvoreniu podmienok na osvojenie kľúčovej kompetencie naučiť sa učiť. Učebné štýly sú jedným z významných faktorov ovplyvňujúce výchovno-vzdelávací proces. Príspevok komparuje definície učebných štýlov od domácich i zahraničných autorov. Vymedzuje hlavné znaky medzi kognitívnym a učebným štýlom na základe charakteru, počtu dimenzií, viazanosti obsahu, zamerania a vnímania.

Uplatnením vhodných vedeckých metód a metodiky sme naplnili parciálne ciele, ktoré vyústili do splnenia hlavného cieľa, a to identifikácie učebných štýlov študentov I. ročníka I. stupňa Fakulty podnikového manažmentu Ekonomickej univerzity v Bratislave. Z výsledkov prieskumu vyplynulo, že študenti prichádzajú zo stredných škôl s učebným štýlom typu reflektor a aktivista. Na základe týchto zistení sme vykonali modifikácie vyučovacích metód, aby zodpovedali učebným štýlom študentov za účelom zvyšovania kvality poskytovaného vzdelávania. Jednoznačne možno konštatovať, že študenti na univerzitu už prichádzajú s takýmito učebnými štýlmi, čo v prvom ročníku pedagógovia nedokážu ovplyvniť. Autori sa do budúcnosti rozhodli, že je žiaduce, aby pokračovali v prieskume a identifikovali, či sa jednotlivé učebné štýly menia počas štúdia na univerzite.

Tento príspevok bol podporený Vedeckou agentúrou VEGA prostredníctvom finančnej podpory projektu č. 1/0336/14.

## 7. BLIOGRAFICKÉ ODKAZY

- [1] CURRY, L. *Learnings styles in secondary schools – An Review of Instrument and Implications for their Use*. Madison: University of Wisconsin, 1990.
- [2] GAVORA, P. a kol. *Elektronická učebnica pedagogického výskumu*. [online] Bratislava : Univerzita Komenského, 2010. Dostupné na internete: <http://www.e-metodologia.fedu.uniba.sk/index.php/kapitoly/ziskanie-hodnotnych-dat/validita.php?id=i9p1> [cit. 2015. 01. 08.] ISBN 978–80–223–2951–4

- [3] HONEY, P., MUMFORD, A. *The manual of learning styles*. Berkshire, UK: Maidenhead Honey Press, 1986.
- [4] HRUBIŠKOVÁ, H. *Diagnostika učebných štýlov žiakov*. [online]. Bratislava : UK v Bratislava, 2000. [cit. 2015-01-30] Dostupné na: [www.hrubiskova.8u.cz/diagnostika\\_ucebne\\_styly\\_st.ppt](http://www.hrubiskova.8u.cz/diagnostika_ucebne_styly_st.ppt)
- [5] KOKLES, M., ROMANOVÁ, A. Komparácia výučby informatiky na bakalárskom stupni štúdia v študijnom odbore Ekonomika a manažment podniku. In *Aktuálne trendy vo výučbe informatiky pre manažérov : zborník príspevkov z metodicko-pedagogického seminára pri príležitosti 60. výročia založenia Fakulty podnikového manažmentu [elektronický zdroj]*. - Bratislava : Vydavateľstvo EKONÓM, 2013. 9 s. ISBN 978-80-225-3703-2
- [6] KOLB, D. *Experiential learning*. New Jersey : Englewood Cliffs, Prentice Hall, 1984.
- [7] MAREŠ, J. *Styly učení žáků a studentů*. Praha : Portál, 1998. 239 s. ISBN 80-7178-246-7
- [8] MULLEROVÁ, L., ŠIKULOVÁ, R. *Cvičebnice obecné didaktiky pro studenty učitelství*. Ústí nad Labem : UJEP, 2001.
- [9] PETLÁK, E. *Pedagogicko-didaktická práca učiteľa*. Bratislava : Iris, 2000. 118 s. ISBN 80-89018-05-X
- [10] PRŮCHA, J., WALTEROVÁ, E., MAREŠ, J. *Pedagogický slovník*. Praha : Portál, 1995.
- [11] ROVNANOVÁ, L. O učebných štýloch žiakov. [online]. Banská Bystrica : Katedra pedagogiky, 2000. 4 s. [cit. 2015-01-19] Dostupné na: <http://www.pdf.umb.sk/~lrovnanova/O%20u%C4%8Debn%C3%BDch%20C5%A1t%C3%BDl%20och%20C5%BEiakov.pdf>
- [12] RUISEL, I. *Od kognitívnych štýlov k mentálnej sebaregulácii*. [online] Bratislava : SAV, 2006. [cit. 2015-01-20] Dostupné: <http://www.psychologia.sav.sk/zvladanie.pdf#page=77>
- [13] SIMUTH. J. *Education toward entrepreneurship. Does cognitive style preference influence the career choice?* [online] Bratislava : VSM, 2014. [cit. 2015-01-20]. Dostupné na: [http://www.cutn.sk/Library/proceedings/mch\\_2014/editovane\\_prispevky/30.%20C5%A0im%C3%BAt.pdf](http://www.cutn.sk/Library/proceedings/mch_2014/editovane_prispevky/30.%20C5%A0im%C3%BAt.pdf)
- [14] TUREK, I. *Didaktika*. Bratislava : Iura, 2010. 598 s. ISBN 9788080783228



# Interaktívne testovanie žiakov v prostredí Samsung School a Socrative

Mgr. Róbert Borbély, PhD.  
ZŠ, Trieda SNP 20, Banská Bystrica  
Trieda SNP 20  
974 01 Banská Bystrica  
Slovensko  
robo.borbely@gmail.com

Mgr. Gabriel Stach  
ZŠ, Trieda SNP 20, Banská Bystrica  
Trieda SNP 20  
974 01 Banská Bystrica  
Slovensko  
gabriel.stach@gmail.com

## ABSTRAKT

Príspevok sa orientuje na využitie moderných informačno-komunikačných technológií pri testovaní žiakov v edukačnej činnosti školy. Syntetizuje a analyzuje inovatívne možnosti tvorby interaktívnych testov v prostredí aplikácií Samsung School a Socrative na tabletových zariadeniach. Poukazuje na prednosti automatického spracovania a vyhodnotenia testov – okamžitú spätnú väzbu, objektivnosť, úsporu času, atraktivnosť či interaktívny charakter. Súčasťou je aj stručná komparácia s prehľadom funkcií a vlastností oboch systémov, ktoré možno vzájomne výhodne integrovať.

## ABSTRACT

The contribution focuses on the use of modern information and communication technologies for testing of students in the educational activities of the school. It synthesizes and analyzes the innovative possibilities of creating interactive tests in the application environment Samsung School and Socrative on tablet devices. It points out the advantages of automatic processing and evaluation of tests-immediate feedback, objectivity, time-saving, attractiveness and interactive character. A brief comparison with an overview of the functions and features of both systems is also included. They can be mutually integrated profitably.

## Kľúčové slová

Testovanie, tablet a škola, dotykové technológie a škola, Samsung School, Socrative.

## Keywords

Testing, tablets and school, touch technology and school, Samsung School, Socrative.

## 1. ÚVOD – TABLETY V EDUKAČNOM PROSTREDÍ

Súčasným trendom vo výchovno-vzdelávacej činnosti je masívnejšie uplatňovanie tabletových zariadení a ďalších dotykových technológií. Ukazuje sa, že tieto prostriedky majú značný didaktický potenciál a umožňujú kreatívnejšiu a názornejšiu výučbu. Žiaci jednoduchým

dotykom vyhľadávajú informácie na internete, pracujú s učivom v digitálnej podobe, hľadajú súvislosti, spájajú, priradujú, učia sa. Zároveň nadobúdajú najnovšie zručnosti v používaní špičkových technológií, ktoré budú môcť využiť v ďalšom štúdiu a neskôr i v práci.

Tablety sú tiež ideálnym nástrojom na testovanie a overovanie znalostí a zručností žiakov vo výučbe. Práve spätná väzba je jednou z najdôležitejších častí práce každého pedagóga, od ktorej neraz závisia budúce kroky v napredovaní kvality procesov vyučovania a učenia (sa). K nesporným výhodám využívania moderných technológií na testovanie patria: zvýšená motivácia učiacich sa, individualizácia vyučovania, objektivnosť, efektívna spätná väzba či úspora času učiteľskej prípravy. Umožňujú okamžité vyhodnocovanie práce žiakov, ukladanie a archivovanie výsledkov či ich analýzu. Súčasne sa pritom programovo eliminujú akékoľvek subjektívne faktory pri vyhodnocovaní, čím sa zvyšuje dôveryhodnosť výsledkov. Žiaci postupujú vlastným tempom a uplatňujú individuálny prístup. Komunikácia učiteľ – žiak je bezpochyby interaktívnejšia, dynamickejšia a atraktívnejšia.

Virtuálny svet dnes ponúka množstvo možností pre uľahčenie práce učiteľa. Už v minulosti boli dostupné viaceré elektronické testovacie prostriedky pre aplikáciu v školskom prostredí (napríklad: eLearning, Moodle, Google dokumenty, a i.). Momentálne sa ale do popredia dostáva populárne testovanie prostredníctvom aplikácie Socrative a prostredia Samsung School. Stalo sa tak najmä vďaka zavádzaniu dvoch inovatívnych projektov – Škola na dotyk (Nezisková organizácia EDULAB a spoločnosť Samsung) a Elektronizácia vzdelávacieho systému regionálneho školstva – Digiškola (Ministerstvo školstva, vedy, výskumu a športu SR). I keď oba nástroje sledujú takmer totožné ciele, možno v ich funkcionalitách, aplikačných možnostiach, vo výsledných výstupoch či ich analýze a hodnotení, pozorovať niekoľko osobitostí. Jednotlivé potenciality, vrátane praktického využitia, sa teraz pokúsime stručne analyzovať a syntetizovať.

## 2. TESTOVANIE V PROSTREDÍ SAMSUNG SCHOOL

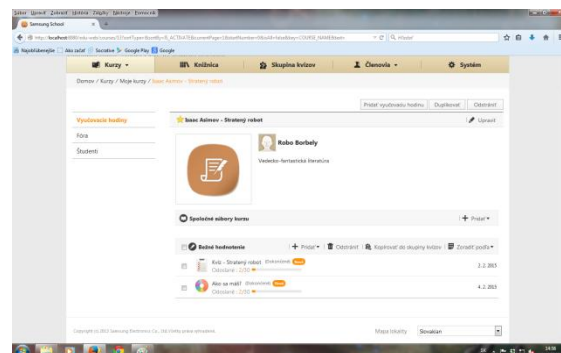
Aplikácia Samsung School je komplexným riešením tabletovej učebne, ktorá vytvára interaktívnu spoluprácu medzi učiteľom a žiakmi v reálnom čase. Umožňuje



efektívne prepájať digitálne sety a tablety, kontrolovať a riadiť edukačný proces, sprístupniť individuálny obsah, zdieľať obrazovky tabletu s interaktívnou tabuľou, spravovať obsah žiackych tabletov, či ich zablokovať, efektívne a rýchlo testovať pre získanie spätnej väzby. Práve posledná možnosť vytvára priestor pre realizáciu vyučovacích hodín s dôrazom na diagnostiku žiackych výkonov. Súčasne dáva učiteľovi možnosť vyučovať akékoľvek učivo atraktívnym spôsobom a mať pritom plne pod kontrolou prácu žiakov na hodinách a aktívne vytvárať digitálny obsah.

Po prihlásení sa do aplikácie máme možnosť vytvoriť si nový kurz, ktorý možno vnímať ako predmet alebo konkrétnu vyučovaciu hodinu. Tento naplníme vlastným obsahom alebo zdieľaním obsahu s ostatnými kolegami v rámci verejných priečinkov či knižnice súborov. V časti bežné hodnotenie sa dostávame k tvorbe overenia žiackych vedomostí formou kvízu, ankety, či učebného zadania (úlohy). Kvíz môže slúžiť ako okamžitá spätná väzba (písomná práca/bleskovka/test). V povinných poliach zadávame názov kvízu, pokyny pre žiakov, časový limit, ako aj jeho bodovú hodnotu (rozráta sa automaticky). Systém nám výhodne ponúka vytváranie otázok s viacerými uzavretými možnosťami odpovede, s odpoveďami pravda/nepravda alebo zadaním možnosti voľnej odpovede (vrátane jej konkrétneho riešenia). Rozsah banky s výberom možností (distraktorov) nie je obmedzený. Otázky sa dokonca dajú pridávať už aj z existujúcej databázy kvízov v rámci školy alebo z vlastnej skupiny kvízov. Názov kvízu tak slúži nielen na jeho identifikáciu, ale tiež ako kľúčové slovo pri vyhľadávaní v rámci ostatných kurzov. Pomenovanie je takisto povinnou súčasťou každej otázky, čo umožňuje jej zaradenie do určitej témy alebo oblasti v obsahu vyučovania, resp. na už spomínané vyhľadávanie. K jednotlivým odpoveďiam a otázkam možno pre ilustráciu a názornosť pripájať obrázky, ktoré vedú mať aj charakter samotnej odpovede. Zaujímavým prvkom je voľba komentára, ktorá sa žiakovi zobrazí a slúži na nevyhnutné dokreslenie učiva, jeho zaradenie do ďalších súvislostí pri kľúčovej systematizácii poznatkov. Rovnako sem patria prípadné kritériá hodnotenia danej otázky/úlohy, čo zvyšuje transparentnosť a objektivnosť celého procesu. V triede dokážeme ďalej realizovať hlasovanie s rovnakými možnosťami ako pri kvíze prostredníctvom anketových otázok. Prirodzene, bez existencie správnej odpovede, čím zaistíme získanie subjektívneho názoru a pohľadu zapojených žiakov na určitý problém. Po kompletizácii testovacích zadaní realizujeme testovanie vo vybranom nástroji. Učiteľ (administrátor) zaraďuje všetkých alebo vybraných žiakov na realizáciu príslušného kurzu, následne spúšťa samotný kurz. Nevýhodou je, že žiaci sa musia prihlasovať na konkrétnu vyučovaciu hodinu z výberu viacerých kurzov, nespustí sa teda automaticky. Pri existencii veľkého počtu vzdelávacích materiálov tak môže dochádzať k problémom s identifikáciou a následne k nežiaducim časovým stratám v rámci vyučovacej

jednotky. Účastníci testovania vyplňajú, hlasujú, riešia úlohy a otázky individuálnym pracovným tempom, s možnosťou návratu k predchádzajúcim zadaniam a ich následnej oprave nesprávnych riešení (v rozsahu prípadnej časovej dotácie). Učiteľ po celý čas monitoruje samostatnú žiacku prácu z miesta pomocou lokálneho počítača alebo prostredníctvom tabletu. V prípade, že žiaci nevykonávajú požadovanú činnosť, potrebujú pomocť alebo majú problémy, učiteľ s nimi dokáže individuálne komunikovať cez samotný systém (odosielanie správ, pečiatok, blokovanie tabletu). Dokonca môže riadiť aj to, ktoré vzdelávacie materiály (obrazové a textové materiály, aplikácie a i.) sa budú žiakom na tabletoch zobrazovať. Ak sú žiaci s testom hotoví, odošlú svoje odpovede jednoduchým zakliknutím na tlačidlo „odoslať“. Po skončení časového limitu sa aktuálny stav kvízu odošle automaticky. V prípade včasného vyriešenia úloh všetkými žiakmi, môže učiteľ testovanie (vyučovaciu hodinu) kedykoľvek ukončiť. Po ukončení testovania/hlasovania je možné zobraziť výsledky potrebné k jeho analýze. Nielen učiteľ, ale aj každý žiak vidí vlastné výsledky na svojom tablete, ako aj vlastnú pozíciu v skupine v prehľadnom koláčovom alebo stĺpcovom grafe. Správne a nesprávne odpovede sú jednoznačne farebne a graficky odlišené. Učiteľ vidí konkrétne odpovede jednotlivých žiakov po odkliknutí na graf alebo exportovaním výsledkov formou súboru v programe Excel ako jednoduchú tabuľku s identifikáciou žiaka, súčtom získaných bodov, ako aj s umiestnením v kontexte ostatných riešiteľov. Využitie testov je obmedzené len na prostredie aplikácie, nie je ich možné stiahnuť v bežne podporovaných formátoch. Výsledky testovania sa síce v systéme ukladajú, ale realizáciou rovnakého testu v inej skupine sa neželateľne prepisujú (pokiaľ nedôjde k duplikácii testov pre inú skupinu/iný kurz). Nespornými výhodami systému sú bezpochyby: hromadná spätná väzba v podobe on-line písomnej práce s výstupmi, pohodlný monitoring a manažment situácie v triede a žiackych tabletoch, časová nenáročnosť samotnej realizácie či zážitkový charakter práce.



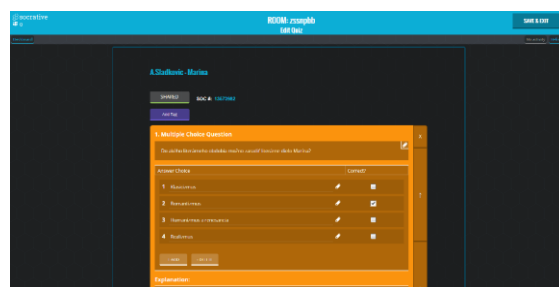
Obrázok 1. Samsung School

### 3. TESTOVANIE V PROSTREDÍ SOCRATIVE

Obdobným, no azda jednoduchším a rýchlejším, nástrojom na testovanie je aplikácia Socrative zameraná na tvorbu interaktívnych kvízov, testov, ankiet, hlasovaní a súťaží. Jej značnou výhodou je, že funguje nielen na tablete, ale aj na PC a notebooku. Pre plnú funkčnosť je potrebné stiahnuť si dva druhy voľne dostupných aplikácií – učiteľský a žiacky účet (Socrative Teacher a Socrative Student). Príprava testových on-line otázok je tu veľmi jednoduchá, nezaberie veľa času a je vhodná pre okamžité využitie na vyučovacej hodine. Možno ju prirovnať k bežnému vytvoreniu testu vo Word a jeho následnému kopírovaniu na papier. Súčasťou programu však nie je komplexný manažment sledovania a organizovania situácie na tabletoch v triede tak, ako to bolo pri predchádzajúcej aplikácii.

Učiteľ si v systéme zriaďuje svoju vlastnú miestnosť („room name“), žiaci sa nemusia osobitým spôsobom prihlasovať, postačuje, ak poznajú názov učiteľovej miestnosti. Správa testov poskytuje široké možnosti úpravy a realizácie vyučovacieho obsahu. V porovnaní s predchádzajúcim systémom Samsung School je tu orientácia intuitívnejšia a prehľadnejšia. Príprava kvízu neobmedzuje množstvom povinných polí, naopak ponúka rýchly vstup priamo do kompozície tela testu s okamžitou možnosťou realizácie. Hotové testy sú tu ale omnoho menej dostupné ako v predchádzajúcej aplikácii. Ich zdieľanie s ostatnými učiteľmi je možné len so súhlasom autora a disponovaním jeho jedinečného identifikátora v podobe číselného kódu. Toto možno vnímať ako veľkú nevýhodu, no na druhej strane je tu chránené autorské právo obmedzujúce voľné šírenie. V novom kvíze si učiteľ podobným spôsobom ako v predchádzajúcej aplikácii volí typy otázok s bankou správnych odpovedí, pravda/lož alebo krátkou odpoveďou. Pozoruhodnou je ale možnosť zaškrtnúť i viaceré správne odpovede k jednej otázke (zaškrťavacie políčka). Ku každej otázke/úlohe je možné doplniť vysvetlenie, prípadne určité návody na spôsob riešenia – ešte predtým než na ňu žiak odpovie. Toto nadobúda značný význam najmä pri kvízových otázkach s voľnou odpoveďou, kedy žiaka vopred upozorníme na formu/tvar správneho zápisu odpovede (napríklad pravopis: veľké a malé písmená, interpunkcia a podobne). Jednotlivé otázky je prirodzene možné neustále upravovať, dopĺňať, resp. odstrániť. Na získanie okamžitej spätnej väzby bez predchádzajúcej prípravy je k dispozícii funkcia „rýchla otázka“ (rýchly prieskum), kde žiakom len ústnym spôsobom zadávame otázku a rovnako ústne uvádzame možnosti súboru odpovedí alebo typu pravda/lož (resp. písomne na tabuľu). Žiaci tak odpovedajú hlasovacím spôsobom kedykoľvek v priebehu vyučovacej hodiny bez zdĺhavej predchádzajúcej prípravy. Zaujímavou súčasťou Socrativu je skupinová aktivita Space Race (preteky skupín). Žiaci pracujú na kvíze v kolektíve, body sa tak nezarátavajú študentom jednotlivo, ale celej skupine. Tieto možno vytvoriť vedomým výberom alebo celkom

náhodne. Na záver vyučovacej hodiny je vhodné ešte realizovať krátku výstupnú aktivitu s názvom Exit Ticket (výstupný lístok), ktorá slúži na získanie odozvy od žiakov na uskutočnenú vyučovaciu hodinu alebo aktivitu. Žiaci odpovedajú na už systémom predpripravené otázky (v anglickom jazyku) súvisiace s tým, či sa im vyučovacia hodina páčila/nepáčila, ako aj či dostatočne pochopili tému a preberanú učebnú látku. Pre všetky druhy testovania v aplikácii sú významným prvkom tri voliteľné nastavenia pravidiel testu. Prvou možnosťou je okamžitá spätná väzba pre žiakov o správnosti jednotlivých odpovedí a jej vysvetlenie, nemožnosť preskakovať otázky a meniť odpovede. Druhý spôsob nastavenia umožňuje preskakovať otázky, meniť odpovede až do odoslania alebo ukončenia kvízu učiteľom. V tretej možnosti kontroluje spúšťanie otázok učiteľ, ten dokáže každému študentovi automaticky zamiešať poradie otázok a distraktorov. Vďaka tomu sa dá predchádzať nežiaducej spolupráci žiakov a nutnosti vytvárať rozdielne skupiny testov (A/B a pod.). Vzhľadom na to, že pred samotným riešením testu musí každý žiak zadať svoje meno, má učiteľ priebežný prehľad o úplnosti a správnosti odpovedí prostredníctvom tabuľky (podobne v Samsung School prostredníctvom grafu). Vďaka tomu vidí, ktoré z otázok sú pre žiakov problematické a vyžadujú si ďalšiu pozornosť. Veľkým pozitívom aplikácie je aj možnosť prípravy testov v rovine off-line. Jednoduchým spôsobom si učiteľ stiahne šablónu v programe Excel, po jej vyplnení ju v okamihu použitia v režime on-line importuje, následne je test pripravený na použitie. V prípade potreby je možné každý vytvorený materiál exportovať vo formáte pdf. dokumentu. Všetky testy a ich prehľad odpovedí sú archivované a učiteľ sa k nim môže kedykoľvek a kdekoľvek dostať, príp. si ich môže vytlačiť alebo zobrazit' na interaktívnej tabuli. Výhodou takýchto interaktívnych materiálov a testov je, že učiteľ už nemusí prenášať z pracovného prostredia do domáceho rôzne papierové formy testov a cvičení, ktoré chce kontrolovať a následne vyhodnocovať. Všetko má totiž umiestnené na internete. Realizované kvízy s výsledkami sa trvalo archivujú (narozdiel od Samsung School sa neprepisujú) a ponúkajú rôzne možnosti zobrazenia a analýzy. Systém umožňuje odoslať výsledky priamo na e-mail učiteľa, stiahnuť na disk vo formáte tabuľky skupinovo alebo dokonca vo forme opraveného testu za každého žiaka individuálne, zdieľať cez Google Drive alebo prezerať v grafickom náhľade.



Obrázok 2. Socrative

**Tabuľka 1. Stručné porovnanie najdôležitejších vlastností a funkcií aplikácií Samsung School a Socrative**

<i>Vlastnosti /Funkcie:</i>	<i>Samsung School</i>
Jazyk	slovenský
Dostupnosť	platený
Pristup	väzba na lokálny webový server, preferovaná lokálna sieť s nutnosťou wifi rozhrania
Funkcia	komplexný triedny manažment, testovanie
Prihlasovanie	obmedzené na mobilné zariadenia
Režim off-line	nie
Užívateľské prostredie	zložité nastavovanie
Grafické rozhranie	rozšírené
Spustenie vyučovacej hodiny (žiak)	manuálne
Kvízy a hlasovania (testovanie)	<ul style="list-style-type: none"> <li>- bez možnosti viacerých správnych odpovedí v teste</li> <li>- s možnosťou priradovania obrázkov k odpovediam</li> <li>- možnosť priameho pridávania testových otázok z iných kvízov (aj od iných autorov)</li> <li>- bez možnosti meniť nastavenia realizácie testov (pevné poradie otázok a distraktorov)</li> <li>- možnosť okamžitej spätnej väzby po kompletnom vyplnení kvízu (správne a nesprávne odpovede, pozícia v skupine)</li> <li>- prístupnosť k zdieľaniu testov medzi pedagógmi</li> <li>- bez možnosti sťahovania zadania testu (bežne nepodporovaný formát)</li> </ul>
Analýza a export výsledkov testovania (štatistika)	<ul style="list-style-type: none"> <li>- bodovanie testu sa rozráta automaticky podľa maximálneho počtu bodov (uvedené na začiatku tvorby testu učiteľom)</li> <li>- základné výstupy (skupinové tabuľky a grafy)</li> <li>- výstupy v Exceli (skupinovo)</li> <li>- bez možnosti priameho zdieľania výstupov</li> <li>- krátkodobé archivovanie výsledkov (prepísovanie novými výsledkami)</li> </ul>
Monitorovanie triedy (manažment)	<ul style="list-style-type: none"> <li>- možnosť sledovať žiacku prácu (zdieľať obrazovky tabletov), komunikovať so žiakmi cez správy, blokovať žiacke obrazovky.</li> <li>- možnosť automatického spustenia stránok, aplikácií a materiálov</li> <li>- mená (prezývky) žiakom priraduje učiteľ</li> </ul>

<i>Vlastnosti /Funkcie:</i>	<i>Socrative</i>
Jazyk	anglický
Dostupnosť	zdarma
Pristup	klasické webové rozhranie, možnosť vstupu z globálnej siete
Funkcia	testovanie
Prihlasovanie	bez obmedzení zariadení (aj na PC)
Režim off-line	áno
Užívateľské prostredie	jednoduché nastavovanie
Grafické rozhranie	strohé
Spustenie vyučovacej hodiny (žiak)	automatické
Kvízy a hlasovania (testovanie)	<ul style="list-style-type: none"> <li>- s možnosťou viacerých správnych odpovedí v teste</li> <li>- bez možnosti priradovania obrázkov k odpovediam</li> <li>- bez možnosti priameho pridávania testových otázok z iných kvízov (len duplikovanie testov)</li> <li>- možnosť meniť nastavenia realizácie testov (náhodné poradie jednotlivých otázok a distraktorov)</li> <li>- možnosť výberu priebežnej spätnej väzby po každej otázke, spätnej väzby na konci kvízu (cez zdieľanie obrazovky učiteľa) alebo bez možnosti náhľadu spätnej väzby žiakmi</li> <li>- obmedzenosť k zdieľaniu testov medzi pedagógmi</li> <li>- možnosť sťahovania zadania testu (+ jeho tlač)</li> </ul>
Analýza a export výsledkov testovania (štatistika)	<ul style="list-style-type: none"> <li>- bodovanie testu prebieha automaticky bez vplyvu učiteľa (1 otázka = 1 bod)</li> <li>- detailné výstupy (skupinové a individuálne tabuľky, grafy a korektúry testov) vrátane percentuálneho vyjadrenia</li> <li>- výstupy v Exceli (skupinovo) a v PDF-readeri (individuálne)</li> <li>- možnosť priameho zdieľania výstupov cez e-mail, Google Drive, stiahnutím na disk</li> <li>- dlhodobé archivovanie výsledkov (limitované len odstránením učiteľom)</li> </ul>
Monitorovanie triedy (manažment)	<ul style="list-style-type: none"> <li>- bez možnosti priameho sledovania žiackych tabletov, bez priestoru na komunikáciu so žiakmi</li> <li>- bez možnosti spustenia sprievodných materiálov</li> <li>- mená (prezývky) si žiaci zadávajú sami, učiteľ len volí medzi možnosťou použitia konkrétnych mien alebo bez mena (anonymne)</li> </ul>

## 5. ZÁVER

Zavádzanie a aktívne používanie interaktívnych technológií v slovenských školách je plne v súlade so Štátnym vzdelávacím programom. Z vlastnej praktickej skúsenosti možno povedať, že samotní žiaci tablety preferujú omnoho viac ako tradičné učebné pomôcky či metódy. Sú pre nich pútavé, zodpovedajú ich záujmom, vzbudzujú aktivitu a prirodzenú motiváciu k učeniu. Urýchľujú prácu s informáciami, prostredníctvom rôznorodých aplikácií a ich nástrojov rozvíjajú žiacku tvorivosť a mediálne zručnosti. Ich ovládanie je pre nich intuitívne až automatické. Vyučovacia hodina už viac nie je založená len na jednotlivcovi pred tabuľou, ale na kolektíve žiakov, z ktorých každý hrá svoju rolu.

Na základe vymenovaných a v praxi realizovaných možností práce s testovacími nástrojmi v prostredí Samsung School a Socrative možno zhrnúť tieto spoločné overené prínosy – z pohľadu učiteľa a žiaka:

- a) učiteľ = nadobúdanie zručností a rozvoj kompetencií v rámci IKT, zavádzanie inovatívnych metód a foriem, on-line prístup k vytvoreným edukačným materiálom, archivácia materiálov a výsledkov žiackej práce, široká variabilita vytvárania testov, možnosť opakovaného použitia a rôznych úprav materiálov, automatické vyhodnotenie a prehľadný grafický súhrn riešení, objektívne skórovateľný vzdelávací obsah.

- b) žiak = rozvoj zručností a kľúčových kompetencií v rámci IKT, rozvoj počítačovej gramotnosti, motivácia a zážitkovosť inovatívnymi metódami a formami práce, vysoký stupeň interaktivity, možnosť vidieť a porovnať svoje výsledky s inými, sledovať individuálny a skupinový pokrok.

Ako produktívne sa ukazuje prepojenie oboch aplikácií a následné profitovanie z ich najlepších vlastností. Systém Samsung School umožňuje priame otváranie/zadávanie vybraných aplikácií pre prácu žiakov na vyučovacej hodine. Tu sa vytvára priestor pre odkaz/presmerovanie na aplikáciu Socrative a jej efektívnejšie využitie s obohatením o komplexný manažment a flexibilnejšiu organizáciu či monitoring triedy.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] *Užívateľská príručka Samsung School* [online]. Dostupné na: <[http://www.samsung.com/global/business/mobile/smartschool/download/20140109/SamsungSchool1.5\\_UM\\_Cze\\_Rev.1.2\\_131231.pdf](http://www.samsung.com/global/business/mobile/smartschool/download/20140109/SamsungSchool1.5_UM_Cze_Rev.1.2_131231.pdf)>
- [2] *Užívateľská príručka Socrative* [online]. Dostupné na: <<http://www.socrative.com/materials/SocrativeUserGuide.pdf>>
- [3] *Projekt Škola na dotyk v ZŠ, Trieda SNP 20, Banská Bystrica*. Dostupné na: <<http://zssnpbb.edupage.org/blog1/?>>

# Peer review pri hodnotení projektov

Veronika Dropčová

KZVI FMFI Uk v BA

Mlynská dolina

842 48 Bratislava

Slovenská republika

veronika.dropcova@fmph.uniba.sk

Zuzana Kubincová

KZVI FMFI Uk v BA

Mlynská dolina

842 48 Bratislava

Slovenská republika

zuzana.kubincova@fmph.uniba.sk

## ABSTRAKT

Pred tromi rokmi sme na kurze webdizajnu nasadili peer review (vzájomné recenzovanie) na aktivity súvisiace s písaním blogových článkov. V roku 2014 sme túto techniku použili pri recenzovaní projektu – implementácii vlastného blogu študenta. Vývoj projektov bol rozdelený do troch fáz a recenzovanie prebiehalo v každej z nich zvlášť. V každej fáze študenti najprv odoslali projekt na recenzovanie, potom recenzovali tri projekty svojich kolegov a následne mali možnosť opraviť si svoj projekt podľa pripomienok z recenzií. Až potom ho odoslali na hodnotenie učiteľovi. V tomto článku sa venujeme názorom študentov na takéto použitie techník peer review a ich názorom na zmenené podmienky pre odovzdávanie projektov.

## ABSTRACT

Three years ago, we employed peer review of blog articles in our web design course. Subsequently, in 2014 we used this method for reviewing of the course project – implementation of student's own blog. The project development was divided into three phases, each reviewed separately. In each phase, students were supposed to submit the project for reviewing, then to review three assigned projects of their colleagues and afterwards they were given a possibility to correct their own project according to the comments in reviews. Only after that they submitted their project for teacher's evaluation. In this article we focus on students' opinion of the use of these peer review techniques and the changed conditions for project submission.

## Kľúčové slová

peer review, vzájomné hodnotenie sa, vysoká škola, hodnotenie projektov

## Keywords

peer review, higher education, project assessment

## 1. ÚVOD

Metóda peer review sa vo vzdelávacích aktivitách používa už niekoľko rokov. Viaceré štúdie [1, 2, 3] potvrdzujú, že vzájomné hodnotenie výsledkov práce spolužiakov na nejakom predmete nielen pomáha študentom rozvinúť ich kritické a analytické myslenie, komunikačné zručnosti, konštruktívnu kritiku a pod., ale podporuje aj ich záujem o samotný predmet a jeho obsah.

Najčastejším spôsobom využitia peer review vo vyučovaní je vzájomné hodnotenie esejí [4] alebo blogových článkov [5, 6]. Študentské hodnotenia pritom môžu ale nemusia prispievať ku známke. Treba však brať do úvahy fakt, že študenti majú spravidla tendenciu dať spolužiakovi lepšie hodnotenie, ako im dáva učiteľ [4]. Aj napriek tomuto však možno na základe študentských hodnotení rozpoznať slabšie práce.

Otázkou formátu peer review sa zaoberal napr. MacAlpine [7]. Poukazuje na to, že kvalita študentských recenzií sa významne zvýši, ak majú predpísanú štruktúru a ak sú dobre špecifikované všetky aspekty, na ktoré sa majú študenti pri hodnotení sústrediť.

Študentské hodnotenie možno vo vyučovaní použiť aj v iných aktivitách, ako je písanie esejí alebo článkov. Gehringer [8,9,10] napr. aplikuje peer review aj pri tvorbe testových otázok, zbieraní zdrojov k učivu a pod. Jeho študenti nielen recenzujú prácu iných, ale hodnotia aj užitočnosť recenzií, ktoré dostali od spolužiakov na svoju prácu.

Niektoré z vyššie spomínaných prístupov sme v našom vyučovaní použili už pred viac ako dvomi rokmi aj my. Študenti bakalárskeho a magisterského štúdia aplikovanej informatiky recenzovali blogové príspevky svojich spolužiakov, pričom ich hodnotenie neovplyvňovalo hodnotenie daného študenta učiteľom. Na základe našich výskumov [11 – 15, a pod.] môžeme potvrdiť, že v súvislosti so zavedením peer review sa nielen významne zvýšila zapojenosť študentov do tejto vzdelávacej aktivity, ale sa nezanedbateľne zlepšili aj ich študijné výsledky.

Aj napriek vysokej účasti študentov v blogovaní a recenzovaní a napriek ich zlepšeným študijným výsledkom mnohí študenti prejavili nevôľu k týmto aktivitám, najmä k blogovaniu. Preto sme v ďalšom školskom roku trochu zmenili podmienky a organizáciu kurzu a študentov sme nechali navzájom si recenzovať projekty. V tomto príspevku uvádzame prvé výsledky z výskumu spojeného s touto aktivitou.

## 2. PEER REVIEW NA KURZE WEBDIZAJNU

Metódu vzájomného hodnotenia sme v roku 2014 použili na kurze webdizajnu, ktorý je zameraný na princípy návrhu webových stránok, ich štruktúry, obsahu a používateľského rozhrania s dôrazom na prístupnosť a použiteľnosť. Počet študentov absolvujúcich záverečnú skúšku bol 54.

Na rozdiel od predchádzajúcich rokov tento rok si študenti namiesto recenzovania blogových príspevkov navzájom recenzovali prácu na hlavnej aktivite kurzu – na projekte. Projekt je celosemestrálna hodnotená aktivita, zadaním ktorej je navrhnuť a vytvoriť svoj vlastný blog. Práca na projekte je rozdelená do troch fáz, pričom na konci každej fázy podlieha aktuálny stav projektu hodnoteniu učiteľa. Pre každú fázu projektu sú presne stanovené požiadavky na blog ako aj kritériá hodnotenia.

Každá z fáz vývoja projektu mala tento rok priradené tri za sebou idúce termíny: 1. termín odovzdávania projektu na recenzovanie; 2. termín odovzdávania recenzií; 3. termín odovzdávania projektu na hodnotenie učiteľom.

V každej fáze po odovzdaní projektov na recenzovanie boli všetky projekty rozdelené študentom na hodnotenie tak, aby každý študent recenzoval tri náhodne pridelené projekty a každý projekt bol recenzovaný tromi náhodne vybranými študentmi. Recenzovanie bolo obojstranne anonymné.

Študentské recenzie pozostávali z odpovedí na vopred stanovené otázky, ku ktorým recenzenti navyše prideliť aj body. Bodovú hodnotu vyberali z Lickertovej škály s rozsahom 1-5 (1-veľmi slabý, 5-výborný). Otázok bolo 6-8 a boli pre každú fázu jedinečné, nakoľko reflektovali na presne stanovené požiadavky na projekt v danej fáze a zodpovedali kritériám, podľa ktorých projekt neskôr hodnotili učitelia.

Medzi termínom odovzdania recenzií a termínom odovzdania projektu učiteľovi na hodnotenie aktuálnej fázy mali študenti nejaký čas (spravidla tri dni) na to, aby si mohli prečítať recenzie, ktoré dostali od kolegov a zapracovať pripomienky z nich do svojich projektov. Až potom odovzdali projekt na hodnotenie učiteľovi.

Učiteľské hodnotenie projektu, ktoré prispievalo ku konečnej známke študenta, nebolo ovplyvnené študentskými hodnoteniami.

Samotný projekt bol súčasťou kurzu Moderný prístup k webdizajnu už aj v minulosti (tento kurz funguje od roku 2006), no až v minulom roku sa odovzdávanie projektu rozdelilo do troch fáz. V tomto roku okrem samotného recenzovania projektu pribudla ešte i iná novinka – tzv. *dvojité kontrola* (Obr. 1). Prvú kontrolu si pri odovzdávaní projektu na recenzovanie zabezpečil sám študent, ktorý projekt odovzdával. Druhú kontrolu mu sprostredkovali študenti, ktorí jeho projekt recenzovali.

1. termín 2. termín 3. termín  
**Blog ==> Recenzie ==> Finál ==> Hodnotenie**

**Obrázok 1. Dvojité kontrola pri odovzdávaní projektov**

Naším cieľom pri zmene organizácie kurzu a najmä projektu bolo vyjsť v ústrety študentom a aplikovať metódu vzájomného hodnotenia takým spôsobom, aby sa nielen rozvíjali ich schopnosti analytického myslenia, kritického hodnotenia a jasného vyjadrovania, ale aby boli aj lepšie motivovaní ku štúdiu problematiky preberanej na kurze a k aktívnej účasti na jednotlivých jeho aktivitách. Zaujímalo nás preto, ako takto nastavenú organizáciu kurzu, projektu a hodnotenia vnímajú a čo si z nich dokázali odniesť. V nasledujúcej časti príspevku prinášame výsledky prieskumu, v ktorom sme sa pokúsili nájsť odpovede na predchádzajúce otázky.

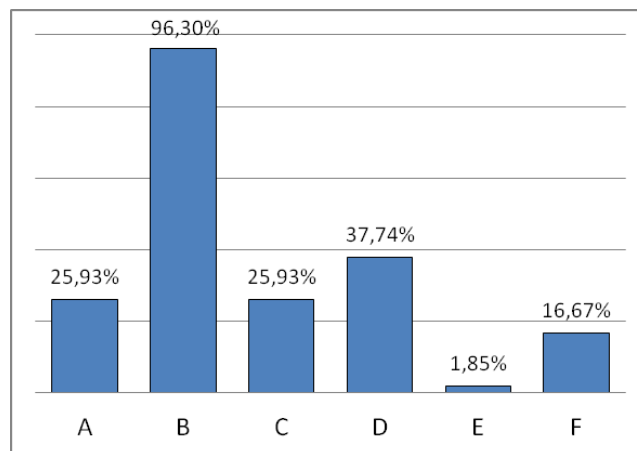
### 3. VÝSLEDKY VÝSKUMU

#### 3.1 Recenzovanie

Po ukončení semestra sme študentov požiadali o vyplnenie dotazníka, ktorého cieľom bolo o.i. zistiť ich spokojnosť a názory na takúto formu recenzií. Tejto téme sa v dotazníku venovali štyri otázky.

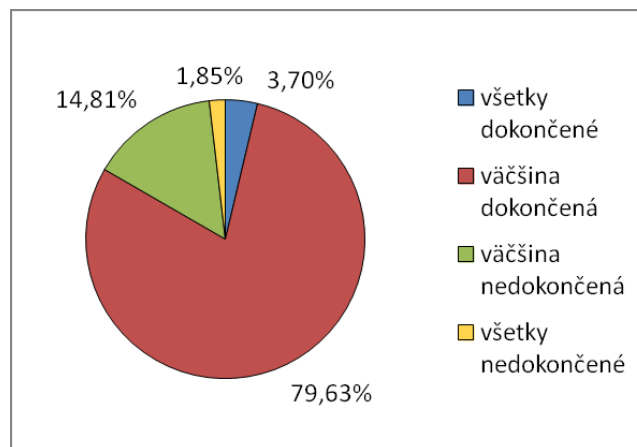
V prvej z nich sme sa študentov pýtali, *čím boli pre nich prínosné recenzie projektu, ktoré dostali od ich kolegov*. Na výber mali šesť odpovedí (vrátane možnosti *iné*), pričom mohli označiť aj viac ako jednu odpoveď (Obr. 2). Jednoznačne najviac študentov (96,3%) si vybralo možnosť „pomohlo mi to opraviť nedostatky, ktoré som si nevšimol pri submitovaní“. Približne tretina

respondentov uviedla, že im to pomohlo pri písaní ďalších recenzií a štvrtina z nich zároveň uviedla, že im to pomohlo lepšie pochopiť zadanie projektu a/alebo opraviť nedostatky, ktoré nevedeli vyriešiť pred odoslaním projektu.



**Obrázok 2. Prínos udelených recenzií projektu: A – lepšie pochopenie zadania, B – oprava nepovšimnutých nedostatkov, C – oprava nedostatkov, ktoré nevedel vyriešiť, D – písanie lepších recenzií, E – nič, F – iné**

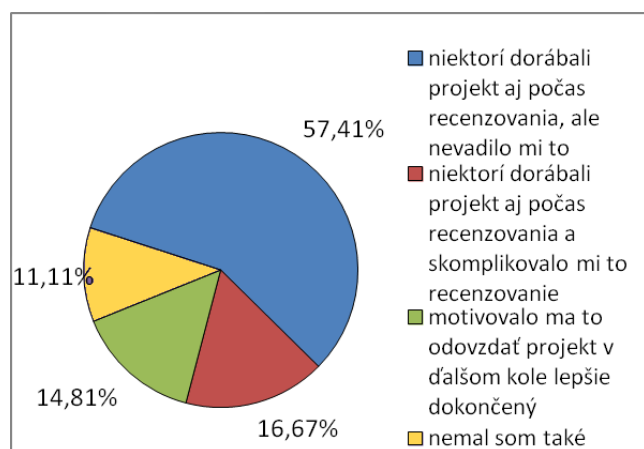
Pri odovzdávaní projektov na recenzovanie sa často stávalo, že študenti odovzdali nedokončený projekt. To mohlo výrazne skomplikovať (prípadne zjednodušiť) prácu iných študentov pri recenzovaní takého projektu. Preto sme sa študentov v záverečnom dotazníku pýtali, či sa s nedokončenými projektmi stretli a ak áno, aká bola ich skúsenosť s recenzovaním takýchto projektov. Takmer 80% študentov sa vyjadrilo, že väčšina projektov, ktoré hodnotili, boli dostatočne dokončené, no pomerne veľa študentov (14,81%) tvrdilo, že väčšina projektov, ktoré recenzovali, bola nedokončená (Obr. 3).



**Obrázok 3. Dokončenosť recenzovaných projektov**

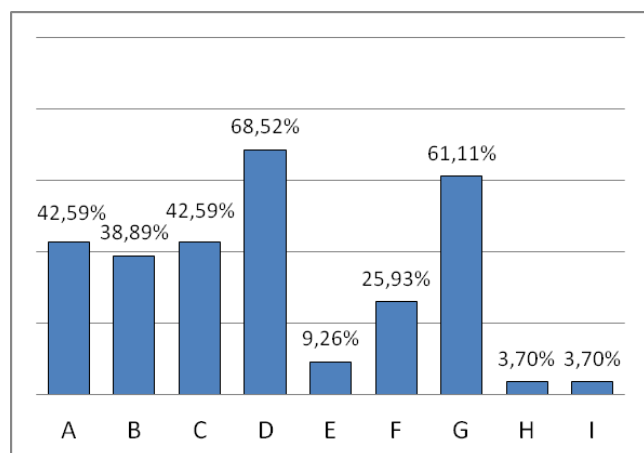
Skúsenosti s recenzovaním projektov, ktoré neboli dostatočne dokončené, sa týkala otázka so štyrmi možnosťami odpovede, z ktorých si študenti vyberali jednu (Obr. 4). Najviac z nich (57,41%) vyhlásilo, že skutočnosť, že niektorí študenti dorábali projekt ešte aj počas recenzovania, im nevadilo. Šestina opýtaných uviedla, že im takéto neskoré dorábanie projektu skomplikovalo recenzovanie. Takmer 15% študentov zároveň uviedlo, že ich takéto skúsenosti motivovali odovzdať svoj projekt v ďalšej fáze lepšie dokončený.





**Obrázok 4. Skúsenosť s recenzovaním nedokončených projektov**

Posledná z otázok dotazníka patriaca do tejto sekcie sa venovala benefítom, ktoré študenti získali z recenzovania projektov svojich kolegov. Vzhľadom k tomu, že za samotné recenzovanie boli študenti odmeňovaní bodmi, očakávali sme najčastejší výber tejto možnosti. Avšak pozitívne bolo zistenie, že najviac študentov (68,52%) vybralo aj možnosť „pri prezeraní projektov kolegov som si uvedomil nedostatky v mojom projekte“ (Obr. 5). Body „navyše“ za recenzovanie ocenilo 61,11% študentov. Študenti taktiež ocenili, že získali skúsenosti s testovaním webových projektov (42,59%), lepšie si uvedomili, aké rôzne chyby sa môžu vyskytnúť na webových stránkach (42,59%) a zistili, ako zadanie projektu pochopili ich kolegovia (38,89%).



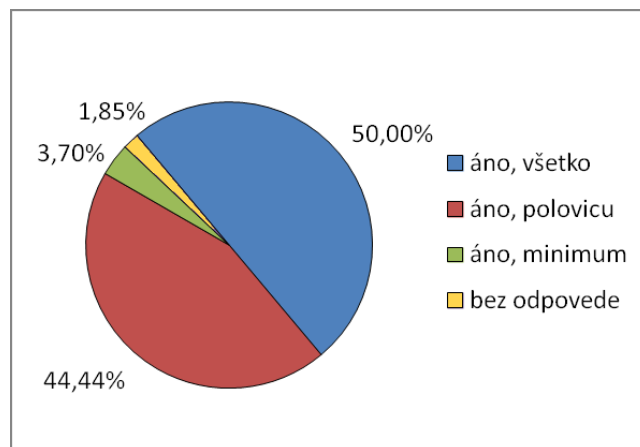
**Obrázok 5. Prínos z recenzovania iných projektov: A – skúsenosti s testovaním, B – ako zadanie chápu kolegovia, C – uvedomenie si rôznych chýb na webových stránkach, D – uvedomenie si nedostatkov vo vlastnom projekte, E – tréning vyjadrovacích schopností, F – tréning konštruktívnej kritiky, G – body navyše, H – nedalo mi to nič, I – iné**

### 3.2 Nová organizácia projektu

Ako sme uviedli v druhej kapitole, odovzdávanie projektov v danej fáze malo tri rôzne termíny. Zaujímalo nás teda názor študentov na tento neštandardný postup pri odovzdávaní projektu, preto sme do dotazníka zaradili i ďalšie tri otázky.

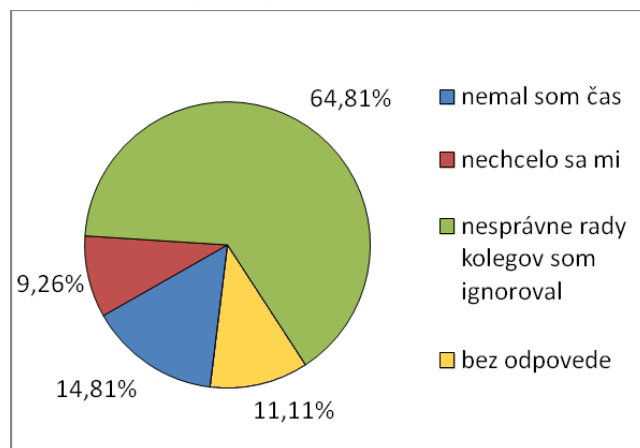
V prvej z nich sme sa študentov pýtali, či využili možnosť opraviť si projekt na základe získaných recenzií ešte pred jeho finálnym

odovzdaním. Polovica z nich sa vyjadrila, že zapracovali všetky pripomienky, ktoré od kolegov dostali (Obr. 6). O čosi menej študentov (44,44%) tvrdilo, že zapracovali približne polovicu pripomienok. Možnosť „napriek kritike som projekty neopravil“ si nevybral ani jeden zo študentov.



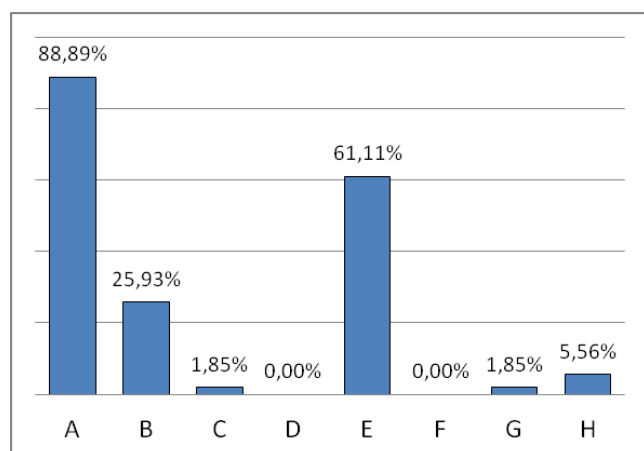
**Obrázok 6. Využitie možnosti na opravu projektu na základe recenzií kolegov**

To, že študenti nezapracovali všetko, čo im ich kolegovia vytkli, nemusí znamenať, že ich rady odignorovali. Mohlo sa stať, že rady kolegov neboli správne a keby boli do projektu zapracované, ten by bol učiteľom horšie vyhodnotený. Preto sme sa v ďalšej otázke snažili nájsť hlavný dôvod, prečo študenti na spätnú väzbu kolegov nereagovali. Väčšina z nich (64,81%) odpovedala výberom možnosti „pochopil som, že niektoré rady od spolužiakov môžu byť aj nesprávne, a tak som ich ignoroval“ (Obr. 7). Našli sa však aj študenti, ktorí si vybrali možnosť „nemal som čas“ (14,81%), či „nechcelo sa mi“ (9,26%).



**Obrázok 7. Dôvody, pre ktoré študent časť spätnej väzby svojich kolegov nezapracoval do projektu**

V otázke týkajúcej sa benefítov z možnosti opraviť si projekty pred finálnym odovzdaním, študenti neprekvapili a jednoznačne najviac z nich vybralo možnosť „mal som možnosť získať viac bodov za projekty“ (88,89%). Zároveň mnohí z nich (61,11%) videli prínos i v čase navyše, ktorý na tvorbu projektu získali. Z ostatných poradili ako odstrániť chyby, ktoré som sám odstrániť nevedel, ktorú označila štvrtina respondentov (Obr. 8).



**Obrázok 8. Prínosy z možnosti opraviť si projekty pred finálnym odovzdaním: A – možnosť získať viac bodov za projekt, B – rady kolegov, ako odstrániť chyby v projekte, C – pomohlo to, ale boli aj nesprávne rady kolegov, D – väčšina rád bola zlá, E – čas navyše, F – žiadny prínos, G – takéto spôsoby odovzdávania projektov nie je fér, H – iné**

#### 4. ZÁVER

Úlohou moderného vzdelávania nie je iba edukovať v istej odbornej oblasti, ale rozvíjať i iné schopnosti a zručnosti potrebné pre život a prácu v 21. storočí. K týmto patrí aj schopnosť kriticky uvažovať nielen o svojej práci, ale i o práci svojich kolegov, a tiež schopnosť túto kritiku formulovať a tým prispieť k ich formatívnemu hodnoteniu. To sa dá viacerými spôsobmi. V našej praxi sme sa rozhodli pre používanie tzv. peer review, teda vzájomného hodnotenia sa študentov, v tomto prípade na kurze Webdizajnu v odbore Aplikovaná informatika. V úvodných rokoch bolo peer review nasadené na aktivity súvisiace s publikovaním blogových článkov. V minulom roku sme recenzovanie presunuli na projekt rozdelený do troch fáz, v rámci ktorého si študenti vytvárajú svoj vlastný blog. Navyše sme pre každú fázu navrhli tri rôzne termíny odovzdania: prvý termín na odovzdanie aktuálnej fázy projektu na recenzovanie, druhý termín na odovzdanie recenzií troch pridelených projektov kolegov a tretí termín na odovzdanie aktuálnej fázy projektu na hodnotenie učiteľovi. V čase medzi odovzdaním recenzií a odovzdaním projektu na hodnotenie mohli študenti zapracovať pripomienky recenzentov do svojho projektu.

V závere semestra sme sa študentov v dotazníku pýtali, aký prínos pre nich mala táto aktivita. Mnohí (96,30%) uviedli, že recenzie kolegov im pomohli opraviť si nedostatky, ktoré si nevšimli pri prvom odovzdávaní, či to, že vďaka recenzovaniu iných projektov si uvedomili chyby v tom svojom (68,52%). Navyše takmer 95% študentov podľa výsledkov dotazníka zapracovalo aspoň polovicu z toho, čo bolo uvedené v recenziách. Negatívne odpovede na všetky uvedené otázky sa buď nevyskytovali, alebo sa takto vyjadril jeden – dvaja respondenti.

Na základe uvedeného môžeme takéto rozvrhnutie aktivít na kurze z pohľadu spokojnosti študentov vyhodnotiť ako vhodné, podporujúce ich štúdium a rozvíjajúce ich schopnosť uvažovať o práci iných.

#### 5. POĎAKOVANIE

Príspevok je súčasťou výskumu v rámci projektu VEGA 1/0948/13.

#### 6. BIBLIOGRAFICKÉ ODKAZY

- [1] LIU, E.Z.F., LIN, S.S.J., CHIU, C.H., YUAN, S.M.: *Web-based peer review: the learner as both adapter and reviewer*. IEEE Transactions on Education 44(3) (2001) 246–251
- [2] POPESCU, E., MANAFU, L.: *Repurposing a wiki for collaborative learning – pedagogical and technical view*. In: ISTCC. (2011)
- [3] STERBINI, A., TEMPERINI, M.: *Dealing with open-answer questions in a peer-assessment environment*. In: ICWL. (2012)
- [4] KEN, P.M., PARK, K.H., DOMAZLICKY, B.R.: *Peer grading of essays in a principles of microeconomics course*. Journal of Education for Business 70(6) (1995)
- [5] LADYSHEWSKY, R.K., GARDNER, P.: *Peer assisted learning and blogging: A strategy to promote reflective practice during clinical fieldwork*. Australasian Journal of Educational Technology 24(3) (2008)
- [6] WU, W.S.: *The effect of blog peer review and teacher feedback on the revisions of EFL writers*. Journal of Education and Foreign Languages and Literature 3(2) (2006)
- [7] MACALPINE, J.M.K.: *Improving and encouraging peer assessment of student presentations*. Assess Eval High Educ 24(1) (1999)
- [8] GEHRINGER, E.F.: *Strategies and mechanisms for electronic peer review*. In: Proceedings Frontiers in Education (2000)
- [9] GEHRINGER, E.F., Ehresman, L.M., Skrien, D.J.: *Expertiza: Students helping to write an OOD text*. In: OOPSLA (2006)
- [10] GEHRINGER, E.F.: *Building resources for teaching computer architecture through electronic peer review*. In: WCAE (2003)
- [11] BEJDOVÁ, V., HOMOLA, M., KUBINCOVÁ, Z.: *Blogging in Obligatory Course: A Bitter Victory*. In: Advances in Web-Based Learning–ICWL 2014, Springer (LNCS), 2014, Vol. 8613, p. 1–10, ISBN 978-3-319-09635-3
- [12] BEJDOVÁ, V., KUBINCOVÁ, Z., HOMOLA, M.: *Blogging activities with peer-assessment in university courses*. In: Information and Communication Technology in Education 2013. Ostrava: University of Ostrava, 2013.
- [13] BEJDOVÁ, V., KUBINCOVÁ, Z., HOMOLA, M.: *Are Students Reliable Peer-Reviewers?* In: Proceedings of the 14th International Conference on Advanced Learning Technologies (ICALT), 2014 IEEE, Athens, ISBN 978-1-4799-4038-7
- [14] DROPČOVÁ, V., KUBINCOVÁ, Z.: *Evaluation of peer-review quality in comparison to teachers' grading*. Information and Communication Technology in Education 2014, Ostrava: University of Ostrava, 2014.
- [15] KUBINCOVÁ, Z., HOMOLA, M., BEJDOVÁ, V.: *Motivational effect of peer review in blog-based activities*. In: ICWL 2013. Kenting, Taiwan. Springer, LNCS, vol. 8167 (2013)

# IT FACTOR

Tomáš Dudík  
Gymnázium Viliama Paulinyho-Tótha v Martine  
Malá Hora 3  
036 01 Martin  
Slovensko  
tduik@gmail.com

Erika Lettrichová  
Gymnázium Viliama Paulinyho-Tótha v Martine  
Malá Hora 3  
036 01 Martin  
Slovensko  
ericka.lett@gmail.com

## ABSTRAKT

IT FACTOR je tímovo-informaticko-experimentálna súťaž určená pre základné školy. Je to jedinečná súťaž. Má jednoduché pravidlá. Tímová preto, lebo je určená pre 4-členné tímy zo základných škôl. Informatická preto, lebo sme chceli u žiakov overiť ich znalosti z informatiky a zručnosť vo vyhľadávaní informácií na internete. Taktiež sme využili princípy zážitkovej pedagogiky na šiestich stanoviskách. Hlavným zadáním súťaže bolo vyriešiť 55 úloh s informatickým a logickým zadáním v časovom limite 90 minút. Za každú správne vyriešenú úlohu súťažiaci dostali zlatú fazuľku a kód. Zlatá fazuľka slúžila ako vstupenka na bonusové stanoviská, ktoré boli inšpirované známymi počítačovými hrami. Stanoviská boli zamerané aj na fyzickú zdatnosť súťažiacich.

## ABSTRACT

IT FACTOR is great team-informatics-experimental competition for elementary schools. It's type of competition which hasn't been here before. It has very simple rules. Team, because there are 4-member teams from elementary schools. Informatics, because we want to check your knowledge and internet skills. We apply principles of experimental education at 6 check points. Your main task as a competitor is to solve 55 information and logic exercises within the time limit 90 minutes. For every correct answer, you receive a golden bean and a code. Golden beans enable you to go bonus check points which are inspired by famous PC games. These checkpoints also focus on physical ability

## Kľúčové slová

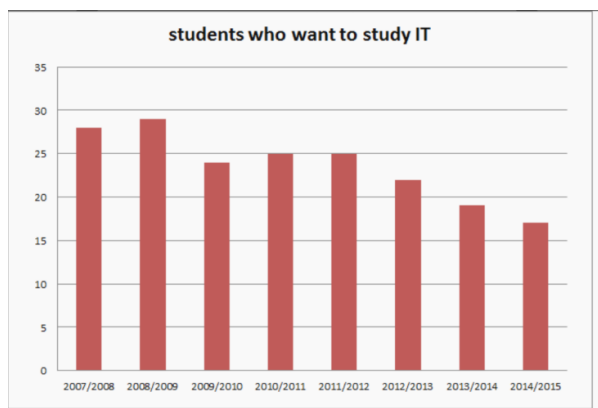
IT FACTOR, súťaž, informatika, škola, žiaci

## Keywords

IT FACTOR, competition, informatics, school, students

## 1. ÚVOD

V posledných rokoch sa našej škole stretávame s klesajúcim záujmom o štúdium informatiky (viď. graf). Na základe týchto údajov sme hľadali možnosti ako tento negatívny trend spomaliť, alebo ho úplne potlačiť.



Obrázok 1. Graf

## 2. ANALYZOVANIE PROBLÉMU

Would you like to continue in studying computer Science?

☐ Yes

☐ No

If no, why not?

☐ I am not interested in computer science

☐ I am not good at it, I cannot solve the task

☐ I don't have enough good knowledge

Other: \_\_\_\_\_

What do you think about the lessons of the computer science?

☐ The lessons are interesting

☐ Not enough lessons

☐ Learning only more types of software

☐ Not many theory, little practice

Other: \_\_\_\_\_

What would you change on the lessons of computer science?

☐ I would add more lessons

☐ I would have more on learning to work in teams

☐ I would try to make the lessons more interesting

☐ I would more participate in competitions

Other: \_\_\_\_\_

Submit

Powered by Formulizr - Google docs online questionnaire tool

Obrázok 2. Dotazník

Pomocou vstupného dotazníka, ktorý sme dali vyplniť žiakom prvého ročníka (štatistická vzorka približne 100 žiakov) sme ako hlavné príčiny analyzovali tieto:

1. Nedostatočná hodinová dotácia predmetov, kde sa venujú informatike.
2. Informatika v posledných ročníkoch nebola v učebnom pláne zastúpená vôbec, alebo len minimálne (1 hodina do dvoch týždňov).
3. Obsah hodín je zameraný na konkrétny softvér (najčastejšie prezentačný).

Na základe týchto poznatkov sme sa rozhodli pre vytvorenie súťaže IT FACTOR. Súťaž sme zorganizovali 5. novembra 2014 v priestoroch našej školy - Gymnázium Viliama Paulinyho-Tótha v Martine. Dovolíme si tvrdiť, že táto súťaž je jediná svojho druhu v Žilinskom kraji.

Radi by sme vám priblížili jej princíp a herný systém, aby ste v prípade záujmu mohli túto súťaž zorganizovať aj na vašej škole.

## 3. CHARAKTERISTIKA SÚŤAŽE

Súťaž IT FACTOR môžeme charakterizovať ako tímovo-informaticko-zážitkovú súťaž určenú pre druhý stupeň základných škôl.

- **Tímová**, pretože sa jej zúčastnilo deväť štvorčlenných tímov zo základných škôl.
- **Informatická** preto, lebo sme chceli preveriť znalosti z informatiky a prácu s internetom a ukázať im, že informatika nie je len o sociálnych sieťach.
- **Zážitková** – využili sme princípy zážitkovej pedagogiky, tak aby jej priebeh bol pre súťažiacich čo najviac zaujímavý.

### 3.1 Priebeh súťaže

Hlavnou úlohou súťažiakov bolo v časovom limite 90 minút vyriešiť 55 úloh s informatickým a logickým zameraním. Počas riešenia týchto úloh mali k dispozícii notebooky a internet.

### 3.2 Úlohy

16	Na internete zistite 6 spoločných zastávok pre pre linku č.1 a linku č.3 mestskej hromadnej dopravy v Žiari nad Hronom.	5b	D	2YDo16ZB
----	---	----	---	----------

#### Obrázok 3. Príklad súťažnej úlohy

Jednotlivé úlohy sme koncipovali na základe školského vzdelávacieho programu pre základné školy. Na začiatku súťaže dostal každý tím prvých päť úloh. Následne ak tím vedel správnu odpoveď na otázku išiel ju oznámiť rozhodcovi (každý príklad obsahoval písmenko, ktoré reprezentovalo príslušného rozhodcu, ktorý mal úlohu na starosti). V prípade, že odpoveď bola správna, rozhodca odovzdal súťažiakemu tímu ďalšiu otázku a zlatú fazuľku. Zadanie úlohy obsahovalo aj jednoznačný kód.

Žiak za každý vyriešený príklad dostal:

- Novú úlohu
- Jednoznačný kód
- Zlatú fazuľku

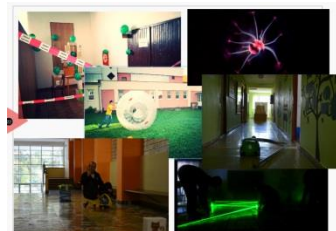
3.2.1 *Jednoznačný kód* – slúžil na online hodnotenie výsledkov daného tímu. Žiaci mali k dispozícii rozhranie, do ktorého mohli zadávať svoje kódy. Za takto zadaný kód sa im pripočítali v systéme body. Tento proces bol pre jednotlivé tímy vizualizovaný pomocou virtuálneho zvieratka „Tamaguchi“. Tamaguchimu za každý zadaný kód pribudla nejaká „fičúrka“ ako napríklad jedlo, voda, oblečenie. Jednotlivé tímy mohli tak svoje získané body vidieť aj na pribúdajúcich „fičúrkach“.

3.2.2 *Zlaté fazuľky* – slúžili ako vstupenka na špeciálne stanoviská (celkový zoznam stanovísk je popísaný nižšie). Žiaci mali k dispozícii mapku, kde bola zakreslená poloha špeciálnych stanovísk a rovnako bol zobrazený na mapke aj počet fazuľiek, ktoré potrebovali pre vstup na dané stanovisko. Teoretickým podkladom pre implementáciu týchto stanovísk do priebehu súťaže bola zážitková pedagogika. Využili sme jej základný princíp, ktorý hovorí, že pri aktivitách a situáciách, ktoré sú sprevádzané intenzívnou **emóciou** žiaci dokážu lepšie prijímať a uchovávať informácie. Aktivity na týchto stanoviskách sme sa ďalej snažili koncipovať tak, aby boli zamerané na tímovú spoluprácu. Na pozadí každého stanoviska bola počítačová hra a to preto, aby princíp stanoviska bol súťažiacim čo najbližší. Je zrejme, že tím, ktorý v danom časovom limite, získal, čo najväčší počet bodov vyhral.

### 3.3 Zážitkové stanoviská

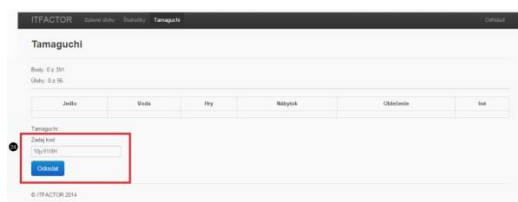
1. Beňuškov vysávač - úlohou žiakov zostreliť plechovky detskými hopsalkami, ktoré slúžili ako náboje do „guľometu“ vyrobeného z odpadovej rúry a vysávača.
2. Angry birds – v tejto bonusovej úlohe si žiaci mohli vyskúšať strelbu z veľkého praku a popritom zničiť pevnosť priasotok v podobe balónikov a krabíc.
3. Plazmacar - Súťažiaci mali za úlohu prejsť prekážkovú dráhu na zradnom autíčku plazmacar.
4. Laser - každý člen tímu dostal malé zrkadielko a pomocou tímovej spolupráce museli doviest paprsok lasera až do cieľa. Na zvýraznenie lasera sme použili glicerín.

5. Slenderman – úlohou súťažiakov bolo v zatemnenej miestnosti nájsť konkrétnu informáciu. Informácie na papierikoch si mohli prečítať len s pomocou trubice z neónovej žiarivky, ktorú mohli priložiť ku plazma guľi
6. Poslednou bonusovou úlohou bol valec. Úlohou súťažiakov bolo počas kotúľania sa v nafukovacom valci vypočítať čo najviac jednoduchých matematických príkladov.



Obrázok 4. Fotografie z bonusových úloh

## 4. HODNOTIACE ROZHRAVIE SÚŤAŽE



Obrázok 5. Hodnotiace rozhranie

Hodnotiace rozhranie slúžilo na zadávanie kódov, ktoré dostali súťažiaci za každý správne vyriešený príklad. Po zadaní kódu tímu pribudla nejaká „fičúrka“ ako napríklad jedlo, oblečenie, voda, hry a iné. Taktiež po zadaní tohto kódu sa tímu prirátali body.

Hodnotiace rozhranie bolo vytvorené študentom na našej škole a je možné si ho stiahnuť na adrese:

<https://github.com/Kak3n/itfactor>

## 5. FINANCOVANIE SÚŤAŽE

Financovanie súťaže sme zabezpečili pomocou projektu Komprax. (<https://www.iuventa.sk/sk/KomPrax/Home.alej>)

## 6. ZÁVER

Súťaž IT FACTOR mala u žiakov úspech a preto ju plánujeme zorganizovať aj budúci rok.

V prípade akýchkoľvek realizačných otázok je možné obrátiť sa na autora článku.

## 7. BIBLIOGRAFICKÉ ODKAZY

[1] PELÁNEK, Radek. Příručka instruktora zážitkových akcí, vyd: Portál, 2008, str. 40 -123, ISBN: 978-80-7367-353-6

# Vzdělávání budoucích učitelů informatiky

Rostislav Fojtík  
Ostravská univerzita v Ostravě  
30. dubna 22  
Ostrava  
Česká republika  
rostislav.fojtik@osu.cz

## ABSTRAKT

Výuka informatiky a výpočetní techniky je nucena navazovat na velmi dynamický rozvoj informačních a komunikačních technologií (ICT). Je proto potřeba se velmi důkladně věnovat změnám, které se v tomto oboru neustále objevují a zavádět je do vyučovacího procesu. Velkou nevýhodou výuky informatiky je, že se jedná o relativně velmi mladý a rychle se rozvíjející obor, ve kterém se nedá tolik stavět na předcházejících zkušenostech. České školství se neustále potýká s nedostatkem kvalifikovaných vyučujících informatiky. Mnozí z vyučujících stále tápou a hledají vhodnou náplň předmětu a adekvátní metody. Vzhledem k často obecným formulacím obsaženým v Rámcovém vzdělávacím programu se tematické plány jednotlivých škol znatelně liší. Analýzou vybraných tematických plánů středních škol bylo zjištěno často nevhodné uspořádání učiva. Často se pak výuka informatiky zaměřuje za výuku konkrétních softwarových aplikací. Článek rovněž přináší výsledky průzkumu, který se zabývá srozumitelností vybraných programovacích jazyků.

## ABSTRACT

Teaching computer science is forced to follow on from a very dynamic development of ICT. It is therefore necessary to very thoroughly discuss the changes that are in this field continuously appear and introduce them into the teaching process. The big drawback for computer science education, it is a relatively young and rapidly developing field, where it is not so much to build on previous experience. The Czech education system is constantly facing a shortage of qualified teachers of informatics. Many of the teachers are still groping to find a suitable content of the subject and adequate methods. Thematic programs of schools are often very different. This is due to the fact that the content of FEP is written very unspecific. Analysis of selected thematic programs of secondary schools were found to often inappropriate arrangement of the curriculum. Often it is confusing for teaching science teaching specific software applications. The paper also presents the results of the survey, which deals with the clarity of selected programming languages.

## Klíčová slova

Didaktika informatiky, informatika, programování, tematický plán.

## Keywords

Didactics of informatics, computer science, programming, thematic program.

## 1. ÚVOD

Výuka informatiky je v trochu jiné situaci než většina ostatních předmětů na základní či střední škole. Zatímco tradiční předměty jako například matematika, fyzika, dějepis, biologie a další mohou navazovat na mnohaleté zkušenosti generací učitelů, pak informatika je relativně velmi mladý obor, který se navíc velmi

dynamicky vyvíjí. Výuka informatiky nemá možnost navazovat na zkušenosti výuky z minulých let. Nejen, že se mění velmi často zásadně obsah učiva, ale také chybí důkladněji zpracované metodické návody a postupy. Dalším problémem je neustálý deficit aprobevaných učitelů informatiky.

## 2. ROLE DIDAKTIKY INFORMATIKY

### 2.1 Odlišnosti

Didaktika informatiky se do jisté míry liší od oborových didaktik ostatních předmětů. Zatímco oborové didaktiky „klasických“ předmětů se hlavně zabývají tím, jak učit jednotlivá témata, jaké použít metody a formy a jak tyto informace vhodně předat studentům učitelství, tak u didaktiky informatiky velmi důležitou roli hraje rovněž výběr témat.

Nedostatek zkušeností, nemožnost navazovat na metodické postupy a zkušenosti z minulých let, malý počet plně aprobevaných vyučujících, nepochopení hlavních cílů výuky informatiky a nejasnost výběru témat vede k snižování kvality výuky informatiky. Problémem výuky informatiky je například značně rozdílný přístup k časové dotaci u jednotlivých škol. Například u šesti zkoumaných základních škol byla velmi znatelná diference. Zatímco dvě ze škol vyučují informatiku jen hodinu týdně v pátém a šestém ročníku, tři školy vyučují informatiku jednu hodinu týdně ve všech ročnících druhého stupně. Poslední ze zkoumaných škol věnovala výuce informatiky hodinu týdně v pátém až sedmém ročníku. Polovina ze zkoumaných škol uvedla, že nemá kvalifikované učitele. Absence kvalifikovaného učitele neměla vztah k počtu hodin, které škola výuce informatiky věnovala. [1]

Podobný průzkum byl proveden na deseti vybraných čtyřletých gymnáziích z celé České republiky. Z analýzy Školních vzdělávacích programů (ŠVP) vyplývá, že 70 % zkoumaných gymnázií využívá časovou dotaci celkově 4 hodiny týdně od prvního do čtvrtého ročníku. Nejvíce se výuce informatiky školy věnují v prvním a druhém ročníku. [2]

### 2.2 Výběr témat

Výběr témat, které by měly být vyučovány, je zásadním problémem mnoha vyučujících informatiky. Rámcový vzdělávací program (RVP) sice určuje rámec výuky, ale je obvykle popsán velmi obecně a rozdílnost mezi školami je pak značná. Proto by měli budoucí učitelé informatiky věnovat při svém studiu této problematice dostatečnou pozornost.

Například studenti učitelství informatiky na Ostravské univerzitě v Ostravě mají v rámci předmětu Didaktika informatiky pro základní a střední školy zpracovat tematický plán pro první ročník gymnázia. Jejich prvním úkolem je však nejprve přinést na výuku tematický plán libovolného gymnázia. V rámci vyučovací hodiny je tento tematický plán podroben analýze. Studenti jsou nabádáni k tomu, aby hodnotili výběr témat, jejich návaznost a časovou



dotaci. Každoročně zjišťujeme, že minimálně polovina těchto tematických plánů nemá potřebnou kvalitu. U některých je na první pohled patrné, že jsou vypracovány s největší pravděpodobností pouze proto, že je to vedením školy požadováno, a učitel s plánem během roku zřejmě nepracuje. Mezi typické nedostatky patří například názvy témat. Například téma MS Word, MS Excel atd. Tvůrci těchto materiálů si zřejmě neuvědomují, že cílem obvykle není naučit konkrétní software, ale pracovat s informacemi, dokumenty a schopnost správně využívat informační a komunikační technologie v praxi. Mezi další nedostatky patří například výběr témat a jejich časová dotace. Není neobvyklé, aby se v tematických plánech pro gymnázium objevilo téma MS Word s časovou dotací 40 hodin. Dílčí témata jsou pak výčtem funkcionalit a nabídek textového editoru. Žáci se při takové výuce seznámí s téměř vším, co textový editor umí, ale není zaručeno, že budou schopni správně zpracovat textový dokument a že o všech naučených možnostech editoru budou vědět k čemu a jak je v praxi využít. Velká časová dotace pro aplikace MS Office vede k absenci dalších důležitých témat, jako jsou například zpracování informací, multimédia, základy programování a podobně.

## 2.3 Výuka programování

Typickým příkladem problematického tématu je výuka programování. Ta obvykle není na základních ani středních školách dostatečně zastoupena. Z výzkumu provedeného mezi 63 středními školami, 71 základními školami a 15 víceletými gymnázii v Moravskoslezském kraji vyplývá, že programování je mnohde sice zařazováno do tematických plánů, ovšem koncepce výuky není příliš vhodná. [3][4] První samostatně zkoumanou skupinou bylo 63 středních škol v Moravskoslezském kraji, z nichž 84% uvedlo, že se nějakým způsobem věnuje výuce programování. Nejčastěji využívaným jazykem byl Pascal. Druhou zkoumanou skupinou bylo 71 základních škol a 15 víceletých gymnázií (nižší stupeň). Z tohoto počtu jen 32 % respondentů zařadilo do svých tematických plánů v rámci předmětu výpočetní technika nebo v zájmovém kroužku výuku programování. 22 % respondentů vyučovalo programování v dřívějších dobách a zbylých 46 % programování nevyučovalo ani nevyučuje. Jedním z důvodů malého zastoupení výuky programování je i nadále nedostatečná aprobace vyučujících. Pouze 15 % respondentů uvedlo, že má informatiku jako svou původní aprobaci. 30 % respondentů absolvovalo informatiku v rámci některé z forem rozšiřujícího studia a ostatní učí informatiku neaprobovaně.

Další průzkum se zaměřil na důkladnější rozbor školních vzdělávacích programů, učebních plánů a tematických plánů vybraných 14 gymnázií a 13 středních průmyslových nebo odborných škol. Co se týká výuky programování a základů algoritmizace, tak na 17 ze sledovaných škol měli alespoň nějakou část výuky věnovanou programování a algoritmizaci. Některé průmyslové a odborné školy se věnují výuce konkrétního programovacího jazyka, který je nutný například pro programování obráběcích strojů nebo mikročipů v elektronických zařízeních. U některých středních škol je výuka zaměřena pouze na základy algoritmizace. Učební plány studijních oborů zaměřených na informatiku obsahují předměty programování, které se obvykle vyučují 2 až 3 roky. Podobně i část gymnázií má ve svých učebních plánech předměty, ve kterých je nezanedbatelná část hodin věnována algoritmizaci a programování. Na gymnáziích se programování obvykle vyučuje ve volitelných seminářích. V některých tematických plánech bylo nalezeno nevhodné strukturování témat a jejich

časová dotace. Jako nesprávný příklad může sloužit jedno gymnázium, které tématu programování v jazyku Pascal věnuje pouhých 19 hodin. Taková časová dotace je naprosto nedostačující. Studenti v tak krátkém čase v žádném případě nemohou zvládnout ani základy programovacího jazyka a výuka programování se pro ně stává nepochopitelnou. Naproti tomu jsou schopni učitele věnovat tematickému celku MS Word 40 vyučovacích hodin.

Každoročně jsou studenti učitelství informatiky Ostravské univerzity dotazováni, zda a případně v jakém rozsahu by se měli žáci základních a středních škol učit algoritmizaci a programování. Obvykle je naprostá většina dotázaných studentů přesvědčena, že výuka algoritmizace a programování nepatří na základní školu. Část z nich připouští, že tato výuka může být zařazena například na gymnázium, ale až ve vyšších ročnících a jen ve výběrovém předmětu. Jako neobvyklejší důvody proti výuce programování studenti uvádí:

- Výuka programování je příliš složitá pro žáky základních i většiny středních škol,
- programovací jazyky jsou nesrozumitelné a žáci by je nezvládli,
- vývojové nástroje jsou komplikované a nepřehledné,
- výuka programování je časově náročná a nezůstal by čas na důležitější témata,
- studenti mají negativní zkušenost s výukou programování na střední nebo základní škole,
- nepotřebujeme tolik programátorů v praxi.

Studenti často přiznávají, že měli obtíže při absolvování předmětů zaměřených na programování, a proto by se raději ve své budoucí pedagogické praxi výuce programování vyhnuli.

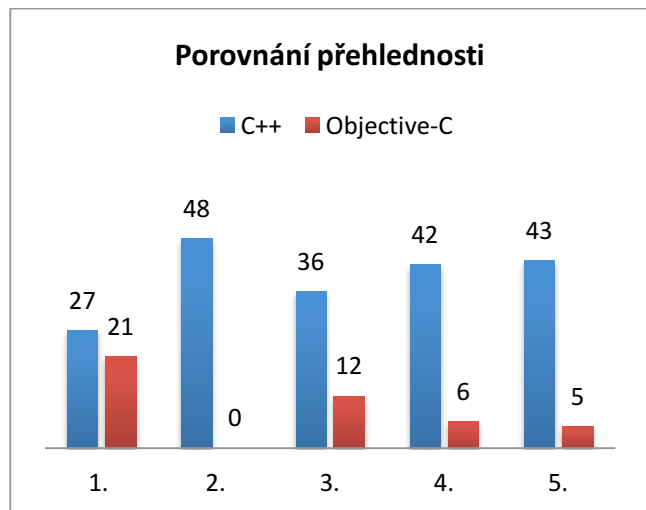
Studentům je potřeba jasně vysvětlit rozdíly mezi výukou programování na úrovni profesionálního programátora či výukou konkrétního programovacího jazyka a výukou algoritmizace a základů programování. Teprve, když pochopí, že cílem výuky na základní a obvykle i na většině středních škol není zvládnout s žáky některý z profesionálně užívaných programovacích jazyků, pochopí, že výuka algoritmizace a základů programování je důležitá. Negativní postoj k výuce programování na střední či základní škole souvisí se špatnými zkušenostmi, které studenti během studia na základní a střední škole měli. Teprve, když se během didaktiky informatiky důkladněji seznámí s nástroji jako Karel, Scratch, Logo či možnostmi programovat jednoduché roboty, zjišťují, že takto koncipovaná výuka algoritmizace a základů programování dává smysl i na základní a střední škole. A získané kompetence nejsou důležité jen pro budoucí profesionální programátory, ale všechny žáky.

Další důležitou roli didaktiky informatiky je studentům objasnit, že učitele informatiky by neměli za každou cenu učit tomu, čemu se sami učili ve škole. Informatika se velmi rychle rozvíjí a je potřeba tyto změny reflektovat ve výuce. Klasickým příkladem je neustálé využívání jazyka Pascal ve výuce. Velká část vyučujících trvá na tom, že se jedná o nejvhodnější programovací jazyk [4]. Neuvědomují si, že tomu tak v současné době již není. Chceme-li naučit pouze algoritmizaci, pak k tomu nepotřebujeme programovací jazyk, který má složitou strukturu, obsáhle syntaktická pravidla pro psaní kódů, komplikovanou soustavu datových typů, ukazatele a další prvky, které žáky jen odvádí od základních problémů. Mnohem výhodnější je používat mikrosvětý typu Karel, Scratch nebo Logo, případně programovací roboty. Uvedené nástroje umožní žákům mnohem lépe pochopit problematiku tvorby algoritmů a základních programů, než



programovací jazyk a učiteli mnohdy používané abstraktní matematické úlohy, které mají žáci naprogramovat.

Pokud je však cílem výuky skutečně naučit žáky programování, pak je vhodné vybrat programovací jazyk, který lépe odpovídá současným standardům a aktuálně používaným programovacím paradigmatům. Ukazuje se, že je vhodnější používat programovací jazyky, jejichž syntaktická pravidla jsou obecně používaná a rozšířená. Například mezi 48 studenty prvního ročníku oboru Aplikovaná informatiky byl proveden průzkum srozumitelnosti kódů programovacích jazyků C++ a Objective-C (obrázek 1). Zatímco jazyk C++ používá podobná syntaktická pravidla jako současně rozšířené jazyky Java, C, C#, JavaScript, PHP a další, jazyk Objective-C vychází z koncepce méně obvyklého jazyka SmallTalk. Studentům bylo představeno pět ukázek různých kódů v obou uvedených programovacích jazycích. Mezi příklady byly jednoduché výstupy na obrazovku i složitější práce suvolňováním paměti při rušení objektů. Respondenti měli popsat, co uvedený kód provádí a který z uvedených jazyků jim připadá srozumitelnější a logičtější. U některých příkladů až 100 % respondentů uvádělo, že kód v jazyce C++ jim připadá srozumitelnější a přehlednější. Respondenti u jazyka C++ mnohem lépe interpretovali činnost kódu. [5] Z výsledků lze odvodit, že pro základní výuku je vhodnější využít programovací jazyk, který svou strukturou a pravidly zápisu je studentům známější a srozumitelnější. Pro úvodní kurzy výuky programování není vhodné využívat málo rozšířené a netradiční programovací jazyky, protože žáci pak musí věnovat mnohem více energie zvládnutí samotného jazyka a nesoustředí se tolik na zvládnutí základních principů algoritmizace a programování.



Obrázek 1. Porovnání přehlednosti programových kódů

### 3. ZÍSKÁVÁNÍ APROBOVANÝCH UČITELŮ

Jak již bylo dříve uvedeno, je velkým problémem nedostatečný počet aprobovaných učitelů informatiky. Jednou z důležitých rolí didaktiky informatiky je ukázat studentům, že učení může být zajímavé. Pořád se mezi studenty učitelství informatiky objevuje nemalá část, která na začátku kurzu deklaruje, že učit nechce a raději dává přednost práce v počítačových firmách.

Jednou z možností, jak zvýšit počet aprobovaných učitelů je rozšiřující studium informatiky. Katedra informatiky a počítačů Ostravské univerzity každoročně otevírá akreditované studium pro zájemce z řad učitelů. Je zajímavé sledovat skladbu původních aprobačních zájemců o studium. Není vůbec neobvyklé, že 40 % studentů má původně humanitní obory. Velmi často se jedná o kombinace oborů, které nemají dostatečný počet hodin v učebním plánu školy. Mnohdy jsou tito učitelé, ale velmi nepříjemně zaskočení v okamžiku, když zjistí, že informatika není pouhé ovládání počítače a práce s MS Office. Asi není potřeba pochybovat, že informatika je velmi těsně spojena s matematikou. Pomocí interview bylo zjištěno, že pouze učitelé s aprobační matematikou a fyzikou mají matematiku v oblíbenosti. Tato skupina ve vzorku studentů rozšiřujícího studia tvořila pouhých 25 % respondentů, 19 % respondentů uvedlo, že matematiku mají rádi jen částečně, ale uvědomují si její důležitou roli. Zbýlých 56 % respondentů se raději využití matematiky vyhýbá.

### 4. ZÁVĚR

Vzdělávání budoucích učitelů informatiky v současné době stojí před důležitým úkolem kvalitně připravit kvalifikované učitele, kteří informatiku na základních a středních školách budou nejen správně vyučovat, ale budou schopni reagovat na rychle se rozvíjející obor. Významnou roli při přípravě budoucích učitelů je skutečnost, že studenti již během studia by měli získat potřebné vědomosti pro přípravu metodické, ale i obsahové náplně výuky informatiky.

### 5. BLIOGRAFICKÉ ODKAZY

- [1] JAKUBČÍKOVÁ, M. *RVP, ŠVP a výuka informatiky na základní škole*, Bakalářská práce, Ostravská univerzita v Ostravě, Ostrava 2010.
- [2] MADAJOVÁ, M. *RVP, ŠVP a výuka informatiky na střední škole*, Bakalářská práce, Ostravská univerzita v Ostravě, Ostrava 2010.
- [3] FOJTÍK, R., JIRÁSKOVÁ, D. Programování na základních a středních školách. *Objekty 2009*. Gaudeamus, Univerzita Hradec Králové, 2009. s. 75-83. ISBN 978-80-7435-009-2
- [4] FOJTÍK, R., DROZDOVÁ, M. Teaching of programming at secondary schools. *ICTE 2009*. Ostrava: University of Ostrava, 2009. s. 77-81. ISBN 978-80-7368-459-4
- [5] VÍT, V. *Srovnání jazyků Objective-C a C++*, Bakalářská práce, Ostravská univerzita v Ostravě, Ostrava 2012.

# Nový spôsob hodnotenia domácich prác z programovania

Roman Horváth

Pedagogická fakulta, Trnavská univerzita v Trnave  
Priemyselná 4, P. O. BOX 9  
9018 43 Trnava  
Slovenská republika

roman.horvath@truni.sk

Veronika Stoffová

Pedagogická fakulta, Trnavská univerzita v Trnave  
Priemyselná 4, P. O. BOX 9  
9018 43 Trnava  
Slovenská republika

veronika.stoffova@truni.sk

## Abstrakt

Na Pedagogickej fakulte Trnavskej univerzity v Trnave považujeme za dôležité a správne viesť študentov k systematickej práci počas semestra. V predmetoch programovania priebežne zadávame študentom domáce práce. Na to, aby bolo toto opatrenie účinné, je potrebné vyžadovať odovzdávanie a kontrolu vypracovaných riešení domácich prác. Klasická ručná kontrola stoviek odovzdaných programov za semester je ťažko zvládnuteľná a málo efektívna. Čiastkovým riešením by mohla byť implementácia odovzdávacieho systému (ktorého základ bol položený ešte v roku 2011, avšak doteraz sa ho nepodarilo nasadiť do praxe) a zavedenie organizačných zmien v spôsobe hodnotenia riešení domácich prác.

## Abstract

Teachers at the Faculty of Education of Trnava University in Trnava consider as important and proper to lead students to systematic work during the semester. It is very common to assign homework to the students at the subjects of programming. It is necessary to require posting and checking the solutions of students' homework in order to keep this measurement effective. Standard hand-made check of hundreds programs returned from students per semester is difficult to make and also not very effective. Partial solution would be the implementation of the system helping by this task. (The foundation of such system falls back in 2011, but so far the in practice failed.)

## Kľúčové slová

vyučovanie programovania, on-line systém, tréning verbálneho prejavu študentov

## Keywords

Teaching Programming, On-Line System, Training of Verbal Expressions of Students

## 1 Úvod

Uplatňovať individuálny prístup pri hromadnom vyučovaní je pre učiteľa veľmi náročné. Zabezpečiť to, aby každý pracoval a napredoval podľa svojich možností a schopností, aby pritom uplatňoval aj svoju kreativitu, neľutoval investovaný čas a tešil sa zo svojho úspechu, si od učiteľa vyžaduje dávku pedagogického citu a didaktických schopností. Vo vzdelávacom procese zohráva významnú úlohu motivácia. Učiteľ musí motivovať študentov tak, aby sa táto vonkajšia motivácia časom zmenila na vnútornú, prípadne aby sa vytvorila vlastná „tajná“ vnútorná motivácia.

## 2 Možnosti motivácie študentov

Priama motivácia študentov a zdôrazňovanie toho aké je dôležité, aby učiteľ informatiky vedel dobre programovať, pretože to podporuje jeho prirodzenú učiteľskú autoritu obyčajne neúčinkuje. Študenti skôr rešpektujú tú možnosť, že ako dobrí programátori budú mať široké možnosti uplatnenia sa v praxi mimo rezortu školstva a získajú možnosť vykonávať zaujímavú kreatívnu prácu. Aj to je možnosťou, z ktorej môže vyplývať motivácia, i keď si študenti spočiatku nemusia uvedomovať, že na takéto uplatnenie musia investovať omnoho viac času než im ponúkajú hodiny programovania a povinnosti s nimi spojené – napríklad vypracovanie povinných domácich prác.

Často pomôže aj to, keď učiteľ študenta pochváli alebo vymenuje výsledky, ktoré dosiahli bývalí študenti buď v rámci rôznych programátorských súťaží, alebo v profesionálnom živote programátora. Aj negatívne príklady môžu mať motivačný charakter. Napríklad, keď učiteľ spomenie, ako málo učiteľov vie dobre programovať a ako málo z nich dokáže pripravovať svojich žiakov na programátorské súťaže, byť im oporou pri prekonávaní prekážok a riešení programátorských problémov.

Je vysoko pravdepodobné, že povinnosť vypracovania domácich prác zostane vždy len v pozícii vonkajšej motivácie. Je však vhodné pokúsiť sa zostaviť taký systém voľby a/alebo zadávania úloh na vypracovanie, ktorý túto činnosť spestrí, čím by ju mohol mierne prikloniť do oblasti vnútornej motivácie. Napríklad zapojiť do výberu a definovania úloh na vypracovanie aj študenta. Učiteľ by mal pripraviť množinu úloh zaradených do rôznych skupín náročnosti a študent by mal určiť len to, koľko úloh z ktorej skupiny si má vybrať na riešenie. Pri tom by bolo možné dve a viac úloh z nižšej skupiny nahradiť jednou úlohou z vyššej skupiny. (To je vhodné pre študentov, ktorí už majú určité programátorské skúsenosti a riešenia úloh z nižšej kategórie by sa im mohli zdať triviálne a primitívne.) Bola by definovaná aj skupina úloh/projektov, ktoré by vyžadovali komplexné riešenia zložitejších úloh a ktoré by sa dali na základe analýzy rozdeliť na úlohy nižšej kategórie náročnosti.

## 3 Návrh systému slúžiaceho na odovzdávanie domácich prác

Programovanie aj proces stať sa dobrým programátorom má svoje špecifiká. Podobne aj vyučovanie predmetov programovania. Rešpektovať a priamo využívať tieto špecifiká i ďalšie atribúty charakteristické pre programovanie neumožňuje ani jeden systém na riadenie výučby (Learning Management System – LMS). LMS teda nie sú „šité na mieru“ na riadenie vyučovania programovania [1, 2, 3, 4, 5, 6].

Domáce práce považujeme za dôležitú súčasť vyučovacieho procesu programovania. Ukázalo sa však, že ich kontrola sa pravdepodobne nikdy nestane rutinnou záležitosťou ako napríklad kontrola priebežných písomných prác. Prekážkou sú viaceré faktory, ktoré opíšeme neskôr. Z toho nám vyplynula požiadavka na ďalšiu zmenu vo vyučovacom procese, ktorá by mala za následok zlepšenie. Zmenami a vylepšeniami v rámci predmetov programovania sa na Pedagogickej fakulte Trnavskej univerzity zaoberáme už niekoľko rokov [7, 8, 9, 10].

Keď chcel doteraz študent odovzdať domácu prácu, musel skomprimovať priečinok s projektom, ktorý poslal v prílohe elektronickej správy. Vznikalo tu určité riziko straty správy s prílohou, ktoré sa v malej miere počas semestra vždy prejavilo. Vyučujúci musel naproti tomu balíček z prílohy každej prijatej správy uložiť na svoj disk, tam ho rozbaľiť a potom projekt otvoriť v prislúchajúcom softvérovom produkte. Potom musel každému študentovi poskytnúť spätnú väzbu o jeho postupe. Kontrola domácich prác sa tým neúmerne predlžovala a ukázalo sa, že efektívnosť v súvislosti s časovou investíciou vyučujúceho nie je vysoká.

Systém, ktorý je stále vo vývoji je prispôsobený spomenutým špecifikám tak, aby prácu študentom aj učiteľom zjednodušil. Aktualizácie študentskej časti systému sú priebežne publikované prostredníctvom webových stránok predmetov programovania [11], na ktorých je uvedený odkaz na samostatnú stránku, ktorý nevyžaduje inštaláciu. Študent sa po jeho spustení prihlási svojimi prihlasovacími údajmi, zvolí priečinok s projektom, v ktorom má uložené riešenie domácej práce a potvrdí jeho odovzdanie. Podobne učiteľ má vo svojom počítači k dispozícii nástroj slúžiaci na automatické prevzatie projektu zo servera, uloženie do jeho počítača a spustenie prislúchajúceho prostredia, v ktorom sa projekt automaticky otvorí.

Doterajšie skúsenosti s elektronickou poštou alebo pamäťovými kľúčmi ukazovali na nevýhody týchto spôsobov odovzdávania a spracúvania domácich prác. Nevýhody plynúce z používania elektronickej pošty sú opísané vyššie. Pamäťové kľúče tiež nie sú ideálnym riešením. Skúsenosti z iných predmetov ukázali, že použitie pamäťových kľúčov bolo bezproblémové iba pri malom množstve týchto zariadení používaných študentmi. Rozpoznanie väčšiny používaných zariadení na počítačoch v učebni (prípadne pracovni vyučujúceho) bolo zdĺhavé a v niektorých prípadoch sa nezdarilo. To viedlo ku zbytočným komplikáciám.

Kolegovia a organizátori rôznych súťaží nás v minulosti informovali o implementovaní podobných systémov napríklad v rámci súťaže Palma (<http://palma.strom.sk/>), kde systém dokáže overiť správnosť riešenia na vopred pripravených vstupoch. Tento systém však nebol v našom prostredí využiteľný, keďže naši študenti odovzdávali zadania grafického charakteru, ktoré nie je triviálne kontrolovať ani manuálne, nieto ešte automaticky. Práve to bol jeden z dôvodov, prečo sme sa rozhodli postupovať pri hodnotení nie hromadne, ale formou individuálnych prezentácií.

### 3.1 Pravidlá nového spôsobu hodnotenia domácich prác

Odovzdané projekty nebudú kontrolované individuálne. Systém vykoná len kontrolu duplicit. V prípade, že budú odovzdané viaceré kópie jedného riešenia, bude do úvahy vzaté len to riešenie, ktoré bolo odovzdané ako prvé.

V prípade, že študent nájde dodatočne vo svojom odovzdanom zadaní chybu, môže pridať svoje zistenie do komentára k projektu, ktoré sú nepovinné. Vyššiu váhu však bude mať osobná prezentácia riešenia priamo na hodinách. Cení sa, keď sa študent bude schopný priamo počas prezentácie opraviť. Prezentáciou zároveň študenti absolvujú tréning vystupovania pred triedou, čo budú ako potenciálni učitelia potrebovať.

Tomu, aby sa ani jeden študent do konca semestra nemohol spoliehať na to, že už raz úlohu prezentoval, má slúžiť systém náhodného losovania, ktorý každého študenta vylosuje viac ráz za semester. Vzhľadom na veľkosť skupín to bude vyžadovať aspoň dve prezentácie odovzdaných riešení počas jednej vyučovacej hodiny.

### 3.2 Dôvody, ktoré viedli k zavedeniu nového spôsobu organizovania a hodnotenia domácich prác

- Klasický spôsob zadávania, riadenia procesu tvorby riešení, opravy a hodnotenia riešení je časovo náročný proces.
- Starý spôsob nezachytával (neregistroval) implicitne cestu, ktorá viedla k prijatiu poslednej verzie riešenia úlohy.
- Jednotlivé priebežné a dodatočné zásahy sa stávali neprehľadnými.
- Pri starom spôsobe bolo značné riziko toho, že sa odovzdávaná práca stratí. (Vždy sa vyskytlo niekoľko prípadov za semester.)
- Elektronická komunikácia je neosobná (v kontraste s osobnou prezentáciou) a môže pri nej dochádzať k nedorozumeniam medzi študentom a učiteľom.
- Spracovanie prác odovzdaných v prílohách elektronickej pošty je málo flexibilné a vyžaduje od učiteľa dôslednosť a veľkú disciplínu.

### 3.3 Výhody a prednosti systému na zadávanie a odovzdávanie domácich prác (SZODP)

- Tému/problém si vyberie študent. Predpokladá sa, že jeho voľba bude niečím motivovaná.
- Riešenia sú odovzdávajú priebežne (s uvedením dátumu) do stanoveného konečného termínu.
- Študent vidí svoje odovzdané zadania, vidí dátum odovzdania a počet súborov, ktoré systém prevzal na server. Tým sa eliminujú technické poruchy súvisiace so stratami domácich prác.
- Vyučujúcemu je poskytnutý vyšší komfort pri spracovaní domácich prác. (Ten je využiteľný ako pri individuálnom spracovaní odovzdaných úloh, tak pri prezentácii náhodne vybraných riešení študentmi pred tabuľou na cvičeniach.)

### 3.4 Výhody nového spôsobu hodnotenia domácich prác

- Učiteľ je odbremený od náročnej úlohy kontroly veľkého množstva úloh.
- Študent je donútený k podobnej príprave ako pri predchádzajúcom spôsobe riešenia, dokonca by sa dalo povedať, že ak nechce zožať neúspech, tak ešte k lepšej.
- Vzniká možnosť eliminácie rizika bezmyšlienkovitého kopírovania riešení, pretože študenti musia riešeniu rozumieť na to, aby ho mohli kvalitne prezentovať.
- Študenti budú trénovať prejav pred triedou, čo je v súlade s filozofiou ich prípravy na budúce povolanie (učiteľa).

## 4 Požiadavky na SZODP v súčinnosti s novým spôsobom hodnotenia

### 4.1 Hlavné zásady

- Implementovaný systém má byť oporou ako pre učiteľa tak pre študenta.
- Spôsob hodnotenia má nadväzovať na tradíciu budovania poznatkov na základe spätnej väzby.
- Niektoré procesy majú byť v porovnaní so starým spôsobom spracovania vykonávané automaticky.
- Budovanie systému má byť dynamickým procesom. — Priebežne získané skúsenosti majú napomáhať pri postupnom vylepšovaní systému.

### 4.2 Požiadavky na vlastnosti a služby poskytované systémom

- Základnú bázu úloh vytvorí učiteľ.
- Báza úloh sa bude neustále priebežne dynamicky aktualizovať a dopĺňať.
- Spočiatku budú mať vzorové riešenia vytvorené len niektoré úlohy.
- K ostatným úlohám budú postupne priradené okomentované nápadité kreatívne riešenia od študentov, ktoré bude učiteľ považovať za vhodné na zverejnenie.

V prípade, že sa používanie softvéru osvedčí, zväžeme jeho rozšírenie tak, aby bol konfigurovateľný a použiteľný na rôznych predmetoch. V súčasnosti však musí byť vývoj rýchly a požiadavka adaptability systému by jeho vývoj spomalila.

## 5 Súčasný stav riešenia a použité prostriedky implementácie

Softvér, ktorý bude základom pre nový systém bol vyvinutý v roku 2011, ale vtedajšie nevysvetliteľné technické problémy zabránili jeho nasadeniu do praxe.

Ide o aplikáciu pozostávajúcu z niekoľkých častí. Klientsku študentskú časť naprogramovanú v Jave si študent prevezme do počítača. Klientska učiteľská časť naprogramovaná taktiež v Jave

bude prítomná v počítači učiteľa (v pracovni alebo doma) a v učiteľskom počítači v učebni.

Serverová časť naprogramovaná v PHP je uložená na jednom zo serverov, ktoré sú v správe zamestnancov Pedagogickej fakulty Trnavskej univerzity v Trnave. Predchádzajúca verzia obsahovala aj prepojenie s databázou MySQL, ktorej vylúčenie vyplýva z technických problémov opísaných nižšie.

V roku 2011 sa nepodarilo identifikovať príčinu problémov. Použitie systému v rámci vyučovacieho procesu muselo byť odvolané. Počas ďalších rokov sme museli pokračovať v starom spôsobe odovzdávania a hodnotenia domácich prác.

V poslednom semestri (zima 2014) sa však prevádzkovanie tohto spôsobu hodnotenia ukázalo byť neúnosné. Zrušiť domáce práce neprípadovalo do úvahy, preto sme sa rozhodli opraviť starý navrhnutý systém za každú cenu. Spolu s tým prišla aj idea o zmene spôsobu hodnotenia domácich prác. Softvér bude mať za úlohu uľahčiť fázu odovzdávania vypracovaných a otvárania odovzdaných domácich prác. Odľahčenie fázy hodnotenia je realizované pravidlami, ktoré sú opísané v podkapitole 3.1 (čiže na úkor individuálnej kontroly zadaní).

V roku 2011 boli webový server (ktorý sprostredkúval komunikáciu na strane servera po technickej stránke, i keď nešlo priamo o webovú aplikáciu) a databázový systém fyzicky na jednom serverovom počítači. Keď došlo k zablokovaniu serverovej časti aplikácie, nebolo možné identifikovať, ktorá časť v skutočnosti zlyhala.

Tento raz, v roku 2014, išlo o inú situáciu. Z dôvodu technických problémov musel byť v priebehu rokov databázový systém presunutý na iný server. To pomohlo k identifikácii možných príčin zlyhania predchádzajúcej verzie systému. Pri druhom pokuse o spustenie systému do prevádzky sme totiž zistili, že webová časť funguje naďalej bez problémov a k nevysvetliteľnému zablokovaniu došlo len na strane databázového systému.

V súčasnosti (2015) je systém v štádiu úplného prepracovania serverovej časti. Zdá sa, že rozhodnutie z roku 2011 o využití databázy na ukladanie údajov pre tento projekt bolo nesprávne. Databáza MySQL pravdepodobne nie je stavaná na prijímanie veľkého počtu drobných požiadaviek, ku ktorých vzniku dochádza pri prevádzke tohto systému a pravdepodobne preto došlo počas testovacích prevádzok (v minulosti aj teraz) k jej zablokovaniu.

Nová verzia systému bude založená na ukladaní a čítaní údajov priamo z pevného disku servera. Prepracovanie projektu je v štádiu rozpracovanosti a pravdepodobne potrvá niekoľko týždňov (nanajvýš). Z toho vyplýva, že zatiaľ bude žiaľ nevyhnuté to, aby odovzdávanie vypracovaných úloh zatiaľ prebiehalo starým spôsobom – prostredníctvom príloh elektronickej pošty. Avšak spôsob hodnotenia už bude podliehať novým pravidlám.

## 6 Záver

Autori sú presvedčení, že postupom času sa pri používaní systému na zadávanie a odovzdávanie domácich prác iteratívnym spôsobom vytvorí hodnotná banka riešených a neriešených úloh vhodných na použitie pri vyučovaní predmetov programovania.

Pre jednotlivé ročníky študentov bude platiť, že kým nebude mať skupina úloh na riešenie ukončený termín odovzdávania, študenti nebudú mať k dispozícii vzorové a alternatívne riešenia. Možnosť porovnania vlastného riešenia so vzorovým riešením a analýzy

a konfrontácie s alternatívnymi okomentovanými riešeniami bude otvorená až po odovzdaní vypracovaných úloh.

Aby sa študenti skutočne zaujímal o optimálne riešenia, bude medzi požiadavkami počas prezentovania vlastného riešenia zaradené aj to, aby uviedli aj možnosti jeho vylepšenia.

## 7 Bibliografické odkazy

- [1] Štofová, Veronika – Lovaszová, Gabriela: Ako viesť začínajúcich programátorov k tvorbe efektívnych programov. In *Informatika v škole*. 1992. č. 9.
- [2] Štofová, Veronika: Automation of programmer's knowledge testing. In *Acta Mathematica & Informatica I*. Nitra : Vysoká škola pedagogická, 1992, s. 97-97.
- [3] Stoffová, Veronika: *Počítač – univerzálny didaktický prostriedok*. 1. vyd. Nitra : Fakulta prírodných vied UKF v Nitre, 2003. 172 s. ISBN 80-8050-450-4.
- [4] Stoffová, Veronika: Testovanie vedomostí z algoritmickej a programovania v prostredí LMS Moodle. (Knowledge testing from algorithmisation and programming in LMS Moodle). In *XXVI. International colloquium on the Management of Educational Process : Proceeding of abstracts and electronic version of reviewed contributions on CD-ROM*. Editori Eva Hájková a Rita Vémolová. Brno. Brno : University of Defence, Faculty of Economics and Management, 2008. Abstrakt na s. 78, Celý príspevok na CD-ROM. ISBN 978-80-7231-511-6.
- [5] Stoffová, Veronika: Számítógépes programozás tudásteresztelés (Programming knowledge testing by computer). In *Zborník z I. medzinárodnej vedeckej konferencie Univerzity J. Selyeho „Vzdelanie – Veda – Spoločnosť” (A Selye János Egyetem „Oktatás – Tudomány – Társadalom” I. Nemzetközi Tudományos Konferenciájának tanulmánykötete)*. Editor Marta Nagyová. Komárno : Univerzita J. Selyeho, 2009. 85-89 s. ISBN 978-80-89234-83-7.
- [6] Stoffová, Veronika: Testovanie vedomostí v LMS Moodle. In *XXII. DIDMATTECH 2009*. Editor Veronika Stoffová. 1. vyd. Trnava – Komárno : Trnava University Trnava – J. Selye University Komárno, 2010. S. 171-181. ISBN 978-80-8122-006-7.
- [7] Horváth, Roman: Teaching one Language in More Depth is better than Many Languages Superficially. In *ICETA 2011*. Stará Lesná, The High Tatras, Slovakia : IEEE, 2011. S. 71–74. ISBN 978-1-4577-0051-4. Dostupné na: <<http://140.98.202.196/stamp/stamp.jsp?tp=&arnumber=6112588>>.
- [8] Horváth, Roman: Nový model vyučovania programovania na Pedagogickej fakulte TU. In *Acta Facultatis Paedagogicae Universitatis Tyrnaviensis*. Trnava : Trnavská univerzita, Pedagogická fakulta, 2011. S. 98–102. ISBN 978-80-8082-514-0. Dostupné na: <<http://pdf.truni.sk/actafp/2011/c/#HORVATH>>.
- [9] Horváth, Roman – Javorský, Stanislav: New Teaching Model for Java Programming Subjects. In *Procedia – social and behavioral sciences*, č. 116. 2014. S. 5188–5193. ISSN 1877-0428.
- [10] Horváth, Roman – Mišút, Martin: New way of teaching programming at university. In *XXVI. DIDMATTECH 2013*.
- Győr : University of West Hungary – Apáczai Csere János Faculty, 2014. S. 159–164. ISBN 978-963-334-184-1. Dostupné na: <<http://ukftp.truni.sk/epc/11099.pdf>>.
- [11] Horváth, Roman: *Predmety*. Webové stránky slúžiace na komunikáciu so študentmi v rámci určitých predmetov vyučovaných autorom. Trnava : 2015. Dostupné na: <<http://pdf.truni.sk/horvath/predmety>>.
- [12] Horváth, Roman: *Odovzdávač projektov BlueJ*. Webová stránka slúžiaca na prevzatie študentskej klientskej časti softvéru. Trnava : 2015. Dostupné na: <<http://pdf.truni.sk/horvath/softver?uploader>>.

# Kurz Multimédiá v LMS Moodle

Dana Horváthová  
Katedra informatiky, FPV UMB  
Tajovského 40  
974 01 Banská Bystrica  
Slovensko  
dana.horvathova@umb.sk

## ABSTRAKT

V tomto článku sa zoznámime s experimentálne overeným kurzom *Multimédiá* v LMS Moodle, v ktorom predstavíme rôzne aktivity kurzu, napríklad fórum novínok, možnosť samotestovania, a niektoré zadania úloh. Z týchto úloh ukážeme niekoľko praktických ukážok, zameraných na prácu s mediálnymi elementami (textom, obrazom, zvukom, animáciou a videom). Podelíme sa o skúsenosti s manažovaním takéhoto kurzu, s hodnotením jednotlivých úloh a využívaním niektorých služieb, ktoré systém Moodle ponúka.

## ABSTRACT

In this article we introduce experimentally verified *Multimedia* course in LMS Moodle, where we present various activities of the course, for example news forum, the possibility of self-testing, and some tasks. We show several practical demonstrations of these tasks that are designed to work with media elements (text, images, sound, animation and video). We share our experience in managing such a course, the assessment of individual tasks and the use of certain services that Moodle offers.

## Kľúčové slová

Vzdelávanie, multimédiá, praktické úlohy, mediálne elementy, LMS Moodle.

## Keywords

Education, multimedia, practical tasks, media elements, LMS Moodle.

## 1. ÚVOD

Konferencia Didinfo ponúka priestor na výmenu skúseností a vzájomné obohacovanie v oblasti vzdelávania informatiky už dvadsať jeden rokov. Je to fórum, do ktorého môže každý účastník niečo priniest a súčasne si z neho môže niečo iné odniesť.

Podobne dlho, takmer dvadsať rokov sa na katedre informatiky vyučuje predmet *Multimédiá* v rámci študijného odboru učiteľstvo akademických predmetov - učiteľstvo informatiky, aj v rámci študijného odboru aplikovanej informatiky. Založenie a vývoj tohto predmetu sa odrážali od skúsených odborníkov v tejto oblasti. V priereze týchto rokov vzniklo niekoľko článkov publikovaných autorkou, ktoré sa viac, či menej venovali problematike *Multimédií* a ktoré odznali práve na konferencii Didinfo [1,2,3,4,5,6,7,8,9,10]. V článku [10] sme sa venovali vyučbe multimédií prostredníctvom LMS systému Moodle a prepojeniu výučby študentov učiteľstva informatiky a študentov aplikovanej informatiky k vzájomnému prospechu oboch skupín. V tejto forme výučby pokračujeme aj naďalej a aj vďaka projektu RIFV<sup>1</sup> zdokonaľujeme elektronický kurz po viacerých stránkach.

## 2. PREDMET A KURZ MULTIMÉDIÁ

Kurz *Multimédiá* slúži študentom 3. ročníka aplikovanej informatiky, ale aj budúcim učiteľom informatiky ako elektronická podpora predmetu *Multimédiá*, ktorá sa osvedčila nielen v externej forme štúdia, ale pomáha aj denným študentom rýchlejšie sa dostať k študijným materiálom, jednoduchšie odovzdať vyriešené úlohy a flexibilnejšie komunikovať s vyučujúcim, prípadne medzi sebou.



Obrázok 1. Kurz Multimédiá v LMS Moodle

V rámci projektu RIFV bola prepracovaná jednotná vizuálna stránka prezentácii aj celého prostredia, niekoľko nových aplikácií, úloh, testov aj populárne fórum novínok, do ktorého môžu študenti prispievať zaujímavosťami a novinkami zo sveta multimédií, na ktoré narazia pri surfovaní po internete.



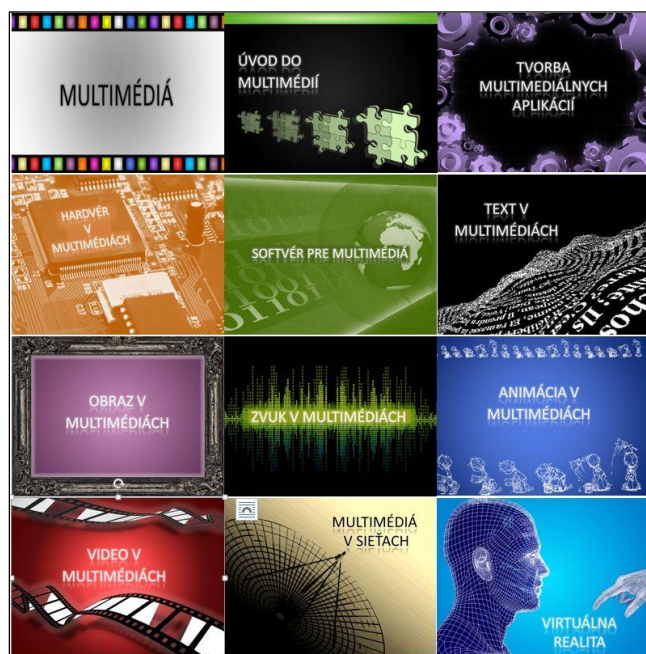
Obrázok 2. Fórum novínok

<sup>1</sup> Rozvoj inovatívnych foriem vzdelávania na UMB v B. Bystrici (Hlavným cieľom projektu RIFV je vytvoriť 280 e-learningových kurzov pre študentov UMB v Banskej Bystrici v dennej aj

externej forme štúdia a pripraviť tak jednotnú platformu tohto typu poskytovania vzdelávacieho obsahu.)



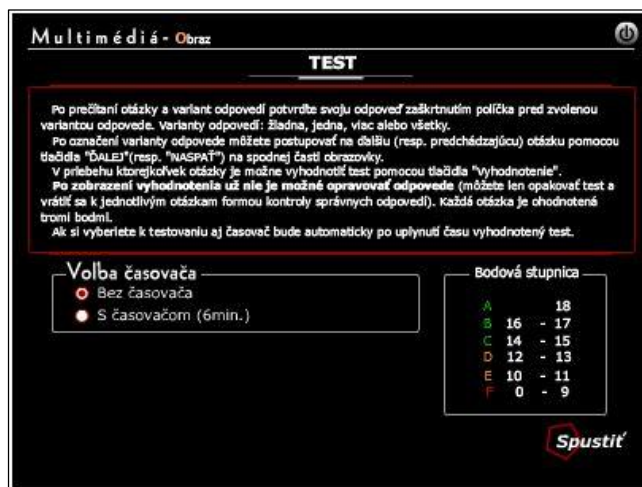
Prezentácie slúžia na zoznámenie sa s predmetnou oblasťou v priebehu celého semestra prostredníctvom 11 tematických celkov.



Obrázok 3. Vizuálna stránka prezentácií

### 3. ÚLOHY V PREDMETE MULTIMÉDIA

Kurz ponúka rôzne formy aktivít. Okrem prezentácií a množstva študijných materiálov je študentom k dispozícii ku každej téme test na overenie základných znalostí. Študent si zvolí, či sa chce otestovať s časovým obmedzením, alebo bez neho. Po skončení samotestovania dostane okamžitú spätnú väzbu s hodnotením, aj s možnosťou vidieť správne odpovede.



Obrázok 4. Test určený na samotestovanie

V priebehu kurzu má študent možnosť vyskúšať si množstvo praktických úloh. V každej úlohe je vysvetlené zadanie, navrhovaný softvér (väčšinou voľne dostupný), nápoveda, spôsob odovzdávania úlohy aj spôsob jej hodnotenia. Z dôvodu eliminácie subjektivity pri hodnotení je uvedených viacero kritérií, takže študenti dopredu vedia, na čo sa kladie dôraz. Ku každej úlohe je študentovi k dispozícii vzorové riešenie, prípadne

pri zložitejších úlohách aj návod, resp. videotutoriál s postupom riešenia danej úlohy.



Obrázok 5. Ukážka aktivít v kurse Multimédia

Študent odovzdá úlohu cez aktivitu *Odvádzanie úlohy* do určitého stanoveného termínu a do týždňa dostane spätnú väzbu o úspešnosti jej zvládnutia. V prípade, že spraví niekde chybu a nezíska plný počet bodov, v poznámke pri odovzdanom súbore dostane vysvetlenie, kde je problém. Takto postupne počas celého semestra zdoláva úlohy a získava body do záverečného hodnotenia, ktoré sa spočítajú s bodmi za záverečný test. O záverečnom teste, resp. o využití univerzálného testovacieho systému pojednáva podrobnejšie publikácia [11]. Do hodnotenia niektorých úloh účinne vstupujú študenti 5. ročníka učiteľského štúdia, ktorí sa tak zoznamujú nielen s multimediálnou problematikou, ale majú možnosť po prvýkrát používať prostredie Moodle aj zo strany učiteľa [10].

V tomto článku sa chceme podeliť o skúsenosti s praktickými úlohami, ktoré mali za úlohu preveriť základné zručnosti s prácou s mediálnymi elementami, ktorými sú text, obraz, zvuk, animácia a video.

### 3.1 Úloha Text

#### Zadanie:

- Na stránke [www.makeuseof.com/tag/create-your-own-fonts-symbols-characters-windows/](http://www.makeuseof.com/tag/create-your-own-fonts-symbols-characters-windows/) si preštudujte návod k práci s programom *Private Character Editor*.
- V editore súkromných znakov eudcedit.exe vytvorte vlastné skriptové a ozdobné písmo pre znaky [M, m, u, U, i, I, j, J, k, K, l, L, n, N, o, O, p, P, q, Q, r, R, s, S, t, T, v, V, w, W, x, X, y, Y, z, Z].
- Z mapy znakov do textového editora prekopírujte Vami vytvorené znaky tak, aby vytvorili slovo „Multimédia“ a to podľa nasledujúcich inštrukcií:  
Napište „Multimédia“ Vaším novovytvoreným skriptovým písmom, veľkosť písma 20.  
Napište „Multimédia“ Vaším novovytvoreným skriptovým písmom, veľkosť písma 48.  
Napište „Multimédia“ Vaším novovytvoreným ozdobným písmom, veľkosť písma 20.  
Napište „Multimédia“ Vaším novovytvoreným ozdobným písmom, veľkosť písma 48.  
Každý retazec umiestnite na nový riadok.
- Dokument preveďte do formátu pdf, nazvite ho ako vaše meno\_priezvisko a odovzdajte ho na Moodle cez Odovzdávanie úlohy Text.

#### Bodovanie:

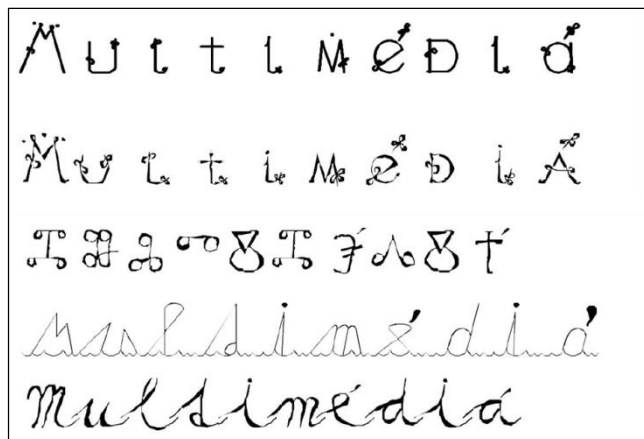
Za splnenie každej úlohy z bodu 3 dostanete 1 bod, teda 4 body.

Za zaujímavý vzhľad a celkový dojem 1 bod, spolu teda 5 bodov.

Obrázok 6. Zadanie úlohy Text

Jedna z prvých úloh je venovaná tvorbe vlastného písma v editore znakov, z ktorého sa potom v textovom editore jednotlivé písmená vyberajú a zostavujú sa potrebné slová.

Úloha nie je technicky náročná, ale vyžaduje trpezlivosť a aspoň minimálnu úroveň kreativity pri vytváraní vlastného ozdobného a skriptového písma. Stačí vytvoriť len niekoľko písmen, ktoré sú potrebné na zostavenie požadovaného slova.



Obrázok 7. Ukážky úlohy Text

Hodnotenie tejto úlohy je pomerne rýchle, pretože výsledok je viditeľný okamžite, aj kontrola splnenia stanovených kritérií nie je zaťažovaná subjektivitou.

### 3.2 Úloha Obraz

Na prácu s vektorovým alebo rastrovým obrazom už bolo vymyslených veľa úloh, preto je ťažké pre študentov pripraviť niečo, čo by ich bavilo a bolo by to pre nich primerane náročné.

Úloha Obraz zahŕňa niekoľko potrebných schopností a zručností, ktoré musí študent preukázať. Opäť je to kreativita spolu s predstavivosťou a technickou zručnosťou. Najprv treba nájsť sériu piatich spolu súvisiacich obrázkov, ktoré zachytávajú nejakú spoločnú tému. Potom je potrebné vystrihnúť jeden objekt z ďalšieho obrázku, pričom od študenta závisí, aký zložitý objekt si vyberie. Tento vystrihnutý objekt, vhodne zakomponovaný do piatich obrázkov, bude mať zakaždým inú funkciu.

**Zadanie:**  
Vytvorte sériu (piatich) tematicky spolu súvisiacich obrazov (napr. všetky budú od renesančných maliarov, alebo všetky z prostredia vojenského letiska, atď....), do ktorých zakomponujete vyrezaný objekt a zmeníte tak jeho funkciu (napr. model štruktúry mušle, ktorý bude mať funkciu prúdiaceho vzduchu, topánky, čipky, šperky...). V každom obraze bude objekt využitý s inou funkciou. Výsledný projekt môže obsahovať aj rôzne úpravy pôvodných fotografií (napr. orezanie, orámovanie obrázka a pod.), čo môže prispieť k lepšiemu hodnoteniu.

**Softvér:**  
Koláž môžete vytvárať v ľubovoľnom grafickom programe. Ak by ste sa rozhodli pre softvér GIMP, (freeware) dá sa stiahnuť [tu](http://www.gimp.org/). Návod pre používanie tohto programu sú stiahnuteľné na hore spomenutej stránke, alebo odporúčam navštíviť stránky uvedené na adrese <http://gimp-rozcestnik.ic.cz/>. Ti, ktorí obľubujú prácu vo Photoshope, alebo Zoner Photo Studio resp. iných aplikáciách slúžiacich na editáciu obrázkov a majú aplikáciu zakúpenú, môžu využiť aj tieto.

**Odpovedanie:**  
Úlohu odovzdajte v súbore formátu .zip, v ktorom budú uložené všetky vami vytvorené obrázky (vo formáte .jpg) aj s pôvodným, z ktorého bol vyrezaný objekt. Názov skomprimovaného súboru bude vaše meno\_priezvisko, ktoré pošlete cez vaše konto na Moodle do Odovzdávanie úlohy Obraz.

**Hodnotenie:**  
3body – za splnenie stanoveného počtu vložených objektov (5 obrazov (2,5b), 1 vyrezaný objekt (0,5b)) – hodnotí sa aj nápaditosť a kreativita  
1bod – za vizuálnu úpravu vyrezaného vloženého objektu (zložitost)  
1bod – za pozitívny estetický dojem  
2bod – za dostatočne dobré technické spracovanie  
Spolu: 7 bodov

Obrázok 8. Zadanie úlohy Obraz

Hodnotenie takýchto úloh je pomerne časovo náročné, pretože vyžaduje otvorenie viacerých súborov od každého študenta a posúdenie viacerých kritérií podlieha značnej subjektivite.



Obrázok 9. Ukážka zaujímavej koláže v úlohe Obraz

### 3.3 Úloha Zvuk

Zadanie úlohy Zvuk vyžaduje okrem technických zručností a kreativity aj mikrofón, od kvality ktorého do značnej miery závisí kvalita celkového výstupného súboru. Túto skutočnosť musí posudzovateľ zobrať do úvahy a zamerať sa pri hodnotení najmä na dôslednosť dodržania kritérií, pretože nie každý má k dispozícii špičkovú techniku.

**Zadanie:**  
• Vyberte si ľubovoľnú hudobnú ukážku so spevom – zvukový súbor v stereo formáte mp3.  
• Nahrajte svoj hlas – čítaný text – definíciu pojmu hudba od E. Šonara: „Hudba je umenie usporiadať v časovom postupe sluchové vnemy determinované vibráciami telesami, vytvárajúc pôžitok z vnútornej kvality týchto vnemov a z ich vzájomných vzťahov.“  
Text nahrajte tak, aby dĺžka nahrávky bola presne 20 sekúnd (Ak Vám bude čítanie textu trvať dlhšie – orežte ho. Ak bude trvať kratšie – čítajte text viackrát za sebou)  
Vytvorte skladbu ako sled nasledujúcich zvukových úsekov:  
1. **Vaša hudobná ukážka** – trvanie 20 sekúnd.  
– postupne zosilnená na začiatku (Fade In) a zoslabená na konci skladby (Fade Out).  
2. **Vaša hudobná ukážka** – trvanie 20 sekúnd.  
– „karaoke“ verzia – z ukážky odstráňte hlas.  
3. **Hlasová nahrávka** – trvanie 20 sekúnd.  
– vložte predtým „karaoke“ verziu hudobnej ukážky ako pozadie pre nahrávku vášho hlasu.  
4. **Hlasová nahrávka** – trvanie 10 sekúnd.  
– pôvodná hlasová nahrávka (bez hudobného pozadia) zrychlená na dvojnásobnú rýchlosť

**Softvér:**  
Úlohu môžete vypracovať v ľubovoľnom zvukovom editore.  
Odporúčame voľne dostupný softvér Audacity (<http://audacity.sourceforge.net/>). Pre Audacity potrebujete knižnicu lame\_enc.dll. Pri exportovaní súboru vo formáte mp3 vám Audacity automaticky ponúkne možnosť pre jeho doplnenie.

**Nápoveda:**  
Návody pre prácu s programom Audacity nájdete v kurze lms2.umb.sk medzi materiálmi k téme Zvuk: Zvuk v MUME webstránka (cvičenia, videotutorialy).  
Karaoke úpravu nie je možné vytvoriť pre každú ukážku, preto pri výbere ukážky odporúčam najprv skúsiť možnosť odstránenia vokálov (hlasu). Z niektorých skladieb sa dá hlas odfiltrovať len čiastočne. Rozdiel pri použití rovnakej metódy na rôznych skladbách môžete počuť v priložených ukážkach. Snažte sa zvoliť ukážku a metódu, tak, aby bol výsledok čo najlepšie. Existujú viaceré postupy a nástroje, ktoré môžete na túto úpravu použiť.  
Jednoduché invertovanie jedného z kanálov: [http://audacity.sourceforge.net/help/faq\\_118n?#editing&i=remove-vocals](http://audacity.sourceforge.net/help/faq_118n?#editing&i=remove-vocals)  
Pomocou karaoke nástroja ako plug-in v Audacity <http://www.youtube.com/watch?v=0RMPgcnv9XE>

**Odpovedanie:**  
Vypracované úlohy odovzdajte cez svoje konto na lms2.umb do prísušného týždňa cez Odovzdávanie úlohy Zvuk ako zvukový súbor vo formáte MP3. Súbor pomenujte v tvare: MenoPriezvisko.mp3

**Hodnotenie:** Spolu môžete získať 7 bodov.  
Vaša práca bude ohodnotená nasledovne:  
1 bod – za 1. časť  
3 body – za 2. časť  
1 bod – za 3. časť  
1 bod – za 4. časť  
1 bod – za ucelenosť celej vytvorenej zvukovej stop, presnosť trvania, originalnosť pri výbere hudobnej ukážky a jej zladenie s nahrávkou čítaného textu.

Obrázok 10. Zadanie úlohy Zvuk

Hodnotenie úlohy Zvuk vyžaduje primerané technické zariadenia (reproduktory, slúchadlá, zvukovú kartu) a je tiež pomerne časovo



náročné, pretože učiteľ musí pozorne odsledovať každú zvukovú nahrávku od začiatku až do konca. Pri hodnotení sa nám osvedčilo podrobne zapisovať pripomienky ku každému kritériu pri každej nahrávke a tie potom vložiť do poznámok v LMS Moodle.

### 3.4 Úloha Animácia

Pri zadávaní úlohy Animácia sme vychádzali zo samotného pojmu, ktorý vyjadruje vdychnutie duše, rozpochybovanie neživého objektu. Študenti majú za úlohu rozpochybovať jeden, príp. viacero neživých objektov na scéne pomocou tzv. „Stop-motion“ animácie.

#### Zadanie:

Vytvorte animáciu typu „Stop-motion“ (animácia pozostávajúca z niekoľkých fotiek reálnych ale neživých objektov, napríklad hračky, postavičky z papiera, či plastelíny a pod., v rôznych „stavoch“) bez zachytenia manipulácie rúk.

Vami vytvorená animácia musí spĺňať nasledujúce podmienky:

- musí byť nepretržitá (Looping, Repeat Forever),
- musí obsahovať vaše meno v trvaní aspoň 2 sekundy,
- ďalšie 2 sekundy musí obsahovať názov vašej animácie,
- počas celej animácie musí byť na scéne jablko, príp. iné ovocie (či už ako súčasť animácie alebo len ako kulisa v pozadí),
- animácia musí obsahovať minimálne 2 pohyblivé objekty na scéne,
- musí pozostávať aspoň zo 60 obrázkov,
- rozlíšenie výslednej animácie minimálne 640 x 480 px,
- mala by byť nápaditá a mať dej.

Animácia ako napríklad pohyb sekundovej ručičky po ciferníku a podobne nie sú akceptovateľné. Pri snímaní jednotlivých snímok rovnakej scény vytvorte rovnaké svetelné podmienky a zafixujte kameru v rovnakej pozícii.

#### Softvér:

Na vytvorenie animácie môžete použiť **MS Gif Animator** (voľne dostupný napr. na adrese: <http://www.slunecnice.cz/sw/ms-gif-animator/stahnout/>).

Fotky, ktoré budete importovať do tohto programu je vhodné najprv zmenšiť (na 640 x 480 px) a uložiť vo formáte .gif. Na toto môžete použiť **IrfanView** (voľne dostupný napr. na adrese <http://www.slunecnice.cz/sw/irfanview/stahnout/>).

#### Odvodzenie:

Výslednú animáciu odovzdajte vo formáte .gif, názov súboru: Meno\_Priezvisko (napr. Jozef\_Mirkvicka.gif). Vypracované zadanie odovzdajte cez svoje konto na lms2.umb do príslušného týždňa cez Odovzdávanie úlohy Animácia.

**Hodnotenie:** môžete získať spolu 7 bodov:

- 1 bod za odovzdanie „primerane plynulej“ animácie typu „Stop-motion“
- 1 bod za dodržanie minimálneho počtu snímok
- 2 body za minimálne 2 pohyblivé objekty na scéne,
- 2 body za nápaditosť
- 1 bod za technickú náročnosť

### Obrázok 11. Zadanie úlohy Animácia

Objekty zachytenia pomocou kamery, alebo fotoaparátu v rôznych fázach pohybu tak, aby nebolo vidieť ich ruku, prípadne ich vymodelujú z plastelíny, alebo vytvoria za pomoci nejakého kresliaceho nástroja priamo v počítači a potom vložia všetky snímky napr. do programu MS GIF Animator, kde nastaví parametre prehrávania a uložia ich do výsledného formátu.

Hodnotenie tejto úlohy, aj keď je náročné na čas, je väčšinou príjemné. Tu sa tvorivosť študentov prejaví naplno a niektoré práce sú skutočne nápadité.

### 3.5 Úloha Video

Poslednou úlohou, prezentovanou v tomto článku je úloha Video. Dnes už nie je problém zachytiť pohyb objektov v reálnom svete prostredníctvom kamery, alebo fotoaparátu, resp. mobilného telefónu. Tieto zariadenia sa stali bežnou výbavou takmer každého študenta. Práve preto, aby mali študenti aspoň základné skúsenosti s réžiou a strihom videa v nelineárnom editovacom systéme, majú v ďalšej úlohe pripraviť krátke video, zachytávajúce nejaký životný príbeh, alebo zaujímavú situáciu podfarbenú hudbou, vhodne zostrihanú, otitulkovanú s pridanými efektami.

Takáto úloha je pre študentov atraktívna, tvorivá a syntetizuje ich viaceré doterajšie skúsenosti. Jej hodnotenie je síce náročné, ale väčšinou aj zaujímavé pre učiteľa, keďže úlohy sú rôznorodé, nápadité a občas aj vtipné.

#### Zadanie:

Vytvorte krátku videosekvenciu zo svojho života, ktorá musí spĺňať nasledovné požiadavky:

- zachytáva určitý príbeh, z vášho života vystihujúci vašu záľubu, hobby, zážitok a pod.
- má minimálne 30 sekúnd (maximálne 1 minútu),
- obsahuje úvodnú snímku s názvom filmu,...
- záverečné titulky s menom autora,
- obsahuje najmenej 3 prechody a 2 vizuálne efekty,
- obsahuje zvukovú stopu (hovorené slovo, alebo hudbu),
- je uložená za pomoci vhodného kodeku tak, aby výstupný video súbor nepresiahol 20 MB.

#### Softvér:

Na vytvorenie použite Windows (Live) Movie Maker - dostupný napr.:

<http://explore.live.com/windows-live-movie-maker>, používatelia Linuxu použite PiTiVi a používatelia Mac použite iMovie.

#### Odvodzenie:

Výsledné video odovzdajte v jednom z nasledovných formátov .wmv, .mov, .avi, alebo mpg. Názov súboru bude Vaše Meno a Priezvisko. Odovzdajte aj textový súbor so scenárom vášho videa, kde bude zachytená obrazovka príslušného programu s projektom s viditeľnými všetkými stopami s prechodmi, efektami, zvukom a titulkami.

**Hodnotenie:** Spolu 7 bodov

- za príbeh 1 bod
- za dodržanie dĺžky 1 bod
- za úvodnú snímku s názvom filmu a záverečné titulky s menom autora, 1 bod
- za najmenej 3 prechody a 2 vizuálne efekty, 2 body
- za zvukovú stopu (hovorené slovo, alebo hudbu), 1 bod
- za kreativitu a originalitu 1 bod

### Obrázok 12. Zadanie úlohy Video

## 4. ZÁVER

Priprava a manažovanie kurzu *Multimédiá* je pomerne náročný a nekončiaci proces, ktorý každoročne využíva približne 50 študentov katedry informatiky na FPV UMB v Banskej Bystrici. Aj keď v oblasti vzdelávania multimédií je stále čo vylepšovať, inovovať a dopĺňať, kurz je obrovskou pomocou pre učiteľa aj študenta.

Predmet *Multimédiá* od svojho vzniku prešiel a vlastne stále prechádza neustálou inováciou. V prvých rokoch v rámci praktickej časti predmetu študenti pracovali kolektívne na tvorbe multimediálnych aplikácií rôzneho druhu a zamerania. Tak vznikali študentské práce, z ktorých mnohé sa dodnes vo výučbe využívajú. Hodnotenie výsledkov úsilia študentov nebolo však dostatočne individuálne a pritom jednotné. Preto sa postupom času vytvoril nový predmet *Multimediálna tvorba*, ktorý priamo nadväzuje na predmet *Multimédiá* a využíva v ňom získané skúsenosti a zručnosti na tvorbu multimediálnych aplikácií a videofilmov (napr. prezentačné video na hlavnej stránke FPV UMB). Do predmetu *Multimédiá* sa mohol zaviesť systém viacerých menších úloh, ktoré majú všetci študenti rovnaké a je tak ľahšie ich prácu individualizovať. Za posledné roky sa tieto úlohy stabilizovali aj čo do počtu (celkovo ich je 8) aj po stránke obsahovej. Ukážky študentských prác z úloh *Text* a *Obraz* sú súčasťou tohto článku. Ukážky úloh z tém *Zvuk*, *Animácia* a *Video* nebolo možné vložiť do článku pre ich dynamický charakter. Preto budú súčasťou prezentácie priamo na konferencii.

Kurz je po prihlásení sa oprávneným používateľom k dispozícii na stránke: <https://lms2.umb.sk/course/view.php?id=1266>.

## 5. POĎAKOVANIE

Kurz Multimédiá aj tento článok vznikol aj vďaka projektu Agentúry Ministerstva školstva, vedy, výskumu a športu SR pre štrukturálne fondy pod názvom RIFV - Rozvoj inovatívnych foriem vzdelávania na Univerzite Mateja Bela v Banskej Bystrici ITMS 26110230077 v rámci programu Moderné vzdelávanie pre vedomostnú spoločnosť, s podporou Európskeho sociálneho fondu a Operačného programu Vzdelávanie.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] TURČÁNI, M., HORVÁTHOVÁ, D.: *Tvorba multimediálnych aplikácií*. Didinfo 2000, Zborník z konferencie, Banská Bystrica, 2000, s. 30 – 32, ISBN 80-8055-363-7
- [2] TURČÁNI, M., HORVÁTHOVÁ, D.: *Nové trendy v hardvérovej a softvérovej podpore pre návrh a tvorbu multimediálnych aplikácií*. Didinfo 2001, Zborník z kolokvia o vyučovaní informatiky na stredných a základných školách, Metodické centrum Banská Bystrica, 2001, s. 96 – 100, ISBN 80-8041-374-6
- [3] TRAJTEL, Ľ., HORVÁTHOVÁ, D., HURAJ, L.: *Nové predmety pre učiteľov informatiky*. Didinfo 2001. Zborník z kolokvia o vyučovaní informatiky na stredných a základných školách, Metodické centrum Banská Bystrica, 2001, s. 30 – 34, ISBN 80-8041-374-6
- [4] HORVÁTHOVÁ, D., VÍTKO, P.: *Ako by mohli vyzerat' digitálne hodiny, alebo využitie multimédií v elektronickom vyučovaní*. DidInfo 2002. UMB FPV Banská Bystrica, 2002, s. 46 – 50, ISBN 80-8055-641-5
- [5] HORVÁTHOVÁ, D.: *Výsledky výučby predmetu Multimédia na FPV UMB*. DidInfo 2003. FPV UMB Banská Bystrica 2003, s. 45-46, ISBN 80-8055-786-1
- [6] HORVÁTHOVÁ, D., SILÁČI, J.: *Multimediálny trend vo vzdelávaní*. Didinfo 2004, Zborník z konferencie o vyučovaní informatiky na stredných a základných školách, Banská Bystrica: FPV UMB, 2004, s. 54-57, ISBN 80-8055-908-2.
- [7] HORVÁTHOVÁ, D., MARUŠKA, M., DĚDIČ, P., IŠTVÁN, J.: *Dynamické mediálne elementy*. Didinfo 2005, Zborník z konferencie o vyučovaní informatiky na stredných a základných školách, Banská Bystrica: FPV UMB, 2005, ISBN 80-8083-090-8
- [8] HORVÁTHOVÁ, D., MELICHER, M.: *Virtuálna realita v osnovách predmetu Multimédia*. Didinfo 2008, Zborník z konferencie o vyučovaní informatiky na stredných a základných školách, Banská Bystrica: FPV UMB, 2008, ISBN 978-80-8083-556-9
- [9] KRŇÁČ, M., HORVÁTHOVÁ, D.: *Ako môžeme vidieť zvuk a počuť obraz*. Didinfo 2011, Zborník z konferencie o vyučovaní informatiky na stredných a základných školách, Banská Bystrica : FPV UMB, 2011, ISBN 978-80-557-0142-4
- [10] HORVÁTHOVÁ, D.: *Prepojenie didaktiky informatiky a multimédií*. Didinfo 2012, Zborník z konferencie o vyučovaní informatiky na stredných a základných školách, Banská Bystrica: FPV UMB, 2012, ISBN 978-80-557-0342-8, str. 87-91
- [11] HORVÁTHOVÁ, D., BRODENEČ, I.: *E-learning Environment in Multimedia and Internet Technologies Teaching*. DIVAI 2012, In 9th International Conference on Distance Learning in Applied Informatics : Conference proceedings. Štúrovo, 2012, p. 137-145. ISBN 978-80-558-0092-9

# Tvorivá informatika – Prezentácie

Roman Hrušecký, Andrea Hrušecká, Monika Tomcsányiová, Ivan Kalaš

Univerzita Komenského v Bratislave, Fakulta matematiky, fyziky a informatiky,

Katedra základov a vyučovania informatiky

Mlynská dolina, 842 48 Bratislava, Slovensko

hrusecky@fmph.uniba.sk, hrusecka@fmph.uniba.sk, tomcsanyiova@fmph.uniba.sk,

kalas@fmph.uniba.sk

## ABSTRAKT

V príspevku predstavíme učebnicu Prezentácie zo série Tvorivá informatika pre druhý stupeň ZŠ, ktorá je zameraná na získanie vedomostí a zručností potrebných pri práci s informáciami a rôznych foriem ich prezentácie. Zameriame sa na priblíženie koncepcie učebnice, obsahu a navrhovaných aktivít v nej.

## ABSTRACT

In this paper we introduce a textbook *Presentations of series* Tvorivá informatika for the secondary school, which is focused on gain knowledge and skills needed to work with information and various forms of their presentation. We focus on the concept of a textbook approach, content and activities proposed in it.

## Klíčové slová

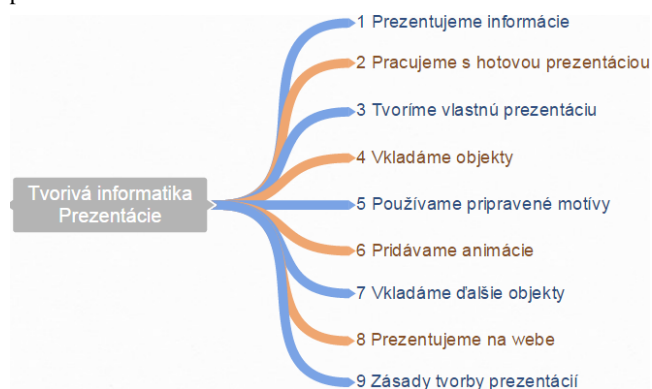
učebnica, prezentácia, bilbord, Microsoft PowerPoint, OpenOffice Impress, Google Prezentácie, Prezi

## Keywords

textbook, presentation, billboard, Microsoft PowerPoint, OpenOffice Impress, Google Presentation, Prezi

## 1. ÚVOD

Učebnica *Tvorivá informatika – Prezentácie* je súčasťou série učebníc *Tvorivá informatika* pre základné školy. Je určená žiakom 2. stupňa základných škôl a ekvivalentných tried 8-ročných gymnázií a je zameraná na získanie vedomostí a zručností potrebných pri práci s informáciami a pre rôzne formy ich prezentovania.



### Obrázok 1. Obsah učebnice Prezentácie

Učebnica je rozdelená na 9 kapitol. Jednotlivé kapitoly obsahujú množstvo nadväzujúcich úloh. Niektoré úlohy sú uvedené priamo s návodmi na ich vyriešenie, iné obsahujú „iba“ námety na samostatné aktivity.

ŠVP–Informatika pre ISCED 2 [1] je zameraný na snímkové prezentácie, preto sme väčšinu učebnice venovali práve práci so snímkovou prezentáciou. Avšak nezabúdame ani na alternatívne programy na tvorbu takýchto prezentácií a iné formy prezentovania informácií.

Rozsah učebnice pokrýva viac ako jeden ročník. Jedným z dôvodov je to, že školy môžu mať rôznu časovú dotáciu na danú tému, takisto tému prezentácie môžu zaradiť do rôznych ročníkov. Niektoré časti sú nad rámec ŠVP, bol to však zámer. Chceli sme tak dať učiteľom na školách, ktoré majú viac hodín informatiky, viac obsahu, ktorý môžu so žiakmi použiť na vyučovacích hodinách.

## 2. PREZENTUJEME INFORMÁCIE

V úvodnej kapitole popisujeme rôzne formy prezentácie informácií. Poukazujeme na to, že prezentovanie informácií neznamená nutnosť použiť prezentačný softvér (najčastejšie pracujúci so snímkami), napr. Microsoft PowerPoint, Apache OpenOffice Impress, Prezentácie Google, Prezi atď. Informácie totiž môžeme prezentovať aj formou letákov, plagátov, na bilbordoch, pomocou infografiky alebo na webových stránkach. Viacero ukážok je súčasťou priloženého CD. Niektoré úlohy v tejto časti sú praktické.

---

*Úloha:*

Zober obrázky z CD k súťaži Scratch Cup (priečinok **Scratch\_Cup**) a v grafickom editore priprav informačný leták o konaní súťaže.

- Aké informácie **musí** leták obsahovať?
- Aké doplnkové informácie **môže** obsahovať?

Predpokladaj, že leták budeš tlačiť.

- Prečo je to dôležité?
- Aké si zvolíš pozadie letáku?
- V akom formáte dáš leták k dispozícii na internet?

Iné úlohy sú zamerané na diskusie medzi žiakmi.

*Úloha (skrácená oproti učebnici):*

Pozri si infografiku [3] z CD a odpovedz na otázky:



- Koľko sekúnd sa vodiči, resp. spolujazdci pozerajú na billboard?
- Koľko slov by malo byť na bilborde?
- Čo upúta na bilborde viac ako text?

### 3. PRÁCA SO SNÍMKOVOU PREZENTÁCIOU

Ako sme spomenuli v úvode, prevažná časť učebnice je venovaná práci so snímkovou prezentáciou (MS PowerPoint, prípadne iný alternatívny program). Tvorbe a úprave takejto prezentácie v učebnici venujeme 6 kapitol.

#### 3.1 Práca s hotovou prezentáciou

V prvej časti žiaci pracujú s hotovou prezentáciou. Počas toho sa naučia ovládať prezentáciu, opravovať chyby v textoch, prechádzať po snímkach, spúšťať a ukončiť prezentáciu.

**Úloha:**

Zobraz druhú snímku prezentácie. Hneď v prvom slove je chyba. Nájdi ju a oprav.

Ktoré klávesy používame na mazanie znakov?

V ďalších kapitolách žiaci vytvárajú svoju (novú) prezentáciu a postupne sa na nej učia nové možnosti prostredia, vďaka ktorým svoju prezentáciu stále vylepšujú. Najprv vkladajú texty, kopírujú z textového editora, formátujú texty (opakujú si vedomosti získané pri práci s textovým editorom), vkladajú nové snímky, menia rozloženie snímkov.

**Úloha:**

Do snímky s nadpisom 1999 skopíruj príslušné texty z textového dokumentu tak, aby odrážky boli v jednom stĺpci a otázky s odpoveďou v druhom.

**1999**

- Bluetooth
- oficiálne spustený vyhľadávač Google
- e-mailový vírus Melissa

Koľko počítačov napadol vírus Melissa?

viac ako 1 milión počítačov

Akú škodu spôsobil?



**Obrázok 2. Úprava skopírovaného textu z textového dokumentu v prezentácii**

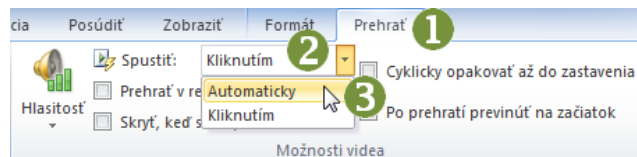
#### 3.2 Vkládanie objektov do prezentácie

Dve kapitoly sú venované vkladaniu rôznych objektov do prezentácie. Najprv ukazujeme vkladanie základných objektov (rôzne obrázky, tvary, video a zvuk) a až neskôr v inej kapitole predstavujeme vkladanie iných (zložitejších) objektov (tabuľky, odkazy...).

Prvá z týchto kapitol sa venuje vkladaniu obrázkov, ClipArt-ov, tvarov, videa a zvuku zo súborov. Týmto objektom potom žiaci nastavujú základné vlastnosti, prípadne ich umiestňujú v rámci snímkov.

**Úloha:**

Uprav video tak, aby sa spustilo automaticky.



**Obrázok 3. Ukážka postupu riešenia úlohy**

V ďalšej kapitole ukazujeme ako vkladáť do prezentácie ďalšie objekty, napríklad hotové snímky z inej prezentácie, tabuľky, grafy či hypertextové odkazy. Pri vkladani snímkov z inej prezentácie sa žiaci naučia aj meniť poradie snímkov, pričom môžu využiť aj iné zobrazenie prezentácie. Podobne ako môžeme meniť poradie snímkov, môžeme presunúť aj ľubovoľný objekt z niektorej snímky na inú.

V učebnici ďalej prezentujeme vkladanie tabuliek do snímkov. Zameriavame sa na základnú možnosť – vytvoriť a vložiť tabuľku priamo na snímke. Žiaci vložia jednoduchú tabuľku, ktorej nastavia farebný motív, podobne ako pri snímkach. Takisto sa naučia vkladáť a mazať riadky a stĺpce tabuliek.

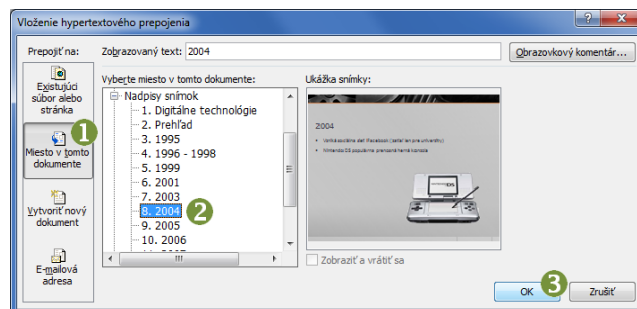
**Úloha:**

Vyskúšaj aj ďalšie možnosti zo skupiny **Štýl tabuľky – možnosti**. Čo to robí s tabuľkou? Kedy je vhodné jednotlivé možnosti použiť a kedy nie?

Posledným objektom, ktorý sa žiaci naučia vkladáť, sú hypertextové odkazy. Ukazujeme vloženie odkazu nielen na webovú stránku a existujúci súbor v počítači, ale aj na iné snímky v rámci prezentácie.

**Úloha:**

Na druhej snímke vytvor prepojenie na snímku s nadpisom 2004.



**Obrázok 4. Vkládanie odkazu na snímku**



**Úloha:**

Prepojenia na snímky, ktoré sme robili doteraz, spôsobili, že sme preskočili niekoľko snímkov. Navrhni spôsob ako sa zo snímkov vrátiť priamo na prehľad.

### 3.3 Zmena motívu prezentácie

Ďalšou funkciou, ktorá sa často využíva pri prezentáciách, je zmena vzhľadu celej prezentácie. Každý si totiž chce prispôbiť prezentáciu podľa seba, resp. nejakých iných požiadaviek. Možností, ako to doceliť, je niekoľko. V učebnici sme sa rozhodli predstaviť len použitie preddefinovaných farebných motívov. Je to asi najrýchlejší spôsob, ako oživiť prezentáciu. Toto je obsahom kapitoly 5 – *Používame pripravené motívy*. Okrem zvolenia základného motívu predstavujeme aj možnosť zvoliť si farebnú variáciu motívu, ktorých každý motív obsahuje niekoľko.

### 3.4 Pridávame animácie

V tejto kapitole učebnice sa venujeme prechodom medzi snímkami a nastavovaniu vlastných animácií objektom na snímkach. Pri prechodoch medzi snímkami sa žiaci naučia nastavovať základné vlastnosti prechodov.

Problematika s vlastnými animáciami objektov je už rozsiahlejšia. Ukazujeme definovanie len jednej animácie pre objekt. Vysvetľujeme rozdiel animácií objektov pri ich zobrazení, zvýraznení a opustení stránky. Poukazujeme na to, že niektoré efekty môžu mať aj ďalšie nastavenia, ktoré si musia žiaci pozrieť pri konkrétnom efekte. Keďže na snímkach môžeme mať viacero objektov s animáciami, žiaci sa naučia riadiť poradie týchto animácií. Takisto sa naučia nastavovať spôsob spúšťania animácií.

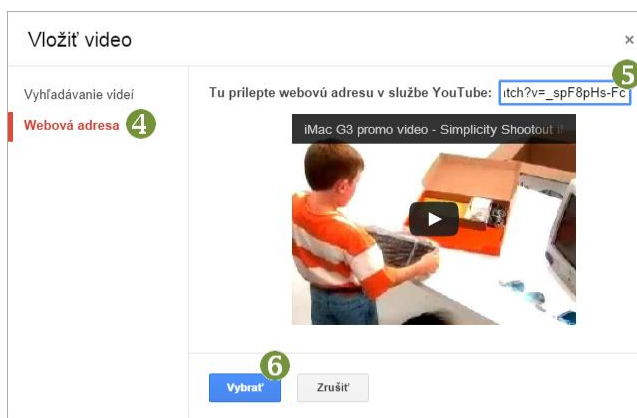
## 4. PREZENTUJEME NA WEBE

V tejto kapitole predstavujeme formy prezentovania na internete. Do prezentovania na webe patrí aj tvorba webových stránok, ktorej sa však v tejto učebnici nevenujeme, lebo táto téma je súčasťou učebnice *Tvorivá informatika – 2. zošit s internetom*. My sa zameriavame na vytváranie snímkových prezentácií priamo na internete (napr. Google Prezentácie na adrese [docs.google.com](https://docs.google.com), Office Online na adrese [office.com](https://office.com)) a vytváranie prezentácií pomocou on-line verzie Prezi ([prezi.com](https://prezi.com)).

### 4.1 Online webové dokumenty

V tejto kapitole predstavujeme vytváranie snímkovej prezentácie na internete v Prezentácie Google. Prezentácie na internete majú niekoľko výhod oproti programom na počítači – žiaci ich môžu vytvárať a upravovať z ľubovoľného počítača pripojeného na internet, môžu ich zdieľať so spolužiakmi alebo inými ľuďmi, ktorí si ich nemusia len pozerať, ale môžu ich spolu s nimi aj vytvárať a upravovať.

Keďže základné funkcie v Google Prezentácii sú veľmi podobné funkciám v MS PowerPoint, nevysvetľujeme ich podrobne. Pri práci s Google Prezentáciou sme zvolili iný prístup ako v prípade MS PowerPoint. Keďže predpokladáme, že žiaci už majú vytvorenú nejakú prezentáciu, ako prvé ju nahrajú a skonvertujú do Google Prezentácie. Tu poukazujeme na možné riziká pri prenose. Po prenose si prezentáciu žiaci spustia a otestujú, čo všetko a ako sa prenieslo. Či všetko z pôvodnej prezentácie funguje správne alebo nie. Postupne žiaci skontrolujú a prípadne nastavujú prechod medzi snímkami a nahradia pôvodné video zo súboru videom zo servera YouTube.



Obrázok 5. Vkládanie videa zo servera YouTube

V ďalšej časti vysvetľujeme veľmi užitočnú funkciu online webových dokumentov – zdieľanie. Vysvetľujeme a ukazujeme ako nastaviť zdieľanie, komu a čo môžeme dať zdieľať.

**Úloha:**

Do spolužiakovej prezentácie, ktorú ti sprístupnil, pridaj záverečnú snímku s rozložením *Úvodná snímka* a nadpisom *Autori*. Vlož na ňu mená všetkých autorov prezentácie. V prípade potreby uprav veľkosť písma.

### 4.2 Prezi

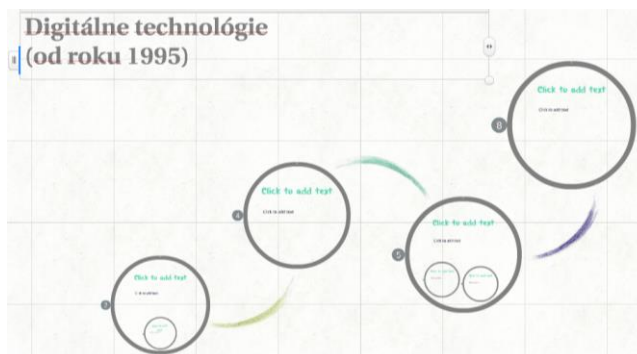
Inou formou internetovej prezentácie je Prezi [4]. Je to online nástroj na tvorbu nelineárnych prezentácií. Proces tvorby Prezi prezentácií môžeme prirovnať k tvorbe myšlienkových máp. Prvou úlohou tejto časti je porovnať snímkové prezentácie a Prezi prezentácie.

**Úloha:**

Pozri si prezentáciu na stránke [prezi.com/3ofglbdapb37/historia-pocitacov/](https://prezi.com/3ofglbdapb37/historia-pocitacov/) alebo ďalšie prezentácie na stránkach [prezi.com](https://prezi.com) v časti Explore.

- Líšia sa prezentácie od tých, s ktorými sme pracovali doteraz?
- Aké sú najväčšie rozdiely?
- Čo ťa na týchto prezentáciách najviac zaujalo?

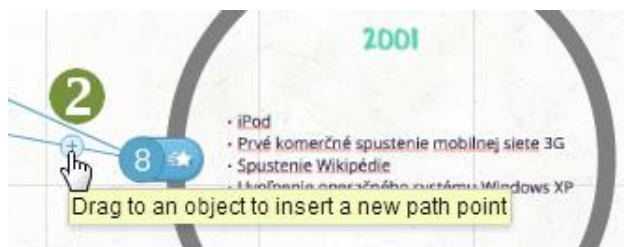
Celá prezentácia sa vytvára v rámci jednej plochy (plátna), čo umožňuje mať nad prezentáciou nadhľad a vidieť súvislosti. V Prezi vytvárame tzv. rámy (frames), ktoré medzi sebou prepájame, a vytvárame tak cestu prezentácie.



Obrázok 6. Náhľad Prezi prezentácie

Asi najväčšou nevýhodou Prezi je, že nie je v slovenskom jazyku, takže môžeme použiť len anglický, prípadne nemecký či maďarský jazyk.

Pri práci s Prezi žiaci začnú vytvárať novú prezentáciu podľa existujúcej šablóny. Pomocou šablóny sa žiaci lepšie oboznámia s princípom prezentácie v Prezi, ktorá je zásadne iná ako snímková prezentácia. Žiaci postupne vytvoria niekoľko rámov s nadpismi, obsahmi, vložia do nich obrázky aj video a takisto aj nový rámec. Naučia sa prezentáciu spúšťať a ovládať. Keďže sme použili existujúcu šablónu, ktorá obsahuje definovanú cestu medzi rámcami, žiaci sa naučia túto cestu meniť. Takisto sa naučia meniť štýl rámca.



Obrázok 7. Vkládanie nového rámcu do cesty prezentácie

## 5. ZÁSADY TVORBY PREZENTÁCIÍ

V poslednej kapitole sa venujeme rôznym zásadám pri tvorbe prezentácie.

Na priloženom CD sme pripravili niekoľko rôznych prezentácií, o ktorých majú žiaci diskutovať, hodnotiť ich podľa nimi definovaných kritérií, hľadať chyby, ale aj pozitívne veci.

### Úloha:

Na priloženom CD v priečinku **prezentacie** nájdeš rôzne prezentácie. Pozri si niektoré z nich. Všimaj si farebnosť, veľkosť písma, množstvo textu, použité prvky, animácie...

- Čo sa ti na prezentáciách páčilo?
- Čo sa ti na prezentáciách nepáčilo alebo ti vadilo?
- Boli prezentácie čitateľné?
- Diskutuj o prezentáciách a svojich zisteniach so spolužiakmi v triede.

### Úloha:

Spolu so spolužiakmi definuj kritériá, podľa ktorých budete prezentácie hodnotiť. Vytvor tabuľku, do ktorej budeš značiť, či prezentácia dané kritérium spĺňa alebo nie.

Na základe tabuľky urč, aké najčastejšie chyby si objavil v prezentáciách.

V kapitole tiež popisujeme zásady, ktoré by sme mali dodržiavať pri tvorbe prezentácie, ale aj pri samotnom prezentovaní. Uvádzame všeobecné zásady, ale aj zásady pri tvorbe elektronickej prezentácie.

## 6. ZÁVER

Prispievame do sady učebníc, ktoré sú zamerané na vyučovanie informatiky pre žiakov na 2. stupni základnej školy.

Učebnica oboznamuje žiakov s tým, ako pracovať v rôznych programoch, ktoré umožňujú prezentovať informácie. Nedávame námety na to, čo prezentovať, pretože vieme, že učitelia rôznych predmetov nájdu pre svojich žiakov množstvo tém, ktoré môžu spracovať a prezentovať pred spolužiakmi alebo na rôznych súťažiach.

## 7. BIBLIOGRAFICKÉ ODKAZY

- [1] STATPEDU. ŠVP – Informatika, príloha ISCED 2 [online]. [cit. 2015-02-02] Dostupné na: <[http://www.statpedu.sk/files/documents/svp/2stzs/isced2/vzdelavacie\\_oblasti/informatika\\_isced2.pdf](http://www.statpedu.sk/files/documents/svp/2stzs/isced2/vzdelavacie_oblasti/informatika_isced2.pdf)>.
- [2] Kalaš, I. a kol.: *rukopis učebnice Tvorivá informatika – Prezentácie* (v tlači)
- [3] euroAWK. *Tajomstvo billboardu* [online]. [cit. 2015-02-02] Dostupné na: <[www.euroawk.sk/uploads/document/1364999798-euroAWK\\_infografika.png](http://www.euroawk.sk/uploads/document/1364999798-euroAWK_infografika.png)>.
- [4] *Prezi* [online]. [cit. 2015-02-02] Dostupné na: <[prezi.com](http://prezi.com)>.

# Automatizovaná príprava hlasového korpusu

Milan Hudec

Inštitút matematiky a informatiky FPV

UMB

Cesta na amfiteáter 1

974 01 Banská Bystrica

Slovenská republika

mhudec@savbb.sk

## ABSTRAKT

Propagácia vedy pre mladú generáciu sa stala súčasťou edukačného procesu na stredných školách. Automatizovaný systém generujúci hlasový korpus umožní študentom vyskúšať si nové, zaujímavé zručnosti z oblasti umelej produkcie reči.

Softvérový automat, generujúci hlasový korpus, pracuje na báze spektrálnej a keprálnej analýzy ľudského hlasu. Pri príprave syntetickej reči vykoná zložité, odborné úpravy signálu. Študent vyhotoví len krátku nahrávku vzoriek svojho hlasu a syntetizér začne jeho hlasom rozprávať.

## ABSTRACT

Publicity of science for younger generation has become part of the educational process in secondary schools. Automated system generating speech corpus will enable students to try new as well as interesting skills in the area of artificial speech production.

Software automat generating speech corpus works on the basis of spectral and cepstral analysis of the human voice. While preparing synthetic speech, it realizes complex, special editing of the signal. The student's part is to create short recording samples of his/her voice and the synthesizer will speak with his/her voice.

## Kľúčové slová

edukácia, motivácia, hlasový korpus, unit selection, spektrálna analýza, keprálna analýza

## Keywords

education, motivation, speech corpus, unit selection, spectral analysis, cepstral analysis

## 1. ÚVOD

Návrh a vytvorenie hlasového korpusu pre konkatenáčny syntetizér<sup>1</sup> je pomerne zdĺhavý proces vyžadujúci v danej oblasti odbornosť a zručnosť. Neskôr pri umelej syntéze reči už nie je celkom zrejmé, z akých základných stavebných kameňov je reč produkovaná.

Softvérový automat na generovanie hlasového korpusu výrazne skracuje čas prípravy hlasového korpusu. Zároveň je však aj prínosom v oblasti stredoškolského vzdelávania. V rámci edukačného procesu si študent môže navrhnúť malú, čiastkovú databázu napr. difónov<sup>2</sup> svojho hlasu a spracovať ju pomocou

softvérového automatu. Syntetizér by potom hlasom študenta povedal napr. „ahoj“.

Jednoduché porozumenie podstaty konkatenáčnej syntézy a jednoduchá príprava čiastkového hlasového korpusu môže byť zaujímavou motiváciou pri výbere vysokej školy a neskôr zamestnania.

V tomto článku ukážeme softvérovú technológiu, ktorá umožní automatizované generovanie hlasového korpusu pre konkatenáčnu syntézu „unit selection“.

## 2. VYTVORENIE NAHRÁVKY VZORIEK

Prvým krokom smerujúcim ku generovaniu hlasového korpusu je vytvorenie nahrávky so vzorkami hlasu. Databáza vzoriek môže byť neúplná, syntetizér bude potom obmedzený len na produkciu niekoľkých, vopred určených viet.

Hlasové vzorky môže študent načítavať priamo do mikrofónu svojho notebooku, čím vytvorí nahrávku v súbore typu WAV.

Takáto nahrávka sa stane vstupom pre softvérový automat, ktorý z nej vygeneruje hlasový korpus.

Keďže ide o syntézu „unit selection“, databáza hlasových vzoriek môže obsahovať rôzne typy stavebných kameňov reči ako napr. fonémy<sup>3</sup>, difóny, trifóny<sup>4</sup>,...

Preprocesor softvérového automatu nájde v nahrávke vždy jednu vzorku, ktorá nesmie obsahovať medzery. Automatizovaný systém priradí k danej vzorke vektor indexov, intenzít, formantových frekvencií<sup>5</sup>... Vzorku spolu s jej popisným vektorom ďalej nazývame hlasový objekt. Na základe vygenerovania tohoto vektora môže byť objekt zaradený do hlasového korpusu.

Vytvorenie hlasového korpusu objektov rozširuje štandardnú syntézu „unit selection“ a budeme ju ďalej nazývať „object selection“.

Jednou z výhod syntézy „object selection“ je možnosť modifikácie hlasu už v existujúcom vytvorenom hlasovom korpuse. Je preto využiteľná pri výučbe študentov a umožňuje dodatočné opravy kontrolovateľné pomocou sluchu.

<sup>1</sup> Syntetizér produkujúci reč na základe reťazenia stavebných jednotiek reči, ktoré sú uložené v tzv. hlasovom korpuse.

<sup>2</sup> Stavebné jednotky reči reprezentujúce fonetický prechod dvojíc hlások - koartikulačný prechod.

<sup>3</sup> Stavebné jednotky reči reprezentujúce jednotlivé hlásky, každý foném môže mať veľa prozodických tvarov.

<sup>4</sup> Stavebné jednotky reči reprezentujúce dva fonetické prechody medzi tromi hláskami a fonetické vzájomné ovplyvňovanie dvoch krajných hlások - mikrokoartikulačný jav.

<sup>5</sup> Rezonančné frekvencie vznikajúce v hrdelnej, ústnej a nosnej dutine, sú vyššie ako frekvencia základného hlasivkového tónu.

### 3. ANALÝZA HLASOVÝCH JEDNOTIEK

Elementárnou operáciou softvérového automatu generujúceho hlasový korpus objektov je teda priradenie popisného vektora k danej vzorke – vytvorenie hlasového objektu. Táto operácia vyžaduje vykonanie analýzy hlasovej vzorky – jej krátkodobé spracovanie.

Vzorky sú analyzované posuvom Hammingovho okienka<sup>6</sup> [2,4] s 50 percentným prekryvom. V každom kroku posuvu sa vykonáva spektrálna analýza pomocou rýchlej Fourierovej transformácie [1,4], v prípade ďalšieho jemnejšieho doladenia sa použije analýza kepstier<sup>7</sup>.

#### 3.1 Spektrálna analýza hlasovej vzorky

Spektrálna analýza vzorky je efektívna pri oddeľovaní znelých hlások od šumu, frikativ a neznelých záverových hlások alebo pri oddeľovaní samohlások a znelých záverových spoluhlások.

Keď máme Hammingovo [2,4] okienko dĺžky  $2N$ , kde  $N$  je mocninou dvojky, dĺžka okienka v mocninách dvojky je nutnou podmienkou pre použitie rýchlej Fourierovej transformácie [1,4] (ďalej RFT). Vstupný signál pulznej kódovej modulácie [2,3,4] (ďalej PCM) sa teda nachádza v poli  $0..2N-1$ . Prevedieme nižšie uvedený algoritmus:

1. Pole so signálom PCM  $0..2N-1$
2. Vlož pole signálu do reálnych zložiek komplexného poľa
3. Imaginárnym zložkám priradiť nuly
4. Preveď RFT
5. Vo výstupnom poli bude uložené komplexné spektrum
6. Z indexov  $1..N$  vyberieme komplexné spektrum dĺžky  $N$
7. Vypočítame vzdialenosti komplexných hodnôt od nuly
8. Získali sme reálne výkonostné spektrum

Výstupom RFT je komplexný vektor, ktorý má na indexe nula hodnotu vyjadrujúcu priebeh časti periódy, ktorá pre svoju nízku frekvenciu nemohla byť v okienku umiestnená v celej šírke. Ide o príliš hlboké frekvencie, ktoré sa pre malú dĺžku okienka už analyzovať nedajú.

V časti výstupného vektora RFT ohraničenej indexami  $1..N$  sa nachádza komplexné spektrum analyzovaného okienka.

Časti  $1..N$  a  $N..2N-1$  sú symetrické podľa indexu  $N$ , teda dvojice: 1 a  $2N-1$ , 2 a  $2N-2$ , 3 a  $2N-3$ , atď...

Pre ďalšiu spektrálnu a kepstálnu analýzu sa používa komplexné spektrum – indexy  $1..N$ , ktoré sa prevedie do komplexného poľa dĺžky  $N - 0..N-1$ .

Ak vypočítame v komplexnom spektre vzdialenosť komplexných hodnôt od nuly, dostávame reálny vektor – výkonostné spektrum. Jednotlivé položky tohoto poľa vyjadrujú vybudenie v oblasti danej frekvencie.

<sup>6</sup> Malá časť postupnosti hodnôt pulznej kódovej modulácie - mikrosegment - ktorý reprezentuje spracúvaný zvuk - niekoľko desiatok milisekúnd. Hodnoty zvuku sú na krajoch tohoto mikrosegmentu zatlmené.

<sup>7</sup> Vektor získaný logaritmicizáciou vektora spektra po zložkách. Rozdeľuje sa na zvukové kanály pomocou MEL-filter matice z ktorých sa získavajú tzv. kepstálne MEL-filter koeficienty aplikovaním kosínusovej transformácie.

Ak okienko prenáša maximálnu frekvenciu FM, prvý index výkonostného spektra odovzdáva vybudenie frekvencie FM/N, druhý index  $2FM/N$ , tretí  $3FM/N$ , ...,  $N$ -tý  $NFM/N = FM$ .

Výkonostné spektrum je aditívne, pri sledovaní vyšších frekvencií je potrebné radiť hodnoty spektra do skupín. Tieto skupiny sú tým väčšie, čím je vyššia sledovaná frekvencia. V danej skupine sa počíta priemer zachytených hodnôt.

Vyššie uvedený algoritmus funguje ako indikátor pásmového ekvalizéra.

Energetickým porovnaním lokálnych extrémov výkonostného spektra získame pre dané okienko frekvenciu základného hlasivkového tónu a tiež aj energeticky nižších a frekvenčne vyšších formantových charakteristík.

Výkonostné spektrum je zároveň vstupom pre ďalšiu analýzu kepstra, ktorá je opísaná v ďalšej podkapitole.

#### 3.2 Kepstrálna analýza hlasovej vzorky

Pre vygenerovanie kepstra je potrebné najskôr vytvoriť MEL-filter maticu [5] (ďalej MF matica), nazvanú podľa mien jej prvých autorov. Dĺžka riadku matice sa rovná dĺžke výkonostného spektra =  $N$ , počet riadkov je daný počtom požadovaných frekvenčných kanálov. Frekvenčné kanály budú vyjadrovať hodnoty výkonostného spektra neaditívnym spôsobom.

Rozdelenie aditívneho výkonostného spektra sa môže vykonať rôznymi spôsobmi, vhodnými pre danú analýzu. Každý kanál teda definuje interval indexov do výkonostného spektra, čím je určená skupina aditívnych hodnôt tohoto spektra. MF matica má nasledovný tvar, pričom  $S_x$  vyjadrujú stredné frekvencie daného kanála a v MF matici majú hodnotu jedna.  $Z_x$  a  $K_x$  sú začiatky a konce daného kanála, od strednej frekvencie k nim hodnoty lineárne klesajú k nule. Zvyšné hodnoty MF matice sú nulové.

Tabuľka 1. MEL-filter matica

	0	1	2	3	4	5	6	7	8	...	N-
1	$Z_1$	$S_1$	$K_1$	0	0	0	0	0	0	...	0
2	0	$Z_2$	$S_2$	?	$K_2$	0	0	0	0	...	0
3	0	0	$Z_3$	?	$S_3$	?	?	$K_3$	0	...	0
...											

Všimnime si, že keď vynásobíme pole výkonostného spektra po zložkách s riadkami MF matice, dostaneme v nej zastúpenie aditívnych zložiek spektra pre jednotlivé frekvenčné kanály. Po krajoch má každý kanál lineárny úbytok energií až po nulu. Takže ak po zložkách sčítame riadky MF matice, dostaneme opäť výkonostné spektrum.

Pretože kanály s vyššou frekvenciou zachytávajú širší interval zložiek aditívneho spektra, je potrebné vykonať na riadky MF matice normalizačný algoritmus, ktorý tým viac zníži hodnoty daného kanála, čím je jeho interval aditívnych hodnôt väčší.

Majme normalizovanú MF maticu a výkonostné spektrum dĺžky  $N$ . Nižšie uvedený algoritmus z týchto hodnôt vygeneruje tzv. MEL-filter logaritmické kepstum.

```

1 minimum:=1e-4; {Minimum, aby sa nelogaritulovala
nula.}
2 for j:=1 to početkanálov do {Po riadkoch
MFmatice}
3 begin

```

```

4   for i:=0 to N-1 do {Po položkách výkonostného
spektra}
5   if MFmatica[j,i] * spektrum[i] < minimum then
6     f[i] := ln(minimum) {Do pola F ln z minima}
7   else
8     f[i] := ln( MFmatica[j,i] * spektrum[i] );
9     {V poli f máme logaritmizované aditívne
hodnoty}
10  spektra pre daný kanál, vykonáme rýchlu
kosinusovú
11  transformáciu a výsledok uložíme do
príslušného riadku matice kepstra.}
12  rkt( f, kepstrum[j], N )
13 end; {Matica MEL-filter logaritmického kepstra
vytvorená}

```

Presnosť kepstrálnej analýzy sa priamo úmerne zvyšuje s počtom frekvenčných kanálov. Počet kanálov by ale nemal presiahnuť hodnotu  $N/4$ .

Vlastnosťou rýchlej kosinovej transformácie [2] (ďalej RKT) je presun hodnôt v danom vektore na jeho začiatok. Prakticky po prevedení RKT budú na začiatku veľmi vysoké hodnoty, na konci budú hodnoty blížiac sa nule. Pre túto vlastnosť sa RKT používa aj pri stratovej kompresii dát. Posledné skaláre vektora sa pri kompresii stratia a pri rekonštrukcii dát sa nahrádzajú nulami.

V našom prípade sa esenciálna hodnota takto nachádza na začiatku vektora. Môžeme teda zobrať prvý stĺpec z kepstrálnej matice. Hovoríme o kepstrálnych koeficientoch, ktorých je toľko ako počet zvolených frekvenčných kanálov. Kvôli presnosti sme potrebovali čo najviac kanálov, teda aj kepstrálnych koeficientov. Druhá polovica koeficientov však už zachytáva pre ľudský hlas príliš vysoké frekvencie. Preto sa pri analýze môže vynechať.

Ak bolo frekvenčné spektrum rozdelené napr. na 30 kanálov, pre analýzu ľudského hlasu nám bude postačovať prvých 15.

Uvažujme o dvojhlaske „ae“. Ide o dva znelé a podobné zvuky, preto je dosť ťažko zistiť automatizovaným spôsobom, kde začína koartikulačný prechod a kde sa končí.

Načítajme zo začiatku dvojhlasý niekoľko okienok tak, aby bolo časovo zrejmé, že sa nachádzame ešte len v samohláske „a“. Z každého kepstrálneho koeficientu ukladajme maximá a minimá. Dostaneme 15 kepstrálnych intervalov pre „a“ a ak takúto operáciu vykonáme od konca aj pre „e“, dostaneme ďalších 15 intervalov pre „e“.

Takto dostaneme kepstrálny odhad farby zvuku „a“ a „e“ pre daného rečníka, ktorý načítaval vzorku „ae“.

Ak sa potom načítavaním ďalších okienok budeme približovať ku koartikulačnému prechodu, kepstrálne koeficienty sa dostanú extrémne mimo nášho kepstrálneho odhadu práve na mieste, kde sa zvuk začne meniť. To sa udeje práve na začiatku koartikulačného prechodu.

Ak kepstrálny koeficient spadá do intervalu, parciálna chyba je nulová. Ak sa koeficient dostane mimo intervalu, parciálna chyba sa rovná vzdialenosti koeficientu od hranice intervalu. Sumu parciálnych chýb delenú ich počtom budeme nazývať kepstrálnou chybou.

Dostávame takto jediné reálne číslo, ktoré je pri kepstrálnej analýze potrebné sledovať, na základe čoho môžeme automaticky oddeliť lokálne stacionarity od koartikulačných pásiem a tieto údaje vložiť do vektorov hlasových objektov.

## 4. VYTVÁRANIE HLASU V RUKÁCH ŠTUDENTOV

Plne automatizovaný systém spracovania hlasových vzoriek končiaci generovaním hlasového korpusu vykoná všetky zložité operácie bez potreby odborných znalostí používateľa.

Ak teda študent načíta podľa predlohy kompletný materiál potrebný pre úplnú syntézu „text to speech“, môže svoju farbu hlasu použiť na odovzdávanie niektorých informácií medzi počítačom a človekom.

V rukách rodiča sa takýto systém stane zaujímavým „archívom“, pomocou ktorého si môže uchovať napr. hlas svojho dieťaťa a vracat' sa k nemu aj po mnohých rokoch. Hlas pritom nebude uchovaný ako nahrávka, ale ako hlasový korpus. Na základe toho počítač dokáže čítať texty uchovaným detským hlasom.

## 5. BIBLIOGRAFICKÉ ODKAZY

- [1] PRESS, William H., FLANNERY, Brian P., TEUKOLSKY, Saul A., VETTERLING, William T. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2002. pp. 949. ISBN 0 521 43108 5.
- [2] PSUTKA, J., MULLER, L., MATOUŠEK, J., RADOVÁ, V. *Mluvíme s počítačem česky*. Praha : Academia, 2006. s. 752. ISBN 80-200-1309-1.
- [3] UHLÍŘ, J., SOVKA, P., POLLÁK, P., HANŽL, V., ČMEJLA, R. *Technologie hlasových komunikací*. Praha : Nakladatelství ČVUT, 2007. s. 276. ISBN 978-80-01-03888-8.
- [4] HUDEC, Milan. *Naprogramujte si svoj rečový syntetizér s neuronovou sieťou*. Banská Bystrica : Inštitút matematiky a informatiky FPV UMB, 2013. 228 s. ISBN 978-80-557-0495-1.
- [5] TYAGI, Vivek, WELLEKENS, Christian. On desensitizing the Mel-Cepstrum to spurious spectral components for Robust Speech Recognition. In *Proceedings of the IEEE International Conference on Audio, Speech, and Signal Processing – ICASSP (Philadelphia, USA March 2005)*, vol. 1, pp. 529 – 532, 2005 DOI: 10.1109/ICASSP.2005.1415167.

# Skúsenosti s programovaním na základnej škole

Zdeněk Hájek  
Základná škola Oslany  
Školská 56/9  
972 47 Oslany  
Slovenská republika  
spravca@zsoslany.edu.sk

Peter Vasaráb  
Základná škola Oslany  
Školská 56/9  
972 47 Oslany  
Slovenská republika  
vasarab@zsoslany.edu.sk

## ABSTRAKT

Príspevok poukazuje, že aj v mladšom školskom veku je možné začať s výučbou programovania, hoci pozoruhodné výsledky dosahujú žiaci ôsmych a deviatych ročníkov.

Príspevok opisuje aj skúsenosti s autorskou tvorbou, programovaním a prevádzkovaním interaktívnych testov a programovaných učebníc vo formáte PDF a ich spracovaním a vyhodnotením s použitím sieťových služieb. Štatisticky hodnotí rozdiel medzi klasickým papierovým testom a jeho interaktívnou formou.

Zároveň opisuje niektoré negatívne skúsenosti spojené s programovaním, pedagógmi a rodičmi, problémom prekonceptov, očakávania a roli.

## Kľúčové slová

programovanie, Scratch, Python, Javascript, interaktívny, test, programovaná, učebnica, dynamický, server, služba

## 1. ÚVOD

Nová školská reforma priniesla nový povinný predmet *informatická výchova* (ISCED1) a *informatika* (ISCED2). Do tej doby sa na našej škole na druhom stupni vyučoval voliteľný predmet *Práca s počítačmi* s časovou dotáciou 1 hodiny týždenne.

Pri príprave prvého reformného roku nového vyučovacieho predmetu *informatika* boli rozpísané jednotlivé tematické celky do jednotlivých ročníkov spolu s návrhom ich časovej dotácie, čo sa v ďalších rokoch ukázalo ako strategicky výhodné v porovnaní s *informatickou výchovou*. Programovanie bolo zámerne zaradené do deviateho ročníka. Dôvodom bol dosiahnutý stupeň osvojenia matematiky, rozvoj kognitívnych a vôľových vlastností a stupeň abstrakcie v myslení.

Tomuto kroku mala predchádzať alebo aspoň sprevádzať, príprava pedagógov, ale aj celej školy po stránke organizačnej, technickej, pedagogickej.

## 2. PROGRAMOVANIE PRE DETI

V predmete *informatická výchova* (ISCED1) je tematickým okruhom *Postupy, riešenie problémov, algoritmické myslenie* definovaný obsah výučby programovania. Použitie detského programovacieho prostredia je tu výslovne uvedené.

V predmete *informatika* (ISCED2) je obsiahnutý rovnaký tematický okruh. Z kontextu obsahového štandardu vyznieva použitie programovacieho jazyka *Pascal*.

### 2.1 Voľba softvéru

Pre voľbu vhodného softvéru sme si stanovili tieto kritériá: softvér by mal byť zadarmo, musí byť legálny, vždy dostupný, verejne známy, kedykoľvek použiteľný a vedomosti, zručnosti a návyky z

toho softvéru musia byť celoživotne rozvíjateľné a aplikovateľné aj v iných projektoch, v zmysle hesla „Učíme sa pre život“. To školský softvér nespĺňa.

Tu sa ukázalo ako závažný problém humanitné zameranie a vzdelanie pedagógov a ich slabá vôľa učiť sa nové. Napokon bol pre úvodné oboznámenie a prvé experimenty na interaktívnej tabuli zvolený projekt **Little Wizard** a neskôr **Scratch**. Druhý menovaný neskôr vďaka európskemu týždňu programovania „presiakol“ aj na druhý stupeň.

Skúsenosti z prostredia *Scratch* viedli onedlho starších žiakov k záujmu o programovanie hier vo vývojovom prostredí **Stencyl** pre prostredia Microsoft Windows, Google Android a Adobe Flash.

Väčším problémom bola voľba vhodného hlavného programovacieho jazyka pre vyučovací predmet *informatika*. Významným zdrojom informácií pri rozhodovaní boli webové portály s ponukami zamestnania.

Hneď od počiatku bolo jasné, že to nebude jazyk *Pascal*. Je to slepá ulica. Rovnakú, ak nie lepšiu službu, poskytne jazyk *C*.

Ako jazyk pre hlavnú výučbu programovania v deviatom ročníku bol napokon zvolený jazyk **Python**. Pre vytváranie vývojových diagramov bol žiakom odporúčaný program **Dia**.

### 2.2 Programovací jazyk Python

Programovací jazyk **Python** je síce interpretovaný, ale je to jazyk moderný, multiparadigmaticý, objektovo orientovaný [1]. Dokáže bežať na rôznych platformách, vrátane tabletov a mobilných telefónov. Používa sa na písanie programov, skriptov, ale aj tvorbu dynamického webu. Pythonovský kód je možné skompilovať do spustiteľného tvaru.

Veľké množstvo aplikácií tento jazyk používa ako vnútorný skriptovací jazyk. Za všetky menujme 3D modelovací, animačný a vizualizačný systém **Blender**, bitmapový editor **GIMP**, vektorový grafický editor **Inkscape.org**, kancelársky balík **OpenOffice.org** (LibreOffice.org) či DTP systém **Scribus**, čo sú aplikácie vyučované a používané aj na našej škole.

Okrem toho je štandardný jazyk *Python* podporovaný veľkým množstvom modulov (knižníc) poskytujúcich nástroje na riešenie najrôznejších problémov z oblasti vedy a techniky. Ich inštalácia je triviálna.

Je nutné povedať, že hoci je jazyk *Python* objektovo orientovaný, celú záležitosť okolo objektového programovania príliš nekomplikuje. Aj keď žiaci reálne využívajú objekty, nie je táto vedomosť pre výučbu dôležitá. Jazyk však však aktívne núti k štruktúrovanému programovaniu a postupom času aj programovaniu modulárnemu, čo je nesporne pozitívny fakt. Na ostaté programátorské paradigmy je vhodné pri výučbe rezignovať.



Jazyk je silne typový, ale kontrola typov sa vykonáva až počas behu programu. Ak je to len trochu možné, jazyk vykoná automatickú typovú konverziu. Premenné môžeme vytvárať až v okamihu, kedy ich potrebujeme a následne ich môžeme z pamäti počítača odstrániť.

Jazyk sa rýchlo a dobre učí. Programovanie je pragmatické a rýchlo vedie k stručnému a efektívnemu programovému kódu.

## 2.3 Príprava školského roka

Autori si uvedomujú, že bez dôkladnej didaktickej podpory bude akékoľvek úsilie neprimerane veľké a výsledok neistý. Na školskom webovom portáli preto vytvorili webové stránky s Tematicko výchovno-vzdelávacími plánmi predmetu, spôsobom klasifikácie, úlohami riešenými na hodine, vrátane vývojových diagramov („ťahák“) a kontaktom na vyučujúceho.

Chýbajúcu učebnicu autori nahradili vlastným učebným textom obsahujúcim okrem nevyhnutnej teórie aj množstvo príkladov, vzorových úloh a námetov na samostatné riešenie [2], [3]. Samozrejmosťou je aj verejný webový archív zdrojových kódov všetkých uvedených programov a vyriešených úloh.

Z časových dôvodov dostávajú žiaci príchode do učebne poznámky v papierovej forme, ktoré im ostávajú. Žiaci si zároveň zapisujú do zošitov postupy, osobné poznámky, postrehy, námety.

Päťstupňová klasifikácia písomných testov je realizovaná škálou 7% - 23% - 40% - 23% - 7%, čo zodpovedá symetrickému Laplace-Gaussovmu normálnemu rozdeleniu s koeficientom špicatosti  $K = -0,396$ .

Hodnotenie žiackych výkonov je realizované piatimi stupňami:

1. Programový kód je správny, program pracuje správne a dáva správne výsledky.
2. Programový kód je správny, program pracuje správne, ale neošetruje okrajové podmienky (napr. delenie nulou, nesprávne zadanie údajov), čo by neskôr viedlo k chybám.
3. Programový kód nesie zreteľné stopy správneho riešenia, program pracuje, ale nedáva správne výsledky.
4. Programový kód nesie stopy správneho riešenia, program však nepracuje.
5. Nezmyselný alebo žiadny programový kód.

## 2.4 Realizácia výučby programovania

Ako vyučovacia jednotka bola zvolená jedna dvojhodinovka týždenne, neskôr mesačne. Žiaci dochádzali na vyučovanie v max. 12 členných skupinách.

Žiaci pracujú na žiackych PC vo vývojovom prostredí **IDLE** (*Python GUI*). Vývojové diagramy tvoria v spomínanej aplikácii *Dia*. Učiteľ má k dispozícii PC a dataprojektor. Súborný sú obojsmerne presúvané a archivované školským súborovým serverom, poštovým serverom, webovým serverom a vzáčne aj službou FTP.

V objeme 26 vyučovacích hodín sa darí úspešne realizovať nasledujúce tematické celky:

V tematickom celku *Postupy, riešenie problémov a algoritmické myslenie* (4 hodiny) sa darilo definovať a vysvetliť pojem algoritmus, opísať znaky algoritmu a fázy tvorby algoritmu, ale aj definovať a opísať program a programovací jazyk. Ide o typické pamäťové učivo a žiaci ho výborne zvládajú. Naopak výraznú nechuť prejavovali pri osvojovaní si značiek vývojových

diagramov. Tvorba vzorového vývojového diagramu pre výpočet súčtu aritmetického radu „hlúpym“ postupom (cyklom), ale aj Gaussovým vzorcom bola vysoko abstraktná. Veľkým problémom bolo porozumieť významu výrazu  $x = x + 1$ . Tematický celok bol zavŕšený domácou úlohou prostredníctvom e-mailu na tému *Vývojový diagram prípravy ľubovoľného pokrmu*, ktorá sa však stretla s veľkou nevôľou dospelých a vzápätí aj žiakov.

V nasledujúcom tematickom celku *Programovací jazyk* (22 hodín) zmenou myšlienkových modelov došlo k zmenám v štruktúre triedneho kolektívu. Niektorí tradiční premianti triedy ustúpili do úzadia (zdroj rozčarovania) a bývalí outsideri začali byť naopak úspešnejší. Pasívni outsideri sa vzdávali a iba pasívne koexistovali.

Učivo *Interaktívny režim jazyka Python* bolo pre žiakov nové, zábavné a atraktívne. Výborne zvládali operátory, premenné a typy premenných, hoci sekvenčné dátové typy a ich metódy sú pomerne abstraktným učivom. Učivo *Vetvenie programu* je bezproblémové. Vyslovene zaujímavým bolo vytvorenie programu na výpočet obsahu a objemu kocky a gule. Nasleduje korešpondenčná domáca úloha *Vývojový diagram a program na výpočet obvodu a obsahu trojuholníka Heronovým vzorcom s rešpektovaním trojuholníkovej nerovnosti*. Tu si žiaci vzájomne elektronicky radili, pomáhali si a vzájomne posielali svoje riešenia. Zároveň sa ukázalo, že si množstvo žiakov doma ani nenainštalovalo potrebný softvér a necvičia s ním.

Učivo *Cyklus a riešenie úloh s opakovaním činnosti* je dostatočne precvičené a žiaci zvládajú primerané použitie príkazov cyklu v najjednoduchšej forme. Podobne nerobilo problém učivo *Moduly*. Okamžite bolo jasné, čo sú moduly a ako sa používajú. Okrem nevyhnutného modulu *math* sa žiaci zoznamujú s modulom *turtle* na vektorovú korytnačiu grafiku v štýle LOGO. Toto učivo je vysoko zaujímavé a atraktívne a žiaci sa dožadujú ďalších úloh. Výučba začína pravidelnými n-uholníkmi a n-gramami a cez drobné programy pokračuje tvorbou šachovnice (funkcie) a vrcholí tvorbou histogramov a grafov funkcií.

Modul *time* a práca s časom sú nezaujímavé. Tým končí aj pozornosť končiacich deviatakov.

V tohtoročnej časovej dotácii 13 hodín sa darí tieto ciele naplňať iba parciálne.

## 2.5 Skúsenosti s výučbou programovania

Prvým školským rokom programovania boli roky 2012/2013. Časová dotácia predmetu bola 33 hodín, čo znamená jednu dvojhodinovku dvojtyždenne. Klasifikovaný bol iba prvý polrok. Žiaci brali hodiny ako žart bez následkov. Tomu zodpovedali aj študijné výsledky – pozri tabuľku 1.

Druhým školským rokom programovania boli roky 2013/2014. Časová dotácia predmetu bola opäť 33 hodín, čo znamená jednu dvojhodinovku dvojtyždenne. Klasifikované boli obidva polroky.

Prvá korešpondenčná domáca úloha *Vývojový diagram prípravy ľubovoľného pokrmu* skončila údajnou sťažnosťou niektorých rodičov vedeniu školy na preťažovanie žiakov, náročné a nevhodnú komunikáciu e-mailom. S vyučujúcim bol vykonaný pohovor a bolo mu naliehavo odporúčané nedávať domáce úlohy (údajne sa niektorí žiaci doma učili až tri hodiny, čo je nepripustné) a neučiť programovanie, ktoré je ťažké, nepochopiteľné a nikto z žiakov sa ním nebude v budúcnosti živiť. Zároveň mu bolo odporúčané rozdeliť žiakov na tri skupiny

(budúcich gymnazistov, študentov odborných škôl a ostatných) a venovať sa iba prvej skupine. Žiaci musia pri skúšaní a testoch používať poznámky.

Po druhej korešpondenčnej domácej úlohe *Vývojový diagram a program na výpočet obvodu a obsahu trojuholníka Heronovým vzorcom s rešpektovaním trojuholníkovej nerovnosti* bol opäť vedením školy vykonaný pohovor s vyučujúcim a bolo mu naliehavo odporúčané nevyučovať *informatiku* ako takú, ale viesť predmet formou rozhovoru a besedy o téme. Väčšina žiakov však takúto organizáciu vyučovania odmietala a chcela programovať. Dokonca prejavovala hrdosť nad svojimi pokrokmi. Študijné výsledky uvedené v tabuľke 2 ukazujú, že žiaci si učivo osvojili výrazne lepšie ako žiaci v predchádzajúcom školskom roku, hoci s väčšími rozdielmi v hodnotení.

Tretím školským rokom programovania sú roky 2014/2015. Časová dotácia predmetu vo všetkých ročníkoch bola znížená na 16,5 hodiny, čo predstavuje jednu dvojhodinovku mesačne (IX.A, IX.B), alebo jednohodinovku za dva týždne (ostatné triedy). Predmet je neklasifikovaný. Zároveň bol vyučujúcemu znížený úväzok. Žiakov programovanie zaujíma a baví.

**Tabuľka 1. Študijné výsledky tried 2012/2013**

2012/2013	9.A		9.B	
charakteristika	1.polrok	2.polrok	1.polrok	2.polrok
počet žiakov	18	18	17	17
arit. priemer	2,83	---	3,29	---
var.koef. $V_k$	36,82	---	31,77	---

**Tabuľka 2. Študijné výsledky tried 2013/2014**

2013/2014	9.A		9.B	
charakteristika	1.polrok	2.polrok	1.polrok	2.polrok
počet žiakov	19	19	22	22
arit.priemer	1,42	1,42	1,59	1,24
var.koef. $V_k$	42,71	48,73	57,08	35,25

## 2.6 Krúžková činnosť

Krúžková činnosť zameraná na využívanie výpočtovej techniky bola na našej škole úmyselne potlačená. Bol povolený iba jeden takto orientovaný krúžok – *krúžok papierového modelárstva*. Vznikol spojením počítačovej grafiky, papierového modelárstva a programovania. Zameriaval sa na autorskú tvorbu papierových modelov v programoch **LibreCAD**, **Inkscape**, **GIMP** a **Scribus**.

Tento rok došlo pre veľký záujem k rozdeleniu krúžku na *Krúžok počítačovej grafiky* a *Krúžok papierového modelárstva*. Toto delenie vyhovuje lepšie aj organizačne aj didakticky. Spolupráca medzi krúžkami však ostáva naďalej veľmi úzka. Uvoľnil sa priestor aj na zaujímavejšie programovanie v prostredí Scratch – korytnačia grafika a hlavne hry.

Mimoriadne úspechy však zaznamenali žiaci školy mimo školského vyučovania. Každý rok sa našli traja až štyria žiaci, ktorí za vyučujúcim dochádzali vo svojom voľnom čase a venovali sa výrazne pokročilejšiemu programovaniu. Prvý záujemca sa venoval *jazyku C* (a *jazyku Lua*), až kým neukončil štúdium a neodišiel na gymnázium.

Nasledujúci rok sa traja noví záujemcovia venovali jazyku *Python*. Sprvoti grafike a tvorbe grafických užívateľských rozhraní, potom sa ich záujem preniesol na *HTML5* a programovanie dynamického webu v jazyku *Python* cez wrapper *CGI* v serverovom balíku **XAMPP**.

Tento rok sa ďalšia skupina štyroch spolužiakov venuje vývojovému prostrediu *Stencyl*, no jeden z nich už presúva záujem na jazyk *Python*, framework *Kivy* a dynamický web. Nezávisle od nich sa žiak ôsmeho ročníka venuje modelovaniu, animácii a vizualizácii v systéme *Blender* a pokúša sa vytvárať prvé hry, zatiaľ bez skriptovania, z dôvodu nízkej úrovne osvojenia jazyka *Python*.

## 2.7 Odporúčania

Jazyk *Python* sa javí ako ideálny pre primárnu výučbu programovania.

Programovať môže každý, ale programovanie nie je pre každého.

Učiť žiakov dospelé veci spôsobom primeraným ich veku a schopnostiam – v tom ostatne spočíva učiteľské remeslo.

Z každého stretnutia si žiak musí odniesť reálne použiteľnú vec.

Predmet rozhodne klasifikovať!

Nespoliehať sa na medzipredmetové vzťahy. Žiaci automaticky nespoja informatiku s matematikou či anglickým jazykom a budú sa tomuto spojeniu brániť. Medzipredmetové vzťahy treba explicitne vytvoriť.

Očakávajte problémy v syntaxi jazyka. Žiaci nevedia memorovať, neudržia pozornosť, nevnímajú syntaktické chyby, nevedia použiť informačné zdroje. Väčšina dokonca nedokáže bezchybne odpísať text, ktorý majú pred sebou.

Výučba algoritmizácie a programovania vyžaduje iný spôsob myslenia a úsilia. Väčšina žiakov sa tomu bude brániť. Rozčarovaní budú aj niektorí „tradiční“ premianti triedy, pretože programovanie sa nedá nabíťovať.

Očakávajte problémy zo strany rodičov a kolegov. Základná škola je veľmi konzervatívne prostredie. Navyše napr. na dedinách je previazanosť v trojuholníku **zriaďovateľ – škola – rodič** príliš silná. Zvážte, či vstúpíte do vopred prehratej vojny s rodičmi a kolegami o maximálny efekt pri minimálnom úsilí za neurčitých a meniacich sa podmienok.

## 3. PROGRAMOVANIE PRE DOSPELÝCH

Jednou z alternatív, ktorou by sa mohla uberať pedagogická veda v oblasti technologických podnikov a využitia IKT aj v prostredí základných škôl, sú interaktívne PDF formuláre. Pomocou nich možno vytvoriť moderné a vizuálne atraktívne didaktické pomôcky. Tie ponúkajú žiakom v dnešnej dynamicky sa rozvíjajúcej dobe novú, no zatiaľ stále minoritnú formu vzdelávania a overovania vedomostí.

V tomto príspevku sa pokúsia autori ich bližšie predstaviť a pokúsia sa dať odpovede na niekoľko základných otázok.

### 3.1 Čo je interaktívny PDF formulár

PDF formulár je elektronický dokument, ktorý obsahuje predtlačенý a teda ďalej nemenný text (ale aj obrázky, tabuľky...) a PDF polia, do ktorých užívateľ zadáva nami požadované data, či už ide o údaje textové alebo číselné, voľby zo zoznamu, výber odpovede a pod.

Interaktivita PDF formulárov tak spočíva v možnosti komunikácie s užívateľom – žiakom, pričom môže do istej miery nahradiť prítomnosť pedagóga. Ich výhody pre žiakov spočívajú v netradičnosti, v dôraze na vizuálnu stránku, v možnosti zapojiť viaceré zmysly, v aktivizácii širokého spektra požadovaných vlastností a interaktívnych úloh. Pedagógovi významne skracujú čas pri klasifikácii a vyhodnocovaní výsledkov a zároveň mu umožňujú dlhodobo a prehľadne archivovať ich dáta.

### 3.2 Ako vytvoriť PDF formulár

Z celej škály programov, ktoré umožňujú tvorbu interaktívnych PDF formulárov autori vybrali program *Scribus*. Ide o počítačový program na sadzbu dokumentov (tzv. Desktop publishing), ktorý patrí do open-source softvéru a je dostupný v slovenskom jazyku. Práca v Scribe nie je náročná, no má viaceré špecifiká. Z toho dôvodu autori pre budúcich užívateľov vytvorili sprievodcu funkciami programu Scribus, ako aj návod na tvorbu PDF formulárov [4], [5]. Interaktivitu PDF polí môžeme umocniť použitím skriptov programovacieho jazyka JavaScript. Vytvoríme tak nové modely správania a posunieme interaktivitu formulára na oveľa vyššiu úroveň.

I keď je na internete množstvo portálov, ktoré umožňujú pohodlne a efektívne takýto dokument vytvoriť (napr. *edupage.org*), všetky majú jednu nevýhodu – pre správne fungovanie vyžadujú pripojenie na internet. To môže mať v danom okamihu negatívne dôsledky a môže to dokonca zmarit celý edukačný proces. Zároveň do značnej miery limitujú uplatnenie vlastnej tvorivosti a prispôbenie konkrétnym potrebám. Autormi prezentované formuláre tieto nedostatky významne eliminujú.

### 3.3 Využitie PDF formulárov v edukácii

Interaktívne PDF formuláre majú širokospektrálne využitie pričom môžu pedagógovi slúžiť ako didaktická pomôcka, organizačná forma či vyučovacia metóda. Rovnako ich je možné zaradiť do všetkých etáp vyučovacieho procesu od motivačnej až po diagnostickú. Autori zamerali ich využitie na oblasť e-learningu. Vytvorili formuláre v podobe elektronických testov z dejepisu (téma *Slovenská republika v rokoch 1939 – 45, Vek rozumu*, určovanie storočí [6], *Zámorské objavy* [7]), z geografie (*Prírodné pomery Slovenska* [8], členenie Európy) a z matematiky (*Cvičebnica sčítania* [9], *Cvičebnica odčítania* [10], *Premena jednotiek* [11]). Tieto testy automaticky rozpoznávali správne odpovede, vypočítali percentuálnu úspešnosť a na základe zadanej stupnice vygenerovali známku, farebne vyznačovali správne a nesprávne odpovede, nerozlišovali (nebrali za chybu) v texte veľké a malé písmená, náhodne generovali zadania príkladov, odosielali žiakove odpovede na požadovanú mailovú adresu pre ďalšie spracovanie a archiváciu školským serverom. Testy boli jednostránkové aj viacstránkové, s jednotným zadáním pre všetkých žiakov alebo s jedinečným, nikdy sa neopakujúcim zadáním (určovanie storočí, odčítanie...).

PDF formulár môže byť umiestnený aj na webovej stránke školy, kde poslúži pri precvičovaní a fixácii učiva aj počas domácej prípravy alebo pre absentujúcich žiakov. Aj tu má vyhodnotenie testu okamžité k dispozícii, tentoraz pre potreby rodiča, prípadne online aj učiteľa. Formuláre boli výbornou pomôckou pri komisionálnom skúšaní žiaka z geografie, ktorý si plnil povinnú školskú dochádzku v zahraničí. Žiak vyplnil zadanie úlohy v Španielsku, odoslal dáta na učiteľov mail na školskom serveri a ten mal okamžite k dispozícii nielen odpovede, ale aj výsledok

celej skúšky. Odpadla tým zdĺhavá a finančne náročná korešpondencia.

Niektoré z formulárov boli v školskom roku 2012/2013 poskytnuté kolegom z *matematiky* a *fyziky*. Ani jeden ich však dodnes vo vyučovaní nepoužil, čím autorom neumožnil získať ďalšiu spätnú väzbu o ich využití, prípadne podať nové námety na ich vylepšenie alebo doplnenie.

Druhou možnosťou využitia interaktívnych PDF formulárov je **programovaná učebnica**. Formulár tak podľa pravidiel programovaného vyučovania kombinuje prezentáciu nového učiva a okamžité overovanie jeho zvládnutia. Pre potreby výučby autori týmto spôsobom spracovali učivo z geografie (*Vodstvo Slovenska* [12]) a učivo z dejepisu (*Ríša Inkov* [13]), ktoré odučili pomocou interaktívnej tabule. Bol to jeden z mála prípadov, kedy táto didaktická pomôcka neslúžila len ako drahé premietacie plátno. Aj v prípade programovaných učebníc sa po umiestnení na internet ponúkajú rovnaké možnosti využitia, ako pri elektronických testoch. Z nadobudnutej skúsenosti vyplýva, že učivá preberané formou programovanej učebnice boli najlepšie zvládnuté z celého tematického celku a patrili u žiakov k najčastejšie vybraným pri ústnom preverovaní vedomostí.

### 3.4 Porovnanie klasickej a e-výučby

Autori otestovali či je elektronický test (PDF formulár) pre žiakov výhodnejší a dosiahnu pri ňom iné výsledky ako pri testovaní klasickým. Konkrétne išlo o učivo *Prírodné pomery Slovenska* [8].

Pre testovanie rovnosti rozptylov hodnôt  $\sigma_x^2 = \sigma_y^2$  použili F-test. Pre porovnanie rovnosti stredných hodnôt úspešnosti odpovedí  $\mu_x = \mu_y$  použili t-test. Predpokladali, že pri elektronickom teste budú výsledky výrazne lepšie. Pri porovnaní dosiahnutej úspešnosti však dospeli k záveru, že forma testu nemala významnejší vplyv na celkové výsledky, pričom v tomto prípade boli závery totožné s výsledkami podobného prieskumu z katedry matematiky FEM SPU v Nitre [14]. Žiadna „veľká revolúcia“ sa teda nekonala. Dievčatá, chlapci, ale aj celá trieda odpovedali takmer rovnako a rovnako sa aj mýlili (na hladine významnosti  $\alpha = 0,001$ , t.j. s 99,9 % pravdepodobnosťou).

Niektoré popisné charakteristiky výsledkov testov sú uvedené v tabuľke 3:

**Tabuľka 3. Niektoré štatistické charakteristiky testov.**

trieda IX.B	klasická písomná práca			interaktívny PDF formulár		
	dievčatá	chlapci	trieda	dievčatá	chlapci	trieda
priem.úspešnosť v [%]	87,82	61,06	72,5	85,58	72,84	78,30
var. koef. $V_k$ v [%]	14,57	48,29	37,00	12,69	28,24	22,70
počet žiakov	6	8	14	6	8	14

Z tabuľky 3 vyplýva, že v klasickom teste dievčatá učivo úspešne zvládli (87,82 %) a ich odpovede boli veľmi vyrovnané ( $V_k = 14,57$  %). V elektronickej forme bola síce úspešnosť mierne nižšia, napriek tomu dievčatá učivo úspešne zvládli (85,58 %) a ich odpovede boli ešte vyrovnanejšie ( $V_k = 12,69$  %).

Z tabuľky 3 zároveň vyplýva, že v klasickom teste chlapci učivo nezvládli (61,06 %) a ich odpovede boli veľmi nevyrovnané ( $V_k = 48,29$  %). V elektronickej forme bola úspešnosť vyššia (72,84 %), chlapci už učivo takmer zvládli a ich odpovede boli viac vyrovnané ( $V_k = 28,24$  %).

Z tabuľky 3 vyplýva, že trieda v klasickom teste učivo takmer zvládla (72,53 %) a odpovede žiakov boli pomerne vyrovnané ( $V_k = 37,00$  %). V elektronickej forme bola síce úspešnosť ešte vyššia (78,30 %), ale nedosiahla požadovaných 80 % úspešnosti. Odpovede žiakov boli vyrovnanejšie ( $V_k = 22,70$  %) ako pri klasickom teste.

Hoci z uvedených údajov vyplýva, že forma testu nemá štatisticky významný vplyv na celkovú úspešnosť, z hodnôt variačných koeficientov vyplýva, že elektronický test umožňuje lepšiu koncentráciu žiakov a znižuje variabilitu odpovedí (napr. žiaci odpovedajú viacej podobne alebo sa menej mylia). To zrejme vyplýva z aktivizácie viacerých zmyslov a atraktívnosti, ktorá jednoznačne vyplýva aj z výsledkov dotazníka.

### 3.5 Ako hodnotia PDF formuláre žiaci

Hneď na úvod treba povedať, že žiaci základnej školy spočiatku pristupovali k interaktívnym PDF formulárom, a teda k e-learningu s veľkou nedôverou, až obavami. Bolo to pre nich niečo nové, nepoznané, a tým pádom u mnohých vopred zatracované (prvého žiaka siedmej triedy bolo treba k interaktívnej tabuli s programovanou učebnicou doslova donútiť). Našťastie to platilo len do momentu, kedy s nimi začali aktívne pracovať. Bezprostredne po absolvovaní kontrolnej práce s PDF formulárom na tému *Slovenská republika v rokoch 1939 – 45* autori pre 37 žiakov 9. ročníka vypracovali krátky dotazník. Po spracovaní ich odpovedí výsledky prieskumu príjemne prekvapili:

- 58,82% žiakov považuje za potrebné dodržiavať gramatiku,
- 70,59% žiakov považuje za dôležité farebné členenie testu,
- 73,53% žiakov uprednostní elektronickú formu testovania pred klasickou,
- 79,41% žiakov uprednostní písanie pomocou klávesnice,
- 85,29% žiakov uvíta v teste obrázky a mapy,
- 88,23% žiakov uvíta okamžité hodnotenie testu,
- 91,18% žiakov považuje za dôležité označenie správnych odpovedí,
- 97,06% žiakov preferuje otázky s možnosťou výberu odpovede.

Tieto čísla hovoria jasnou rečou: zvolená forma testovania, ako aj spracovanie testu boli žiakmi prijaté nanajvýš pozitívne. V opisovanom prípade takmer 74 % žiakov uprednostní interaktívny PDF formulár pred klasickým testom. Výsledky sú pre autorov o to príjemnejšie, že sú v ostrom kontraste s výsledkami podobného prieskumu z katedry matematiky FEM SPU v Nitre. Tu by pre zmenu až 75 % študentov uprednostnilo klasickú formu testovania pred elektronickou [14].

## 4. POSTOJE PEDAGÓGOV

Aby autori rozšírili rady pedagógov schopných vytvoriť podobné formuláre a zároveň, aby ukázali náročnej a do značnej miery kritickej spoločnosti, že aj učiteľ na základnej škole vie vytvoriť kvalitné, moderné a konkurencie schopné didaktické pomôcky, uskutočnili v priestoroch základnej školy kurz. Na tomto kurze formou workshopu prezentovali pedagógom z okresu Prievidza prácu a možnosti využitia programu *Scribus*. Napriek vysokej účasti a pozitívnym ohlasom prítomných kolegov nepokračoval po skončení kurzu v začatej práci ani jeden. Sľubne sa rozvíjajúci potenciál tak zostal trestuhodne nevyužitý. Nepomohol ani fakt,

že si všetci zo stretnutia odnášali pracovný list s tvorivými úlohami, ku ktorým bol priložený aj vytlačený manuál o práci v programe.

A nebola to jediná negatívna skúsenosť autorov pri pokuse priblížiť danú problematiku širokej pedagogickej obci. Prácu s jazykom JavaScript sa pokúsili prezentovať prostredníctvom akcie organizovanej pod záštitou MŠ SR s názvom **Európsky týždeň programovania**. Akcia sa konala v dňoch 24.11. až 28.11.2014 v priestoroch základnej školy a zúčastniť sa jej mohli žiaci, pedagógovia aj verejnosť. Program bol nasledujúci:

- pondelok – beseda s profesionálnym programátorom
- utorok – interaktívne PDF testy
- streda – programovanie v detskom prostredí Scratch
- štvrtok – programované učebnice
- piatok – programovanie v jazyku Python

Na rozdiel od iných, napr. športových či kultúrnych akcií, vedenie školy umožnilo poriadkať túto aktivitu iba v poobedňajších hodinách. Úlohou propagácie pripravovaného podujatia boli poverení triedni učitelia. Výsledkom bola nulová účasť. Po rýchlej osobnej pozvánke jednotlivých tried sa situácia významne zmenila, aj keď dôvody účasti žiakov boli rôzne.

O možnosť zoznámiť sa s JavaScriptom, urobiť prvé kroky v programovaní a vytvoriť svoj prvý interaktívny PDF formulár, neprejavil záujem ani jeden (!) pedagóg.

Prezentácie, spojené s praktickými ukážkami programovania, sa síce zúčastnili dvaja učitelia, no dôvodom nebol záujem o problematiku, ale dozor nad prítomnými žiakmi v rámci poobedňajšieho vyučovania. K počítaču si, na rozdiel od všetkých žiakov 9. ročníka, ani nesadli. Po skončení vyučovacej hodiny okamžite odišli domov, a svojím odchodom a nezaujmom bohužiaľ strhli aj väčšinu žiakov. Vedenie školy neprejavilo záujem o aktivitu žiadnym spôsobom. Úspešne zavŕšená akcia nebola ohodnotená, či dokonca odmenená žiadnym spôsobom.

Príčiny takéhoto prístupu si možno len domyslieť. Na druhej strane je potešujúci relatívne vysoký počet žiakov, ktorí prišli slobodne a dobrovoľne, ale aj relatívne veľký počet tých, ktorí už pri aktívnom programovaní ostali.

## 5. ZÁVER

Programovanie na základnú školu rozhodne patrí. Programovať môžu aj žiaci najnižších ročníkov. Netreba túto oblasť ľudskej činnosti nijako glorifikovať, zatracovať alebo inak deformovať jej význam a úlohy. Mimoriadne dôležitá je v tomto ohľade osobnosť učiteľa. Nemusí to byť školený profesionál, pedagóg, ale predovšetkým vnímavý participujúci partner. Je nesmierne dôležité, aby neustále správne vnímal a interpretoval, čo žiak chce a očakáva, aby správne zvolil, ponúkol a aktualizoval vzdelávací obsah, formu, metodiku a osobný prístup. Vtedy si programovanie s určitosťou istý počet zájemcov nájde. Vždy však pôjde o oblasť minoritného záujmu. Preto navrhujeme nezavádzať programovanie do hlavnej výučby, ale povoliť krúžky aj s relatívne malým počtom žiakov.

Zároveň odporúčame veľmi dôkladne zvážiť použitie školského softvéru v tejto oblasti informatiky.

Niekoľkoročné osobné skúsenosti v oblasti tvorby a využitia interaktívnych PDF formulárov autorov jednoznačne presvedčili, že tento spôsob prezentovania a diagnostiky obsahu vzdelávania

si úspešne hľadá svoje miesto aj v prostredí základných škôl. Dôkazom sú pozitívne ohlasy žiakov a do istej miery aj výsledky ich práce. Formuláre sa javia ako jedna z možných alternatív efektívneho využitia súčasnej vysokej popularity informačných technológií u žiakov. Dôležité je ich správne a vyvážené použitie.

Jediným problémom tak zatiaľ z pohľadu autorov zostáva nedostatočný počet tvorcov z radov pedagógov, ktorí napriek súčasným možnostiam využívajú softvér a programy vytvorené pred desiatkami rokov. Dôkazom je fakt, že drvivá väčšina dokumentov na portáli *www.zborovna.sk* sú textové dokumenty (*Microsoft Word*) alebo prezentácie (*Microsoft PowerPoint*). Tu neobstoja dôvody nedostupnosti techniky, či softvéru, ani dôvody nedostatku financií, času či priestoru.

Týmto príspevkom chceli autori ukázať a dokázať, že programovanie a tvorba moderných a vizuálne atraktívnych didaktických pomôcok nemusí byť výhradne záležitosťou programátorov, respektíve ľudí vzdelaných v oblasti počítačových technológií.

Preto za najzávažnejší problém výučby a využívania IKT považujú postoj rozsiahlej a dominantnej skupiny pedagógov základných škôl, najmä ich neochotu a často aj neschopnosť vymaniť sa z pevnej a pokojnej ulity svojich predmetových aprobácií, pohodlia a bezpečia školských múrov a dynamicky reagovať na meniaci sa svet a jeho potreby.

Jedným dychom voláme po spoločenskom uznaní, spravodlivom a dôstojnom ohodnotení, no na druhej strane, nie všetci sme ochotní obetovať čas ani energiu, ktoré tak nekompromisne požadujeme od svojich žiakov.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] SUMMERFIELD, M.: Python 3. Výukový kurz. 1. vyd. Brno : Computer Press, a.s., 2010. 584 s. ISBN 978 –80–251–2737–7. str. 13.
- [2] HÁJEK, Z.: Jemný úvod do programovania v jazyku Python pre žiakov ZŠ Oslany. [online]. [cit 2015-02-06] Dostupné na: <<http://www.zsoslany.edu.sk/psp/kniha.pdf>>.
- [3] HÁJEK, Z.: Programovanie v jazyku Python. [online]. [cit. 2015-02-06]. Dostupné na: <<http://www.zsoslany.edu.sk/psp/python.html>>.
- [4] VASARÁB, P.: Scribus 1.3.3.14. Desktop Publishing. Sprievodca funkciami programu. [online]. [cit. 2015-02-06] Dostupné na: <[http://www.zsoslany.edu.sk/scribus/manual\\_scribus\\_dtp.pdf](http://www.zsoslany.edu.sk/scribus/manual_scribus_dtp.pdf)>.
- [5] VASARÁB, P.: Scribus a tvorba interaktívnych PDF formulárov. [online]. [cit. 2015-02-06] Dostupné na: <[http://www.zsoslany.edu.sk/scribus/manual\\_scribus\\_form.pdf](http://www.zsoslany.edu.sk/scribus/manual_scribus_form.pdf)>.
- [6] [online]. [cit. 2015-02-06] Dostupné na: <<http://www.zsoslany.edu.sk/scribus/storocia.pdf>>
- [7] [online]. [cit. 2015-02-06] Dostupné na: <[http://www.zsoslany.edu.sk/scribus/zamorske\\_objavy.pdf](http://www.zsoslany.edu.sk/scribus/zamorske_objavy.pdf)>
- [8] [online]. [cit. 2015-02-06] Dostupné na: <[http://www.zsoslany.edu.sk/scribus/prirodne\\_pomery\\_slovenska.pdf](http://www.zsoslany.edu.sk/scribus/prirodne_pomery_slovenska.pdf)>
- [9] [online]. [cit. 2015-02-06] Dostupné na: <<http://www.zsoslany.edu.sk/scribus/cvicebnica.pdf>>.

- [10] [online]. [cit. 2015-02-06] Dostupné na: <<http://www.zsoslany.edu.sk/scribus/odcitanie.pdf>>.
- [11] [online]. [cit. 2015-02-06] Dostupné na: <[http://www.zsoslany.edu.sk/scribus/premena\\_jednotiek.pdf](http://www.zsoslany.edu.sk/scribus/premena_jednotiek.pdf)>.
- [12] [online]. [cit. 2015-02-06] Dostupné na: <<http://www.zsoslany.edu.sk/scribus/vodstvo.pdf>>.
- [13] [online]. [cit. 2015-02-06] Dostupné na: <[http://www.zsoslany.edu.sk/dejepis/7\\_rocnik/risa\\_inkov.pdf](http://www.zsoslany.edu.sk/dejepis/7_rocnik/risa_inkov.pdf)>.
- [14] PECHOČIAK, T.: E-testovanie verzus klasické overovanie vedomostí. In : Slovenský učiteľ. str. 4 [online]. [cit. 2015-02-06] Dostupné na: <<http://technologiadvzdelavania.ukf.sk/index.php/tv/article/download/398/414>>.

# Prvé skúsenosti s realizáciou súťaže iBobor pre nevidiacich žiakov druhého stupňa ZŠ

Ľudmila Jašková

KZVI FMFI UK

Mlynská dolina

842 48 Bratislava

Slovenská republika

jaskova@fmph.uniba.sk

Natália Kováčová

KZVI FMFI UK

Mlynská dolina

842 48 Bratislava

Slovenská republika

kovacova.naty@gmail.com

## ABSTRAKT

Súťaž iBobor už niekoľko rokov zohráva významnú úlohu v určovaní trendov pre oblasť vyučovania informatiky. V našom článku navrhujeme prispôbenia, ktoré by umožnili zapojiť sa do nej nevidiacim žiakom. V nadväznosti na to navrhujeme pravidlá pre špeciálnu kategóriu nevidiacich žiakov 2. stupňa ZŠ a uvádzame niektoré úlohy, ktoré sme vytvorili a overili počas reálneho behu súťaže v rokoch 2013 a 2014. Venujeme sa aj analýze úspešnosti nevidiacich žiakov v tejto súťaži a rozoberáme faktory, ktoré ju ovplyvňujú.

## ABSTRACT

Bebras contest plays an important role in determining trends in the area of informatics education for several years. In this article we propose special adaptation that would allow blind pupils to participate in this contest. Following this, we propose the rules for a special category of blind pupils of lower secondary education. Besides we present several tasks that we developed and verified during the real contest running in 2013 and 2014. We analyse results of blind pupils and discuss the factors that affect it.

## Kľúčové slová

iBobor, nevidiaci, prístupnosť, prispôbenia, analýza úspešnosti

## Keywords

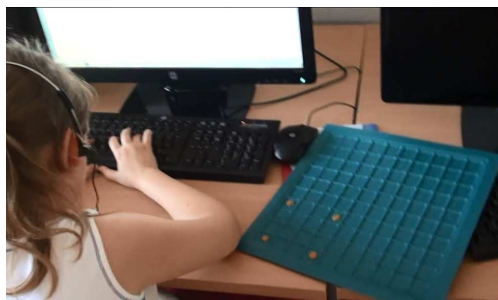
Bebras contest, blind pupils, accessibility, adaptations for the blind, analyse of results

## 1. ÚVOD

*“Súťaž Informatický bobor, ďalej iBobor, je jedna z mála súťaží, ktoré si kladú za cieľ osloviť veľkú skupinu žiakov, rozvíjať ich kognitívne schopnosti v oblasti využívania digitálnych technológií a informatiky v každodennom živote, inšpirovať ich k záujmu o témy informatiky, učiť ich argumentovať a zdôvodňovať svoje riešenia ...”* [1] Hlavným cieľom súťaže je podporiť záujem o IKT u všetkých žiakov. Súťaž má ambície iniciovať v deťoch využívanie IKT, posmeliť ich v intenzívnejšom a kreatívnejšom používaní moderných technológií pri učení sa [2] a [3].

V súčasnosti nie sú vo všeobecnosti úlohy prístupné pre žiakov s rôznymi typmi zrakového postihnutia. V mnohých úlohách je pre pochopenie zadania dôležitý obrázok, ktorý nemá vytvorenú textovú alternatívu. Problém môžu mať aj žiaci s poruchami farebného videnia, pretože častokrát sú dôležité objekty na obrázkoch odlíšené farbami. Interaktívne úlohy neumožňujú ovládanie pomocou klávesnice, ale iba myšou. Občas sa v úlohách vyskytujú tabuľky, ktoré sú pre nevidiacich neprehľadné. Slabozrakí žiaci používajú zväčšovací softvér, ktorí nemajú

problémy s vnímaním farieb však dokážu úlohy riešiť bez problémov. Na základnej škole pre nevidiacich a slabozrakých, na ktorej sme pôsobili v priebehu ostatných rokov, sa súťaže zúčastňovali vidiaci a slabozrakí žiaci. Nevidiaci žiaci a žiaci so zvyškami zraku sa však súťaže nemohli zúčastniť, hoci na bežných hodinách informatiky dosahovali veľmi dobré výsledky. Našou snahou bolo hľadať možnosti, ako by sa do súťaže mohli zapojiť aj títo žiaci. Navrhli sme prispôbenia pravidiel súťaže, ktoré by im to umožnili. Uvádzame ich v nasledujúcej kapitole. V kapitole 3 prezentujeme niektoré úlohy vhodné pre nevidiacich žiakov. Ich vhodnosť sme overili počas reálneho behu súťaže. Priebeh a výsledky overovania uvádzame v kapitole 4.



Obrázok 1. Nevidiaca žiačka rieši na hodine informatiky algoritmickú úlohu

## 2. PRISPÔBENIA PRE NEVIDIACICH ŽIAKOV

Pokúsili sme sa navrhnúť prispôbenia pravidiel súťaže tak, aby sa do nej mohli nevidiaci zapojiť ako plnoprávni súťažiaci. Bolo však pre nich potrebné vytvoriť samostatnú kategóriu, keďže sa jednalo o pomerne rozsiahle zmeny.

Vytvorili sme kategóriu NEVIDIACI pre nevidiacich žiakov 5. až 9. ročníka ZŠ. Spojiť žiakov štyroch ročníkov do jednej kategórie sme si dovolili z toho dôvodu, že žiaci piateho a šiesteho ročníka mali na prvom stupni informatickú výchovu, vďaka čomu boli ich informatické zručnosti porovnateľné so zručnosťami žiakov vyšších ročníkov.

Bežní žiaci v kategóriách Benjamínov a Kadetov, ktoré vekovo zodpovedajú našim nevidiacim žiakom, majú riešiť 15 úloh (po 5 úloh v každej úrovni – ľahká, stredná, ťažká). Na vyriešenie jednej úlohy majú v priemere približne 3 minúty. Pri testovaní nevidiacich žiakov sa zvykne predĺžiť čas podľa individuálnych



potrieb jednotlivcov. Podľa odporúčaní organizácie RNIB<sup>1</sup>, treba pri testovaní nevidiacich predĺžiť čas o 25 až 100 %. Podľa odporúčaní C. B. Allmana [4] pre testovanie osôb so zrakovým postihnutím sa má čas predĺžiť o 50 %. Keďže bežná vyučovacia hodina trvá 45 minút, nechceli sme tento čas nadstavovať, aby žiaci zbytočne nevymeškávali z ďalších vyučovacích hodín. Rozhodli sme sa počas tohto časového limitu zaradiť pre nich iba 9 úloh (3 ľahké, 3 stredne ťažké, 3 ťažké). V priemere na jednu úlohu pripadalo 5 minút, čo je o 60% viac času v porovnaní s intaktnými žiakmi.

Úlohy neboli prezentované online na webe, ako to majú intaktní žiaci. Použili sme formát textového dokumentu v počítači. S textovým dokumentom dokázali pracovať všetci nevidiaci žiaci, s webovým prehliadačom mali ešte niektorí z nich problémy. Na dodržanie časového limitu mal dohliadať prítomný učiteľ a nie počítač, ako je to v prípade riešenia online. Pri tvorbe úloh sme sa snažili dodržiavať pravidlá prístupnosti určené výnosom o štandardoch pre informačné systémy verejnej správy<sup>2</sup>.

- Všetky potrebné informácie boli v textovom formáte.
- V zadaniach sa nevyskytovali obrázky.
- Neodvolávali sme sa na farby.
- Informácie v tabuľkách dávali zmysel pri čítaní po riadkoch a v rámci riadkov pri čítaní zľava doprava.

Okrem toho sme prihliadali na fakt, že nevidiaci nemajú náhľad na celú obrazovku a mnohé dôležité informácie si musia uchovávať v pamäti. Krátkodobá pamäť však dokáže uchovať len asi 7 položiek a to po dobu 15 až 30 sekúnd [5]. Bolo preto potrebné skrátiť v úlohách niektoré postupnosti inštrukcií alebo prvkov.

### 3. ÚLOHY

Prvýkrát sme súťaž pre nevidiacich realizovali v novembri 2013. Najprv sme realizovali takzvané **pilotné kolo**, ktoré sa konalo približne dva týždne pred **súťažným kolom**. Cieľom pilotného kola bolo overiť vhodnosť úloh a pripraviť žiakov na to, aby vedeli, ako bude súťažné kolo prebiehať. Pilotné kolo prebiehalo počas bežných hodín informatiky. Na nasledujúcej hodine informatiky ich učiteľ informoval o ich úspešnosti a diskutovali v skupine o riešení jednotlivých úloh. Takto pripravení sa žiaci, ktorí mali záujem, zúčastnili súťažného kola, ktoré sa konalo počas reálneho behu súťaže v novembri 2013.

O rok neskôr, v novembri 2014, sa súťaž pre nevidiacich konala druhýkrát. Počas uvedených troch behov postupne vznikli 3 sady úloh, ktoré popíšeme bližšie.

#### 3.1 Pilotná sada úloh

Úlohy pre pilotné kolo sme vyberali z databázy úloh, ktoré boli použité v predchádzajúcich ročníkoch súťaže pre žiakov kategórie Benjamíni. Naším zámerom bolo, aby sa v súhrne úlohy týkali každej tematickej oblasti. Najviac úloh sa týkalo oblasti Informácie okolo nás (2,5 úlohy) a Postupy, riešenie problémov, algoritmické myslenie (2,5 úlohy). Oblasť Princípy fungovania digitálnych technológií sa týkala dve úlohy. Jedna úloha sa týkala zvyšných dvoch oblastí – Informačná spoločnosť a Komunikácia prostredníctvom IKT.

<sup>1</sup>[http://www.rnib.org.uk/sites/default/files/Overview\\_of\\_exam\\_access\\_arrangements\\_May\\_2014.doc](http://www.rnib.org.uk/sites/default/files/Overview_of_exam_access_arrangements_May_2014.doc)

<sup>2</sup>[http://www.informatizacia.sk/ext\\_dok-vynos\\_2014-55\\_standardy\\_isvs\\_s\\_prilohami/17060c](http://www.informatizacia.sk/ext_dok-vynos_2014-55_standardy_isvs_s_prilohami/17060c)

Tabuľka 1 znázorňuje porovnanie zaradenia úloh z hľadiska obtiažnosti pre intaktných a pre nevidiacich žiakov. Všimnime si rozdielne zaradenie pri úlohách 4, 5, 6 a 9. Inak boli zaradené v pilotnom kole pre nevidiacich žiakov a inak v predchádzajúcich ročníkoch súťaže pre intaktných žiakov.

**Tabuľka 1. Prehľad obtiažnosti úloh pilotného kola**

Číslo	Názov	Téma	Obtiažnosť		Modifikácia
			nevidiaci	intaktní	
1	E-mailové šialenstvo	SPOL, KOM	Ľahká	Ľahká	nie
2	Čo chýba?	PC	Ľahká	Ľahká	nie
3	Čo nehrá?	PC	Ľahká	Ľahká	nie
4	Menovka	INF	Stredná	Ľahká	áno
5	Algoritmus	ALG	Stredná	Ľahká	nie
6	Robot	ALG	Stredná	Ťažká	áno
7	Člnkovanie	INF	Ťažká	Ťažká	nie
8	Knihy	ALG	Ťažká	Ťažká	nie
9	Odzadu	ALG, INF	Ťažká	Ľahká	nie

Úloha **4 Menovka** bola pôvodne zaradená ako ľahká a v zadaní sa používali obrázky zvierat. My sme obrázky nahradili textami – názvami zvierat a úlohu sme zaradili ako stredne ťažkú.

Aj úloha **5 Algoritmus** bola pôvodne zaradená ako ľahká. Nevidiaci žiaci síce už riešili jednoduché algoritmické úlohy, ale slovo algoritmus a jeho význam nepoznali. Preto sme úlohu zaradili ako stredne ťažkú.

Úloha **6 Robot** bola zaradená ako ťažká. V pôvodnej úlohe mal však robot nakresliť komplikovanejší obrázok (tabuľka 2). V úlohe pre nevidiacich mal robot nakresliť štvorec. Keďže sme úlohu zjednodušili, zaradili sme ju medzi stredne ťažké.

**Tabuľka 2. Zadanie úlohy 6 pre intaktných žiakov**

<p>Kresliaci robot pozná príkazy:</p> <p><b>dopredu počet</b> – prejde počet krokov smerom, ktorým je natočený</p> <p><b>vpravo uhol</b> – otočí sa na mieste o daný uhol vpravo,</p> <p><b>vľavo uhol</b> – otočí sa na mieste o daný uhol vľavo,</p> <p><b>opakuj číslo [príkaz príkaz ...]</b> – robot zopakuje príkazy v zátvorkách [príkaz príkaz ...] toľkokrát, koľko udáva číslo</p> <p><b>Ktorým z týchto príkazov nakreslí robot obrázok?</b></p> <p><b>A</b> opakuj 4 [dopredu 40 vľavo 90 dopredu 10 vpravo 90 dopredu 10 vľavo 90]</p> <p><b>B</b> opakuj 4 [dopredu 40 vpravo 90 dopredu 10 vpravo 90 dopredu 10 vpravo 90]</p> <p><b>C</b> opakuj 4 [dopredu 40 vľavo 90 dopredu 10 vľavo 90]</p> <p><b>D</b> opakuj 4 [dopredu 40 vpravo 90 dopredu 10 vpravo 90 dopredu 10]</p>	
--	---

Zadanie pre nevidiacich bolo v úvodnej časti zhodné s týmto zadaním, len otázka bola iná.

**Tabuľka 3. Otázka pre nevidiacich žiakov (úloha 6)**

<p><b>Ktorým z týchto príkazov nakreslí robot štvorec?</b></p> <p><b>A</b> opakuj 4 [dopredu 40 vľavo 90 dopredu 20 vľavo 90]</p> <p><b>B</b> opakuj 4 [dopredu 40 vľavo 90 dopredu 20 vpravo 90]</p> <p><b>C</b> opakuj 4 [dopredu 40 vľavo 90]</p> <p><b>D</b> opakuj 4 [dopredu 40 vľavo 90 dopredu 40 vpravo 90]</p>
--

Úloha **9 Odzadu** bola pôvodne zaradená ako ľahká. My sme ju zaradili ako ťažkú, pretože naši žiaci riešili úlohy spamäti.

Nepoužívali pero a papier. Pre žiakov bez vizuálnej predstavivosti nebolo ľahké nájsť riešenie.

#### Tabuľka 4. Úloha 9 pre nevidiacich žiakov

Kamaráti písali v textovom editore slovo Perinbaba a pritom sa hrali. Jano stlačil po napísaní každého písmena kláves Enter, Anka kláves Backspace (ktorý maže znak pred kurzorom), Marienka šípku vľavo a Peter kláves CapsLock. **Keď potom kontrolovali, čo kto napísal, zistili, že jeden z nich má slovo Perinbaba napísané odzadu. Kto to bol?**

- A. Jano
- B. Peter
- C. Marienka
- D. Anka

### 3.2 Súťažné úlohy v novembri 2013

Pri príprave súťažných úloh sme sa opäť snažili, aby boli v úlohách zastúpené všetky tematické oblasti. Podobne ako v pilotnom kole, najviac úloh bolo tematicky zameraných na oblasti Informácie okolo nás (3 úlohy) a Postupy, riešenie problémov, algoritmické myslenie (3 úlohy). Zvyšným trom oblastiam bola venovaná vždy len jedna úloha.

Snažili sme sa čo najviac úloh prevziať z aktuálnej kategórie pre Benjamínov a iba ich upraviť. Tabuľka 5 znázorňuje porovnanie zaradenia úloh z hľadiska obtiažnosti pre intaktných a pre nevidiacich žiakov.

Tabuľka 5. Prehľad obtiažností úloh súťažného kola v r. 2013

Číslo	Názov	Téma	Obtiažnosť		Modifikácia
			nevidiaci	intaktní	
1	Bobor book	SPOL	ľahká	ľahká	nie
2	E-mailová adresa	KOM	ľahká	ľahká	nie
3	Čo nemôže byť použitý?	PC	ľahká	ľahká	nie
4	Obľúbenosť programov	INF	stredná	ľahká	áno
5	Rodinné vzťahy	INF	stredná	ľahká	áno
6	Zmrzlina	INF, ALG	stredná	stredná	áno
7	Robot	ALG	ťažká	ťažká	áno
8	Výťah	ALG	ťažká	stredná	nie
9	Text odzadu	INF, ALG	ťažká	ľahká	áno

Úlohu **1 Bobor book** sme prevzali bez zmeny a zaradili ju medzi ľahké, ako bola aj pôvodne.

Úlohy **2 E-mailová adresa** a **3 Čo nemôže byť použitý?** sme našli v národnej databáze úloh, ktoré ale zatiaľ neboli v súťaži použité. Ak by boli, tak ich náročnosť by bola posudzovaná ako ľahká, čo sme aj zachovali.

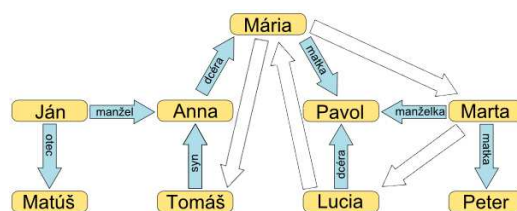
Úloha **4 Obľúbenosť programov** mala podobný osud, ale zadanie sme mierne pozmenili. V pôvodnej úlohe boli informácie o obľúbenosti programov reprezentované stĺpcovým grafom. My sme použili iba slovný popis. Úlohu sme zaradili medzi stredne ťažké.

Úloha **5 Rodinné vzťahy** bola pre intaktných žiakov zaradená ako ľahká, my sme ju zaradili medzi stredne ťažké. Bolo to z toho dôvodu, že v pôvodnej úlohe boli rodinné vzťahy znázornené grafom a my sme ich vyjadrili slovne. Intaktní žiaci mali len interaktívne doplniť do príslušných šípok v grafe kto je komu starý rodič. Nevidiaci si museli najprv prečítať množstvo textu o rozvetvenej rodine a potom určiť, ktoré zo štyroch tvrdení o starých rodičoch je pravdivé. Zadanie úlohy pre intaktných žiakov bolo nasledovné.

Tabuľka 6. Zadanie úlohy 5 pre intaktných žiakov

Na obrázku sú zakreslení členovia našej rodiny a niektoré vzťahy medzi nimi.

Kliknutím označ šípku, ktorá znázorňuje vzťah **stará mama**.



Tabuľka 7. Zadanie úlohy 5 pre nevidiacich žiakov

Bobor Dávid má štyri deti. Volajú sa Ján, Anna, Pavol a Mária. Všetci už majú svoje deti. Ján má syna Jakuba, Anna má syna Tomáša, Pavol má dcéru Luciu a Mária má syna Petra.

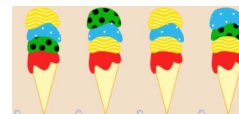
**Ktoré z nasledujúcich tvrdení je pravdivé?**

- A Dávid je Máriin dedko
- B Lucia je Dávidova babka
- C Dávid je Tomášov dedko
- D Mária je Luciina babka

V zadaní úlohy **6 Zmrzlina** pre intaktných žiakov boli použité obrázky. V zadaní pre nevidiacich žiakov sme ich nahradili textom. Úlohu sme zjednodušili tak, že zmrzlinový stroj vytvára zmrzliny s tromi kopčekmi a nie so štyrmi. Úroveň obtiažnosti pre obidva typy žiakov bola stredne ťažká.

Tabuľka 8. Zadanie úlohy 6 pre intaktných žiakov

Zmrzlinový stroj nakladá zmrzlinu do kornútikov špeciálnym spôsobom: do jedného kornútika dá vždy 4 kopčeky zmrzliny. Na obrázku vidno posledné štyri kornútiky so zmrzlinou, ktoré stroj práve pripravil.



**Ako bude vyzeráť ďalší kornútik?**

Tabuľka 9. Zadanie úlohy 6 pre nevidiacich žiakov

Stroj na zmrzlinu vyrába vždy iba trojkopčekovú zmrzlinu a dodržiava pri tom rovnaký systém pri zoraďovaní kopčekov. Prvé dve zmrzliny mali nasledujúce poradie kopčekov:

- Prvá zmrzlina:** jahodová, vanilková, čokoládová
- Druhá zmrzlina:** čokoládová, jahodová, vanilková
- Aké bude poradie kopčekov v tretej zmrzline?**
- A vanilková, jahodová, čokoládová
- B čokoládová, vanilková, jahodová
- C vanilková, čokoládová, jahodová
- D vanilková, vanilková, jahodová

Tabuľka 10. Zadanie úlohy 7 pre nevidiacich žiakov

Kresliaci robot pozná príkazy:

**krok** – urobí krok smerom, ktorým je natočený

**vpravo** – otočí sa na miesto vpravo,

**vľavo** – otočí sa na miesto vľavo,

**opakuj číslo [príkaz príkaz ...]** – robot zopakuje príkazy v zátvorkách toľkokrát, koľko udáva číslo pred zátvorkou

**Ktorým z týchto príkazov nakreslí robot obdĺžnik?**

- A opakuj 4 [krok vľavo]
- B opakuj 2 [krok krok vľavo krok vľavo]
- C opakuj 2 [krok krok vľavo krok vpravo]
- D opakuj 2 [krok krok vľavo krok krok vpravo]

Úloha **7 Robot** nebola zaradená v aktuálnom ročníku súťaže pre intaktných žiakov, ale v niektorom z predchádzajúcich. Už do pilotného kola sme zaradili podobnú úlohu, ale v porovnaní s pôvodnou úlohou bola jednoduchšia (tabuľky 2 a 3). Počas pilotného kola sme si všimli, že príkazy pre robota boli pre

nevidiacich pomerne komplikované. Rozhodli sme sa preto použiť jednoduchšie príkazy. Namiesto príkazu **dopredu 40** sme použili príkaz **krok** a namiesto príkazu **vľavo 90** (prípadne **vpravo 90**) sme použili príkaz **vľavo** (prípadne **vpravo**). Tým sa program skrátil a stal sa prehľadnejším a zrozumiteľnejším. Úlohu sme zaradili medzi ťažké, ako bola zaradená pôvodne.

Úlohu **8 Výťah** sme prevzali bez zmeny. Pôvodne však bola zaradená ako stredne ťažká. My sme ju zaradili ako ťažkú, pretože sa nám zdala byť pomerne abstraktná pre nevidiacich žiakov bez vizuálnej predstavivosti.

Úloha **9 Text odzadu** bola pre bežných žiakov zaradená ako ľahká. Podľa našich pozorovaní z pilotného kola, kde bola podobná úloha a nevidiaci žiaci s ňou mali problémy, sme túto úlohu zaradili medzi ťažké. Mierne sme ju zjednodušili tak, že namiesto trojslovného textu sme použili iba dvojslovný.

### 3.3 Súťažné úlohy v novembri 2014

V novembri 2014 sme úlohy vyberali spomedzi úloh určených pre kategórie Bobríci (2 úlohy) a Benjamíni (7 úloh). Tabuľka 11 znázorňuje porovnanie zaradenia úloh z hľadiska obtiažnosti pre intaktných a pre nevidiacich žiakov.

Tabuľka 11. Prehľad obtiažnosti úloh súťažného kola

Číslo	Názov	Téma	Obtiažnosť		Modifikácia
			nevidiaci	intaktní	
1	Kde sa hrá?	ALG	Ľahká	Stredná	áno
2	Tlač	PC	Ľahká	Ľahká	nie
3	Preukaz žiaka	INF	Ľahká	Ľahká	áno
4	Zubné kefy	ALG	Stredná	Ľahká	áno
5	Na nákupe	ALG	Stredná	Stredná	áno
6	Fotky	INF	Stredná	Stredná	áno
7	Číselný had	ALG	Ťažká	Ťažká	áno
8	Kajuty	INF	Ťažká	Stredná	áno
9	Lienka	ALG	Ťažká	Ťažká	áno

Z hľadiska tematického zamerania bola väčšina úloh zameraná na oblasť Postupy, riešenie problémov, algoritmické myslenie (5 úloh), pomerne dobre bola zastúpená aj oblasť Informácie okolo nás (3 úlohy). Oblasti Princípy fungovania digitálnych technológií sa týkala jedna úloha. Zvyšné dve tematické oblasti neboli pokryté vôbec.

Tento raz sme iba úlohu **2 Fotografie** prevzali bez zmeny a zaradili ju medzi ľahké, ako bola aj pôvodne. Všetky zvyšné úlohy si vyžadovali vykonanie menšej alebo väčšej zmeny. Pôvodné zadania úloh uvádzať nebudeme kvôli limitovanému rozsahu. Nachádzajú sa na webovej stránke súťaže iBobor<sup>3</sup>.

Tabuľka 12. Zadanie úlohy 1 pre nevidiacich žiakov

Bobor sa rozhoduje o tom kde sa bude hrať podľa nasledujúcich pravidiel:
<ul style="list-style-type: none"> <li>Ak je dnes slnečno, ale včera pršalo, pláva v rieke.</li> <li>Ak je dnes slnečno a včera bolo tiež slnečno, hrá sa na piesku na brehu rieky.</li> <li>Ak dnes prší, ale včera bolo pekne, hrá sa so stavebnicou doma.</li> <li>Ak dnes prší a včera tiež pršalo, nehrá sa.</li> </ul>
<b>Kde sa hrá bobor 7.novembra, ak v ten deň prší a 6.novembra tiež pršalo?</b>

Úloha **1 Kde sa hrá bobor?** v pôvodnom znení pre intaktných žiakov bol použitý obrázok. V úlohe pre nevidiacich žiakov sme

tento obrázok nahradili textom a požadovali sme od žiakov otvorenú odpoveď. Pôvodne bola úloha zaradená medzi stredne náročnými. My sme ju zaradili medzi ľahké, keďže aktuálny stav mali žiaci popísaný priamo v otázke a nemuseli si ho vyvodiť z obrázka. Rovnako aj pravidlo, ktoré mali použiť bolo uvedené hneď nad otázkou.

Úloha **3 Preukaz žiaka** vo verzii pre intaktných žiakov obsahovala preukaz zobrazovaný pomocou obrázka a okrem toho aj tabuľku s nevyhovujúcim lineárnym poradím. Museli sme ju preto upraviť. Všetky dôležité informácie sme museli žiakom poskytnúť v textovej podobe. Aby nebola táto textová informácia príliš dlhá a žiaci si ju mohli lepšie zapamätať, znížili sme počet možností.

Tabuľka 13. Zadanie úlohy 3 pre nevidiacich žiakov

Bobry dostávajú v škole preukaz, na ktorom je ich fotografia, meno, bydlisko, identifikačné číslo a dodatkové písmeno. Dodatkové písmeno sa získa nasledujúcim spôsobom:
<ul style="list-style-type: none"> <li>Spočítaj všetky číslice v identifikačnom čísle.</li> <li>Podľa získaného súčtu vyber dodatkové písmeno z nasledujúcich možností: <ul style="list-style-type: none"> <li>T: 1,3,5</li> <li>R: 6, 9, 11</li> <li>W: 2, 4, 8</li> <li>A: 7, 13, 10</li> </ul> </li> </ul>
<b>Aké dodatkové číslo má na svojom preukaze bobor Filip, ak je jeho identifikačné číslo 45?</b>

Aj úloha **4 Zubné kefy** mala dôležitú informáciu vo forme obrázka. Pokúsili sme sa ho opísať slovné a tak umožniť riešiť úlohu aj nevidiacim žiakom. Oproti pôvodnej úlohe, v ktorej museli žiaci pri riešení urobiť dve výmeny zubných kefiek, my sme túto výmenu už zahrnuli do aktuálneho stavu zoradenia zubných kefiek a žiaci mali len vybrať, ktorou ďalšou výmenou dostanú kefy do želaného poradia. Z tohto dôvodu sa nám úloha zdala byť oveľa jednoduchšia a priradili sme jej ľahkú náročnosť, hoci pôvodne bola zaradená medzi stredne ťažkými.

Tabuľka 14. Zadanie úlohy 4 pre nevidiacich žiakov

Mama bobrica má štyri malé bobry: trojročnú Anitu, päťročného Bruna, sedemročného Cyrila a jedenásťročnú Danku.
Malé bobry majú zubné kefy zavesené v kúpeľni od najmenšej po najväčšiu. Raz večer si ich omylom dali na nesprávne miesta. Keď prišla mama bobrica, kefy našla uložené takto:
Anitina, Cyrilova, Brunova, Dankina.
<b>Ktoré dve kefy musí mama bobrica vymeniť, aby boli zavesené v správnom poradí?</b>
<ul style="list-style-type: none"> <li>A. Brunovu a Dankinu</li> <li>B. Cyrilovu a Brunovu</li> <li>C. Dankinu a Anitinu</li> <li>D. Anitinu a Cyrilovu</li> </ul>

Úloha **5 Na nákupe** sa objavila v kategórii Bobríci, kde bolo rozmiestnenie regálov v obchode zobrazované pomocou obrázka. Žiaci mali zostaviť správne poradie v akom nakupovať požadovaný tovar tak, aby sa v obchode nemuseli kvôli ničomu vraci. Pre nevidiacich žiakov sme museli úlohu upraviť tak, aby sme čo najlepšie slovné popísali to, čo je na obrázku a aby textu nebolo príliš veľa.

Úloha **6 Ktorú fotografiu chceš?** bola zameraná na vyhľadávanie obrázkov podľa určitých vlastností. Bola použitá v dvoch súťažných kategóriách: Benjamíni a Bobríci. Aby sme úlohu mohli použiť s nevidiacimi žiakmi, museli sme možnosti popísať slovné tak, aby boli dostatočne zrozumiteľné. Tiež sme namiesto ôsmich obrázkov použili iba päť.

<sup>3</sup> <http://www.ibobor.sk>

V úlohe **7 Číselný had** mali žiaci vytvárať hada (postupnosť) z čísel. V pôvodnej úlohe, ktorou sme sa inšpirovali, išlo o hada z drevených kociek, ktorý bol zobrazený na obrázku. Pôvodné zadanie bolo teda vizuálne zamerané. Pôvodný had z kociek mal dĺžku 10 prvkov. My sme kocky nahradili číslicami a dĺžku sme skrátili na 6 prvkov.

**Tabuľka 15. Zadanie úlohy 7 pre nevidiacich žiakov**

Janka je veľká výmyselníčka a rozhodla sa pre svojho brata Ivka vytvoriť číselného hada. Najskôr poskladala svojho hada takto:

7,9,1,3,5,8.

Potom začala vytvárať nového číselného hada pre Ivka. Zobrala číslo z ľavého (Ľ) alebo z pravého (P) konca svojho hada. Toto číslo vždy dala na pravý koniec nového hada. Po umiestnení všetkých čísel vznikol pre Ivka takýto had:

8,5,7,3,1,9.

**Ako brala Janka čísla z pôvodného hada?**

- A. P, P, Ľ, Ľ, P, P
- B. Ľ, Ľ, P, P, Ľ, P
- C. P, P, Ľ, P, P, Ľ
- D. Ľ, P, P, Ľ, P, P

Úloha **8 Kajuty** v pôvodnom znení pre intaktných žiakov využívala farby a ich prekrývanie, čo bolo pre nevidiacich nevhodné. Navyše bola táto úloha interaktívna (výsledné farby okien sa získavali klikaním priamo na obrázok lode). Zadanie si preto vyžadovalo väčšiu úpravu ako iné úlohy, no snažili sme sa zachovať pôvodnú myšlienku. V našej úlohe pre nevidiacich žiakov sme znížili počet kajút z 8 na 6 a zaujímal nás iba stav v troch kajutách a nie vo všetkých. Obťažnosť sme preto odhadli na strednú, hoci pôvodne bola úloha zaradená medzi ťažkými.

**Tabuľka 16. Zadanie úlohy 8 pre nevidiacich žiakov**

Na lodi je 6 kajút a v každej sú 2 radiátory, jeden pri ľavej a druhý pri pravej stene. V každej izbe môže byť jeden z nasledujúcich stavov:

- Ak nie je zapnutý žiadny radiátor, v kajute je zima.
- Ak je zapnutý jeden radiátor, v kajute je teplo.
- Ak sú zapnuté dva radiátory, v kajute je horúco.

Radiátory pri ľavej stene zapli v kajutách 1, 4, 6 a pri pravej stene v kajutách 5, 4, 3, 1.

**Aká je teplota v kajutách 2, 4, 6?**

- A. 2-teplo, 4-zima, 6-teplo
- B. 2-zima, 4-teplo, 6-horúco
- C. 2-zima, 4-horúco, 6-teplo
- D. 2-teplo, 4-horúco, 6-teplo

Úloha **9 Robotická lienka** bola podobná ako už žiaci riešili v predchádzajúcom ročníku súťaže. V porovnaní s úlohou určenou pre intaktných žiakov sme ponechali zoznam príkazov, ktoré lienka ovláda a pýtali sme sa na postupnosť príkazov, ktorú má lienka vykonať, aby nakreslila obdĺžnik zadaných rozmerov. V úlohe pre intaktných žiakov mala nakresliť komplikovaný schodovitý útvar, takže išlo o jednoduchšiu verziu. Napriek tomu sme obťažnosť ponechali, pretože úloha bola pre nevidiacich žiakov bez vizuálnej predstavivosti pomerne abstraktná.

## 4. OVEROVANIE

Ako sme už uviedli vyššie, uvedené úlohy a navrhované prispôsobenia sme overili na špeciálnej škole pre zrakovo postihnutých žiakov počas reálneho behu súťaže v rokoch 2013 a 2014. Použili sme pri tom výskumnú stratégiu výskumu vývojom (vo viacerých iteráciách sme vyvíjali zadania úloh a analyzujeme ich vhodnosť) [6]. Pri zbere a analýze dát sme použili metódu zúčastneného pozorovania a analýzy žiackych riešení.

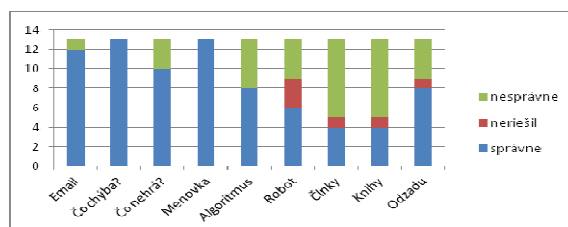
Naším cieľom bolo zistiť:

- či sú zadania úloh vhodné pre nevidiacich (zrozumiteľné, vhodne sformátované, vhodne zaradené podľa stupňa obťažnosti)
- či má na úspešnosť vplyv vek, schopnosť vizuálnej predstavivosti, úroveň počítačových zručností.

V roku 2013 sa do súťaže sa zapojili nevidiaci žiaci a žiaci so zvyškami zraku zo všetkých ročníkov druhého stupňa sledovanej ZŠ okrem šiesteho. V tomto ročníku bola iba jedna žiačka s uvedeným postihnutím, ktorá bola neprítomná kvôli ochoreniu.

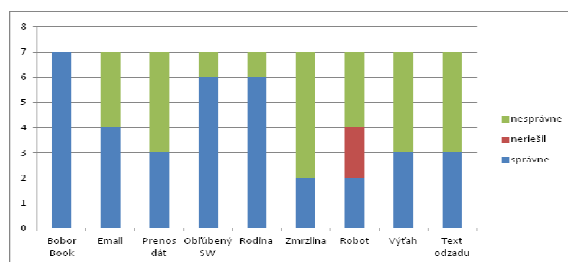
Pilotného kola sa zúčastnilo celkovo 13 žiakov (z toho 6 nevidiacich) a súťažného 7 žiakov (z toho 5 nevidiacich). Žiaci 5., 7. a 8. ročníka sa zúčastnili takmer v rovnakom počte pilotného, aj súťažného kola. Výrazný rozdiel v počte účastníkov jednotlivých kôl bol iba v 9. ročníku, kde sa pilotného kola zúčastnili piati žiaci a súťažného iba jeden žiak.

V roku 2014 sa zapojilo celkovo 7 žiakov zo všetkých ročníkov okrem deviateho. Žiaci 6. a 8. ročníka sa zúčastnili aj predchádzajúceho ročníka súťaže. Najviac žiakov bolo z 8. ročníka (4 žiaci).



**Obrázok 2. Graf úspešnosti pri riešení úloh pilotného kola**

Graf na obrázku 2 ilustruje pre každú úlohu koľko žiakov riešilo úlohu správne, koľko nesprávne a koľko žiakov úlohu neriešilo. Tieto informácie sú pre nás dôležité z hľadiska posúdenia obťažnosti úloh. Z grafu je zrejmé, že obťažnosť úloh 4 Menovka, 6 Robot a 9 Odzadu nebola správne odhadnutá. Úloha 4 mala byť v kategórii ľahkých úloh, nakoľko ju všetci žiaci vyriešili správne. Úloha 6 Robot mala byť zaradená v kategórii ťažkých úloh, pretože ju správne vyriešila necelá polovica žiakov. Úloha 9 Odzadu mala byť v kategórii stredne ťažkých úloh, pretože ju vyriešili správne takmer dve tretiny žiakov.

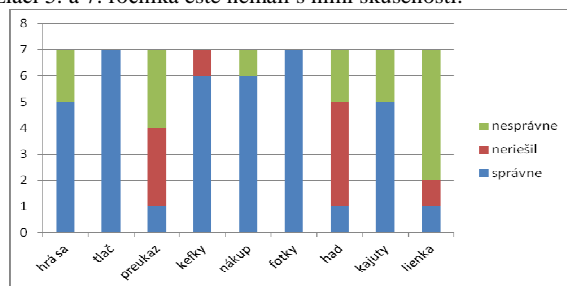


**Obrázok 3. Graf úspešnosti pri riešení úloh v roku 2013**

Venujme sa teraz úspešnosti žiakov v súťažnom kole v roku 2013. Z grafu na obrázku 3 vidíme, že viaceré úlohy neboli z hľadiska obťažnosti správne zaradené. Úlohy 4 Obľúbenosť programov a 5 Rodinné vzťahy mali byť pravdepodobne zaradené medzi ľahkými úlohami. Úloha 6 Zmrzlina naopak patrila medzi úlohy, s ktorými mali žiaci veľké problémy a pravdepodobne mala byť zaradená medzi ťažkými úlohami. Prekvapivo veľa problémov bolo pri riešení úlohy 3 Čo nemôže použiť? Nemyslíme si však, že úloha mala byť zaradená medzi ťažké úlohy. Pravdepodobne



mohla mať strednú ťažnosť. Pre nevidiacich je práca s pamäťovými médiami náročná kvôli ťažko prehľadným oknám a žiaci 5. a 7. ročníka ešte nemali s nimi skúsenosti.



Obrázok 4. Graf úspešnosti pri riešení úloh v roku 2014

Úspešnosť pri riešení úloh v roku 2014 je znázornená na obrázku 4. Potvrdilo sa, že úlohy 1, 4, 8 sme mali ponechať zaradené tak ako boli v kategóriách pre intaktných žiakov. Úloha 7 Číselný had sa ukázala byť ako ťažká, hoci bola aj pre intaktných žiakov zaradená ako ľahká. Tiež aj úlohy 5 a 6, ktoré boli prebrané z kategórie pre Benjamínov, kde boli zaradené ako stredne náročné sa ukázali byť ako ľahké pre nevidiacich žiakov. Úloha 3, ktorá bola zaradená medzi ľahkými úlohami, mala jednu z najnižších úspešností a mala byť zaradená medzi ťažké úlohy.

Tabuľky 17 a 18 sme vytvorili, aby sme mohli skúmať závislosť úspešnosti od veku, zrakových schopností (prípadne stupňa vizuálnej predstavivosti) a počítačových zručností.

Tabuľka 17. Prehľad schopností nevidiacich v roku 2013

Poradie	Body	Ročník	Viz. pred.	PC
1	51	7	áno	veľmi dobré
2	45,67	9	áno	veľmi dobré
3	28,68	5	áno	veľmi dobré
4	28,35	7	áno	stredné
5	26,01	5	nie	stredné
6	16,36	8	nie	slabé
7	12,36	7	nie	slabé

Tabuľka 18. Prehľad schopností nevidiacich v roku 2014

Poradie	Body	Ročník	Viz. pred.	PC
1	36	8	áno	veľmi dobré
2	23	8	áno	veľmi dobré
3	21,34	6	áno	veľmi dobré
4	21,34	8	nie	stredné
5	18,68	8	nie	stredné
6	5,68	7	nie	slabé
7	2,69	5	áno	veľmi slabé

Z údajov v treťom stĺpci je zjavné, že úspešnosť nezávisela od veku (respektíve ročníka) žiakov. Vo štvrtom stĺpci sú informácie o zrakových schopnostiach žiakov. Vidíme, že väčšinou boli úspešnejší žiaci, ktorí dokázali využívať vizuálne predstavy (t.j. žiaci so zvyškami zraku a tí, čo stratili zrak počas života). Najmenej úspešní boli žiaci nevidiaci od narodenia.

V poslednom stĺpci je informácia o tom aké boli počítačové zručnosti žiakov. Vidíme, že najúspešnejší boli žiaci, ktorí mali veľmi dobré počítačové zručnosti.

## 5. ZÁVER

Záverom môžeme konštatovať, že súťaž iBobor – kategória pre nevidiacich bola pre našich žiakov atraktívna. Dozvedeli sme sa to počas neformálnych rozhovorov s nimi. Úlohy považovali za primerane náročné a dostatočne zaujímavé. Traja najúspešnejší žiaci boli na školskom zhromaždení ocenení diplomom a škola im venovala USB kľúč.

Keďže väčšina úloh bola podobná úlohám pre intaktných žiakov – Benjamínov, nevidiaci žiaci mali pocit, že sa zapojili do rovnakej súťaže spolu s tisícmi svojich rovesníkov.

Počet úloh, ktoré mali žiaci riešiť počas 45 minút, bol vhodne zvolený. Žiadny zo žiakov nemal pocit, že bol v časovej tiesni. Náročnosť úloh sme prehodnotili a získali sme cenné skúsenosti pre kategorizáciu úloh v ďalších ročníkoch.

Analýza úspešnosti nasvedčuje tomu, že žiaci s vizuálnymi predstavami a so zvyškami zraku majú šancu lepšie uspieť ako žiaci nevidiaci od narodenia. Intaktní žiaci môžu pri riešení úloh používať pero a papier, čo im veľmi pomáha najmä pri riešení abstraktných úloh. Nevidiaci žiaci však túto možnosť nemajú. Otázne je, čo by bolo pre nich vhodnou alternatívou. Niektorí žiaci tvrdili, že si pomáhali tak, že kreslili prstom po nohe. Toto im však neumožnilo získať dostatočný náhľad porovnateľný s vizuálnou predstavou intaktných žiakov. Hľadanie vhodnej alternatívy ostáva pre nás výzvou do budúcnosti.

Úspešnosť v súťažnom kole tiež priamo úmerne závisela od stupňa počítačových zručností žiakov. Na základe výpovedí učiteľov matematiky, úspešnejší žiaci prejavovali aj lepšie logické myslenie. Potvrdenie, či vyvrátenie týchto tvrdení si však žiada realizáciu hlbšieho výskumu. V budúcom školskom roku opäť pripravíme úlohy pre kategóriu nevidiacich žiakov a pokúsime sa ich overiť s početnejšou vzorkou žiakov navštevujúcich podobné typy škôl na Slovensku a v Čechách.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] TOMCSÁNYIOVÁ, M. Informatické koncepty v úlohách súťaže iBobor In: Lovászová, G. (Eds.) *Didinfo 2014*, FPV UMB - Banská Bystrica, 2014. s. 79-88. ISBN 978-80-557-0698-6.
- [2] DAGIENĚ, V. Competition in Information Technology: an Informal Learning, In: *Proceedings Eurologo 2005*, Warsaw, Poland, 2005. pp. 228 – 234. ISBN 83-917700-8-7.
- [3] HRUŠECKÁ, A. et. al. Informatický bobor – nová súťaž v informačných technológiách pre žiakov základných a stredných škôl. In Huraj, L. (Eds.) *Didinfo 2008*, Banská Bystrica : Univerzita Mateja Bela, 2008. ISBN 987-80-8083-556-9.
- [4] ALLMAN, C. B. *Test Access. Making Tests Accessible for Students with Visual Impairments: A Guide for Test Publishers, Test Developers, and State Assessment Personnel*. 4. Vydanie, American Printing House for the Blind, Louisville, Kentucky, 2009.
- [5] PASCH, M. et al. *Od vzdelávacieho programu k vyučovací hodine*, Praha : Portál, 1998. ISBN 80-7178-127-4
- [6] KALAŠ, I. Pedagogický výskum v informatike a informatizácii (2. časť). In: Rován, B. (Eds.) *Didinfo 2009*. Banská Bystrica : Univerzita Mateja Bela, 2009. ISBN 978-80-8083-720-4.

# Informatika v bicím sekvenceru ukrytá

Jan Jára

Katedra informatiky a výpočetní  
techniky, Pedagogická fakulta,  
Jihočeská univerzita  
Jeronýmova 10  
Česká republika  
Jara@pf.jcu.cz

## ABSTRAKT

Článek popisuje experiment s vnímáním posluchačů programovat bicí online sekvencer jako informatickou záležitost. Zamýšlí se nad schopností posluchačů vytvářet návrh shora dolů, mapováním prostředků jako základních stavebních kamenů, ze kterých bude dílo stvořeno, hledání zákonitostí, aplikace vlastních poznatků. To všechno v prostředí pro posluchače záměrně naprosto cizím – v bicím sekvenceru. Jsou tak vystavováni situaci, kdy z vágního zadání, v problematice, kterou musí vstřebat, s využitím svého informatického backgroundu, hledají způsob, jak problematiku uchopit, zpracovat a určit kritéria hodnocení. To samotné čeká samozřejmě i vyučujícího, který práce následně hodnotí. Bylo analyzováno 26 vzorků od posluchačů za dva semestry. Analýza probíhala pomocí parsování zdrojového souboru a následného zpracování. Tato aktivita byla ceněna na doškolovacím setkání učitelů informatiky konaném na Katedře informatiky v roce 2014 jako atraktivní motivační úloha.

## ABSTRACT

The article describes the experiment with the perception of the students programing the drum sequencer online as a matter of informatics. It reflects on the listener's ability to generate design top-down, mapping resources as the basic building blocks from which the work will be created, search patterns, the application of their knowledge. All this in an environment for the students deliberately utterly alien - the drum sequencer. They are so exposed to a situation where a vague assignment, in the issues which must be absorbed, using its informatics background, looking for a way to grasp the issues, process and identify evaluation criteria. That alone, of course, waiting for the teacher to work subsequently evaluated. It analyzed 26 samples from the students for two semesters. The analysis was carried out using the source file parsing and subsequent processing. This activity was appreciated at in-service teachers' meeting held at the Department of Informatics in 2014 as an attractive motivational problem.

## Klíčové slová

evaluace, autoevaluace, informatika, práce se vzory, drum pattern sequencer, bicí smyčky

## Keywords

evaluation, self-evaluation, informatics, working with patterns, drum pattern sequencer, drum loops

## 1. ÚVOD

Experiment byl proveden u posluchačů prvního ročníku na Pedagogické fakultě Jihočeské univerzity v Českých

Budějovicích na Katedře informatiky v Předmětu principy a využití multimédií.

Realizujte bicí smyčku s dějem. Délka kolem jedné minuty.	
scénář:	použité efekty:
<ul style="list-style-type: none"><li>• úvod</li><li>• varianty</li><li>• vrchol</li><li>• závěr</li></ul>	<ul style="list-style-type: none"><li>• plíživé nástrojové či rytmické obměny</li><li>• (práce s tempem)</li><li>• práce se stereem</li><li>• práce s hlasitostí</li></ul>
<ul style="list-style-type: none"><li>• odevzdává se: 1_nazev.txt - zdrojový kód (save), 2_nazev.mp3 - aranžovaný výsledný produkt</li><li>• (použit bounce dostane se .wav a ten převést na .mp3)</li></ul>	

Zadání, je velmi vágní, přesto obsahuje určitá omezení. Na rozdíl například od zadání v oblasti programování, např. "Naprogramujte řešení soustavy 4 lineárních rovnic pomocí Gaussovi eminační metody", nelze se chytout nějakého známého algoritmu či postupu. Tato hudební úloha je záměrně dávana posluchačům informatiky, u kterých se nepředpokládají znalosti a zkušenosti z tohoto oboru. Simuluje tedy budoucí reálnou situaci, kdy informatik musí vstřebat jemu neznámou problematiku, orientovat se v ní a realizovat příslušné řešení buď pomocí programů či nastavení systémů. V diskusi před realizací, si posluchači uvědomí nějaké zákonitosti, například jak na člověka působí pravidelnost rytmu a jeho případné poruchy.

## 2. ONLINE DRUMBOT

Pro seznámení posluchačů předmětu multimédií s tímto typem SW popřípadě HW, byl zvolen on-line program Drumbot. Uživatel může vytvářet až šest motivů pod označením A – F. Každý motiv může obsahovat různý počet taktů, a různou jemnost členění. K dispozici jsou však pouze násobky dvou. Takže tři čtvrtový valčík, či 5/4 rytmus lze vytvořit pouze obtížně. Každý motiv může obsahovat až 8 nástrojů, u kterých si v rámci celého motivu uživatel vybírá hlasitost a balanc. Jednotlivé motivy lze skládat libovolně za sebou. Výsledný projekt lze uložit v proprietárním txt souboru, nebo jako zvuk ve formátu wav. Výsledné práce byly pochopitelně kontrolovány a hodnoceny převážně sluchem, před autorem zvukového díla a ostatními posluchači, s autorovým komentářem. Později se ukázalo, že textový soubor s bicí sekvencí je analyzovatelný, a vnáší do problematiky hodnocení vytvořeného projektu jiný úhel pohledu. Níže uvedená ukázka je část vyexportovaného souboru.

Tab 1. Zadání práce

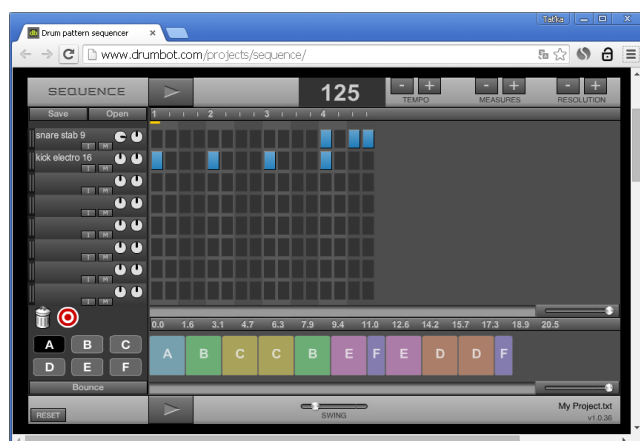


Tento soubor byl autorem článku ručně zalamován a odsazován pro lepší pochopení jeho struktury. V původní verzi je jako jedna dlouhá řádka. Postupným parsováním v tabulce Google, a následným zpracováním byly získány sekvence motivů, jejich délky a dělení, struktury rytmů a nástrojů, a pomocí spider grafu i jejich rozložení. Analyzováno bylo 29 vzorků. Z toho vzorek č. 2 byl vzorový od tvůrce programu a tudíž nesouvisel s výukou. Byl však považován jako referenční. Vzory č. 11 a 27, byly vyřazeny pro algoritmickou obtížnost zpracování dat.

Tab 2 . Ukázka části textového souboru s projektem

```
{
  "swing":0.2,
  "order":[0,1,2,2,1,4,5,4,3,3,5],
  "bpm":125,
  "parts":
  {
    {
      "measures":4,
      "channel":[
        {
          "label":"snare stab 9","data":"snare_stab_009"},
        {
          "label":"kick electro
16","data":"kick_electro_016"},
        {
          "label":"","data":"null"},
        {
          "label":"","data":"null"},
        {
          "label":"","data":"null"},
        {
          "label":"","data":"null"},
        {
          "label":"","data":"null"},
        {
          "label":"","data":"null"}],
      "iso":[0,0,0,0,0,0,0,0],
      "divs":4,
      "grid":[
        [0,1,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [0,1,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [0,1,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [0,0,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [1,1,0,0,0,0,0,0], [0,0,0,0,0,0,0,0],
        [1,0,0,0,0,0,0,0], [1,0,0,0,0,0,0,0]],
      "volume":[0.7941176470588236,0.5,0.5,0.5,0.5,0.5,0.5,0.5],
      "mute":[0,0,0,0,0,0,0,0],
      "pan":[0,0,0,0,0,0,0,0],

```



Obr. 1 Online bicí sekvencer a motiv A, popsáný v

### 3. INFORMATIKA A SEKVENCE ÚDERŮ?

Základní otázkou, kterou si čtenář jistě položí: "Co má společného informatika a sekvence úderů?" Obdobná otázka by mohla být: "Má návrh na pletení svetrů něco společného s informatikou?" Domnívám se, že ano. U svetrů musíte plánovat

jak operace na 1D, se promítanou ve 2D, popřípadě v 3D. U bicího sekvenceru je situace při daném zadání ještě složitější.

### 3.1 Návrh shora dolů

V nejvyšší úrovni si řešitel musí vymyslet scénář: Úvod – varianty – vrchol – závěr. Ten pak rozložit do sekvence motivů A – F. Dále pak řešit návrh jednotlivých motivů a jejich vzájemnou provázanost. Dále pak jaké použít výrazové prostředky a jak se používají. To je v podstatě návrh shora. Je příjemné, když vidíte posluchače tímto způsobem přemýšlet. Ukazuje to, že dokáží strukturovaně a systematicky pracovat.

### 3.2 Lineární návrh

Určitě to jde i od začátku do konce. Postupně přidávat nástroje a jejich rytmické obměny, obdobně jako např. u Bolera op. 81 skladatele Maurice Ravela, či Passacaglia a fuga c mol (BWV 582) skladatele J.S.Bacha. Je potřeba zachovat provázanost jednotlivých motivů, zajistit dodržení scénáře: Úvod – varianty – vrchol – závěr. To ovšem zúženým průřezem: předchozí takt – současný – následující, je obtížně realizovatelné. Takže korekce po pohledu shora jsou nutné.

### 3.3 Bezmyšlenkovitý návrh

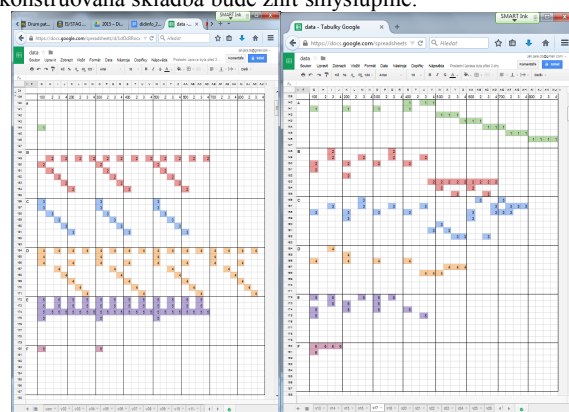
Řešitelům se ukazuje, že tento přístup nevede k cíli. Bicí sekvence je přece jenom jemné předivo svázané pravidly, aby stačilo jen náhodné rozmístění úderů. Přes rozdílnou míru autoevaluace jednotlivých řešitelů vrhne toto řešení řešitele zpět na začátek, k hledání přece jenom strukturovaného návrhu.

## 4. PŘÍSTUPY ŘEŠITELŮ

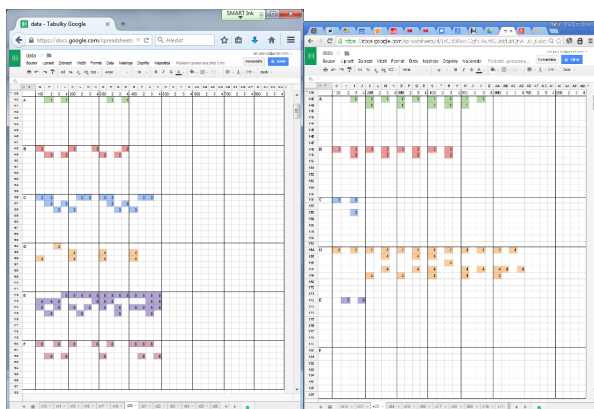
Pro určitou skupinu posluchačů byla úloha zpočátku neuchopitelná. A někteří ji hodnotili jako nejobtížnější z celého semináře, protože nevěděli jak úlohu řešit. Teprve v následující týden, když jsme si znovu zopakovali některá pravidla začali pracovat. V souboru zkoumaných prací se vyskytují práce různých přístupů.

### 4.1 Geometrické obrazce

Tyto práce se vyznačují nápadnými geometrickými vzory. Výhoda těchto postupů je ta, že téměř nelze minout. Každá takto konstruovaná skladba bude znít smysluplně.



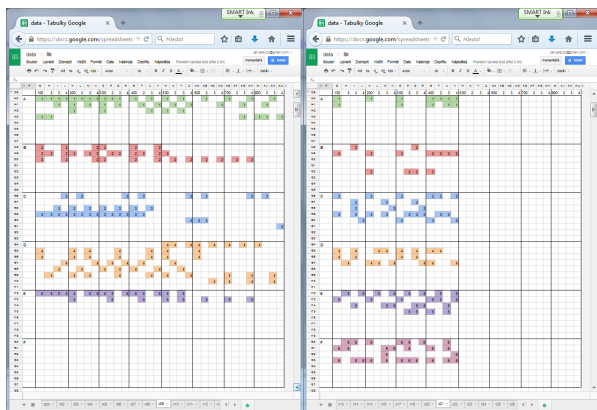
Obr. 2 Nápadné geometrické vzory. Vzorek č. 26, 17



Obr. 3 Nápadné geometrické vzory. Vzorek č. 20, 32

## 4.2 Bohaté verze

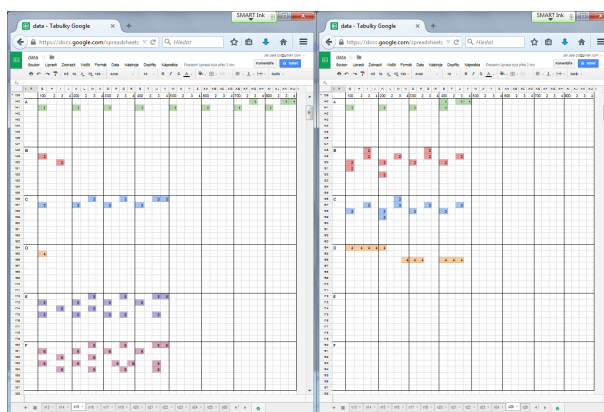
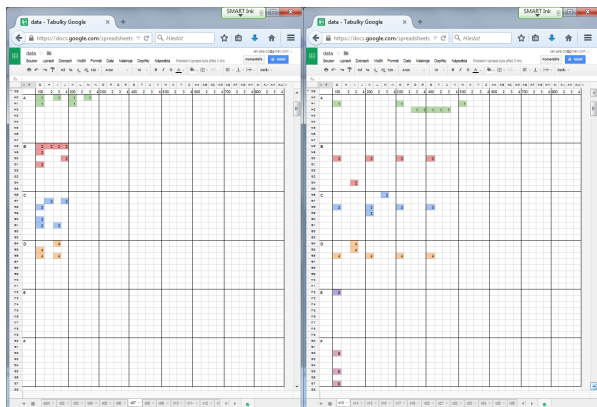
Pokud počet úderů na takt je výrazně vyšší než je obvyklé, budeme hovořit o bohaté verzi. Mohou vznikat například počáteční nadšením z tvorby a pocitem: "Čím je toho víc, tím lepší." Vytrácí se vodící rytmická linka. Zřejmě návrh vznikl zespona.



Obr. 4 Bohaté verze. Vzorek 9, 21.

## 4.3 Úsporné verze

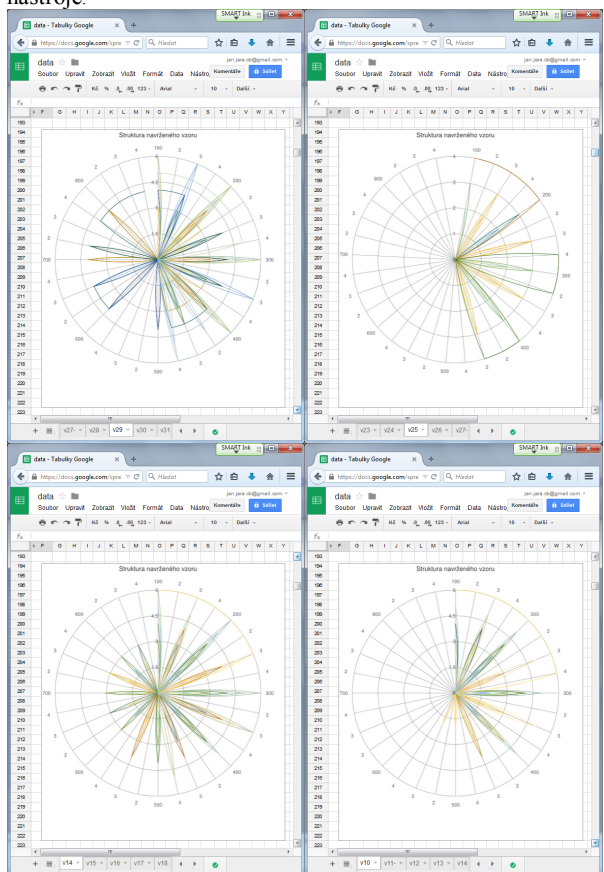
Málo not, nebo málo motivů. Promyšlené střídání opakování motivů. Zjištění: Kratší motivy jsou pohodlnější na údržbu/tvorbu, mohou se z nich skládat celky, které poslouchající snáze pochopí. Lze tedy tento způsob tvorby chápat jako kognitivně náročnější - promyšlenější, než výše uvedená "bohatá verze".



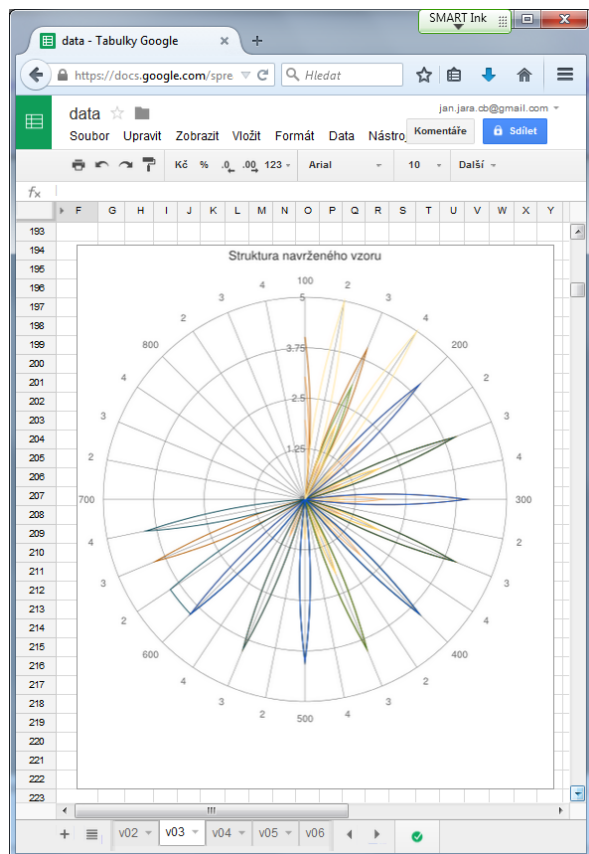
Obr. 5 Úsporné verze. Vzorek 7, 13, 15, 25.

## 4.4 Práce s tichem

Někteří autoři si uvědomili možnost práce s tichem. Systematickou práci s tichem na první pohled odhalí spider graf. V dané velikosti obrázku, nejsou patrné všechny detaily, ale pro tento záměr nejsou potřeba. Delší čísla u os jsou čísla taktů, kratší: doby 2, 3, 4. Těžkou dobu si nenechá nikdo ujít, ale sudé - lehké doby v těchto případech jsou tiché - tedy bez úderu nástroje.



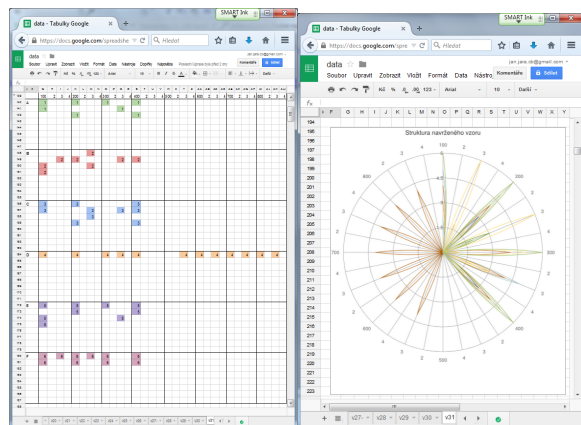
Obr. 6 Spider grafy a práce s tichem. Vzorek 29, 25, 14, 10



Obr. 7 Spider grafy a práce s tichem.  
Vzorek 3.

## 4.5 Generální pauza

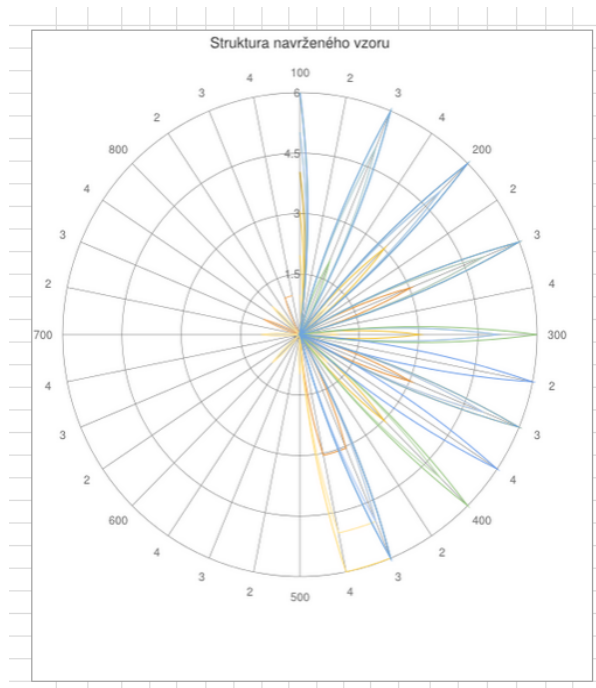
Výrazným výrazovým prostředkem je generální pauza. V uvedeném vzorku č. 31 se motiv D s generální pauzou opakuje 4x. Sekvenci [E,A,A,B,B,A,C,C,D,A,A,B,B,D,D,C,C,A,B,B,A,A,F,F,B,C,E,A,A,D,F,F,F,B,A,D] navrhovala posluchačka, hrající na bicí v kapele a nazvala ji RocknRoll. Generální pauza v pracích není výjimečná. Vzorek 05 má generální pauzu na 13 dob, vzorek 09 na 8 dob, vzorek 13 na 7 dob.



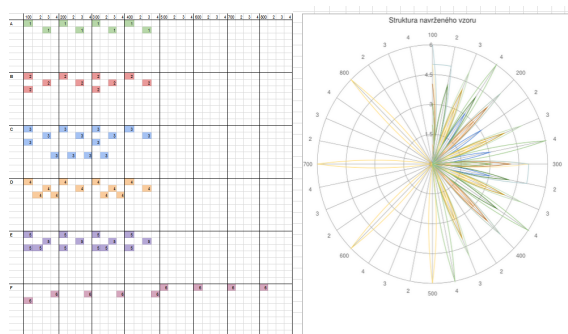
Obr. 8 Generální pauza. Vzorek 31.

## 4.6 Korespondence motivů

Ve zkoumaných vzorcích lze najít práci s motivy, kdy autor se snaží o rytmickou podobnost v jednotlivých motivech. Ve spider grafech se projevují posílenými lístečky jen v určitých dobách. Práce však ukázaly případy, kdy motivy mezi sebou nekorrespondují, a při poslechu budí dojem násilné výměny typu pošoupnutí gramofonové jehly o několik drážek. Tento efekt byl zatím posuzován jenom sluchem. Subjektivně autoři těchto prací budí dojem neochoty se danou problematikou zabývat do hloubky.



Obr. 7 Korespondence motivů. Vzorek 17 - spider graf



Obr. 7 Korespondence motivů. Vzorek 6, struktura rytmu a spider graf - motiv F v grafu kružnice 6 přesun rytmu z lehké doby na těžkou dobu.

Korespondenci motivů se zabývá i práce [3]. V tomto zdroji však analyzují přímo zvukové soubory. V našem případě nemusíme identifikovat jednotlivé údery, protože jsou přímo předepsané autorem. Nicméně způsob zobrazování podobných rytmů pomocí Similarity matrix může být zajímavou alternativou k zde uváděným spider grafům.

## 5. AUTOEVALUACE A EVALUACE

Jak dílo hodnotit?

### 5.1 Z pohledu tvůrců

Všichni posluchači hodnotili své dílo poslechem jako celek. Někteří použili výše uvedené způsoby tvorby, což jim dávalo pocit, že to nemůže být úplně špatně. (Geometrické vzory, provázanost jednotlivých motivů, aj.)

### 5.2 Z pohledu vyučujícího

Bylo potřeba najít způsob objektivizace. Při evaluaci byly veřejně hodnoceny tato kritéria.

- Dodržení předepsaného scénáře. V případě nejasností, doplňující otázky typu
  - ukažte vrchol,
  - proč je to vrchol,
  - které motivy a jejich kombinace jste použil pro 2. bod - varianty scénáře
- Provázání jednotlivých motivů
- Pokusy o přístup shora, zdola, geometrické vzory

Výše uvedená kritéria byla použita spíše jako informativní vzhledem k zaměření předmětu. Důležité bylo, že posluchači něco odevzdali, naučili se pracovat s neznámým programem pro tvorbu multimédií, nesli kůži na trh - před ostatními předváděli své dílo.

## 6. ZÁVĚR

Článek ukázal, že tvorba bicí sekvence/smyčky nemusí být čistě hudební problematika, ale může odkrývat nejen schopnost strukturovaného myšlení, tolik potřebného v informatice, ale i schopnost vstřebat a pochopit neznámou problematiku. Pokud se prokáže, že tato myšlenka hodna dalšího zkoumání, lze výzkum zpřesnit následujícím způsobem:

- možnost citlivěji analyzovat
  - nevypouštět složité motivy
  - řadit osmici podle názvu nástrojů, aby protékaly na stejném místě přes všechny motivy
- předepsat strukturu
  - počet taktů
  - dělení
  - hudební formu např. Jazzový standard [2]:  
i (intro) A [: X :] A c (coda)

Použitý postup zpracování a myšlenka mohou být použity jako motivační úloha či motiv pro projektovou výuku.

## 7. BIBLIOGRAFICKÉ ODKAZY

- [1] *Šablóna príspevku konferencie DIDINFO 2015* [online]. [cit. 2015-03-02]. Dostupné na: <http://didinfo.umb.sk/sk/168/informacie-pre-autorov.html>
- [2] NEUŽIL, J, BENKO, M. *Scripta základů hudební teorie pro uchazeče o studium oboru Hra na klavír na KJJ* [online]. [cit. 2015-03-02]. Dostupné na: [http://download.kjj.cz/pub/vyuka/2010/kl/skripta\\_kl.pdf](http://download.kjj.cz/pub/vyuka/2010/kl/skripta_kl.pdf).
- [3] ZAFAR RAFII. *Rhythm Analysis in Music* [online]. [cit. 2015-03-02]. Dostupné na: <http://www.cs.northwestern.edu/~pardo/courses/eecs352/lectures/MPM12-rhythm.pdf>.

# Programovanie, alebo hra?

Alžbeta Kanáliková  
Elektrotechnická fakulta, KRIS,  
Žilinská univerzita  
Univerzitná 1/  
010 26 Žilina  
Slovakia  
[alzbeta.kanalikova@gmail.com](mailto:alzbeta.kanalikova@gmail.com)

## ABSTRAKT

Článok sa zaoberá programovaním v objektovo orientovanom jazyku JAVA, konkrétne s programovaním pre začiatočníkov. Obsahuje praktické ukážky hodín programovania v prostredí *Greenfoot*. Prostredie *Greenfoot* je primárne určené pre programovanie jednoduchých hier. Praktická skúsenosť práce so študentami pri výučbe OOP prostredníctvom programovania jednoduchých hier je pozitívna. Študenti rýchlejšie pochopia základné objektové pojmy, ale hlavne sú viac motivovaní k tvorivej činnosti.

## ABSTRACT

The article deals of programming in object oriented programming language JAVA, specifically Java programming for beginners. The article include practical examples of exercises programming in environment *Greenfoot*. *Greenfoot* environment is intended primarily for programming simple games. Practical experience working with students in teaching OOP through games and pre-built applications is positive. Students understand quickly the basic concepts of object and are more especially motivated to creative activity.

## Kľúčové slová

Java, objektovo orientované programovanie, *Greenfoot*, výučba v OOP, hra.

## Keywords

Java, object oriented programming, *Greenfoot*, teaching of OOP, game.

## 1. ÚVOD

Ako dobre vyučovať programovanie začiatočníkov, konkrétne objektovo orientované programovanie, to je otázka každého učiteľa informatiky. Podobne ako pri každom predmete aj pri výučbe programovania je dôležitá motivácia. Jednou s možných motivácií je výučba prostredníctvom programovania jednoduchých hier. V zahraničí je venovaný tejto problematike výskum tzv. vzdelávanie prostredníctvom hier *Game based learning*, kde študent prostredníctvom hrania hier nadobúda vedomosti, resp. potrebné schopnosti a zručnosti.[1] Niekoľkoročná skúsenosť s výučbou programovania na vysokej a strednej škole v prostrediach *Pascal*, *Delphi* a hlavne v prostredí *Bluej* ma priviedla k hľadaniu lepšej motivácie pre študentov ako naučiť študentov programovať tak aby sa zvýšil ich záujem. Jednou z mnohých a veľmi vhodných metódik je metodika *Object*

*first*, ktorú som aplikovala vo výučbe v prostredí *Bluej*, kde som viedla výučbu k vytvoreniu konečnej aplikácie od návrhu. Výučba úvodných hodín bola interaktívna a študenti sa základne pojmy učili na vopred vytvorenom príklade. Výučba v prostredí *Bluej* pokračovala postupným vytváraním výslednej aplikácie. Po absolvovaní seminára, ktorý sa venoval jazyku JAVA som našla aj inú alternatívu k prostrediu *Bluej*, prostredie *Greenfoot*, ktoré vychádza z prostredia *Bluej* a obsahuje už vopred pripravené triedy na programovanie hier. Podelením sa o skúsenosti s kolegami, ktorí sa venujú výučbe OOP a hľadaním ďalších ma zaujala výučba OOP prostredníctvom vytvárania hier. A prepojenie vhodného prostredia na vytváranie hier – *Greenfoot*, ktoré zároveň vychádza z prostredia *Bluej* a je veľmi podobné a zároveň vhodnej motivujúcej myšlienky ma priviedlo k rozpracovaniu návrhu metodiky v tomto prostredí. V nasledujúcom texte je rozpracovaný stručný návrh metodiky prvých hodín úvodných hodín výučby OOP. Návrh metodiky nie je dokonalý, ale vychádza z mojich skúseností a so skúseností mojich kolegov na seminári s prostredím *Greenfoot*. Primárne by mala byť určená pre študentov stredných škôl druhého, alebo tretieho ročníka, prípadne po určitých úpravách aj pre študentov prvých ročníkoch vysokých škôl. V budúcnosti plánujem metodiku aplikovať zatiaľ len prostredníctvom krúžkovej činnosti na strednej škole.

## 2. VÝVOJOVÉ PROSTREDIE GREENFOOT

*Greenfoot* je IDE, t. z. vývojové prostredie v objektovo orientovanom jazyku JAVA, ktoré vzniklo na základe projektu výskumnej skupiny edukačných nástrojov University of Kent, UK. Prostredníctvom IDE (*Integrated Development Environment*) *Greenfoot* je určené na vytváranie jednoduchých hier, alebo rôznych interaktívnych aplikácií. Vzhľadom k svojim vlastnostiam je toto vývojové prostredie aj vhodným motivačným prostredím pre výučbu programovania v objektovo orientovanom programovacom jazyku hlavne pre začínajúcich študentov. *Greenfoot* má jednoduché grafické rozhranie, ale je zároveň plne funkčným vývojovým prostredím s riadením projektov, dopĺňaním syntaxe, dokumentáciou, "debugovaním", editorom zdrojového kódu a pod. Zároveň obsahuje vytvorené triedy tzv. okolitého sveta - *World* ako aj triedy aktérov, resp. postáv *Actor* - herec, postava – z ktorých sa dá vyskladať jednoduchá interaktívna hra, alebo iná aplikácia. [2]



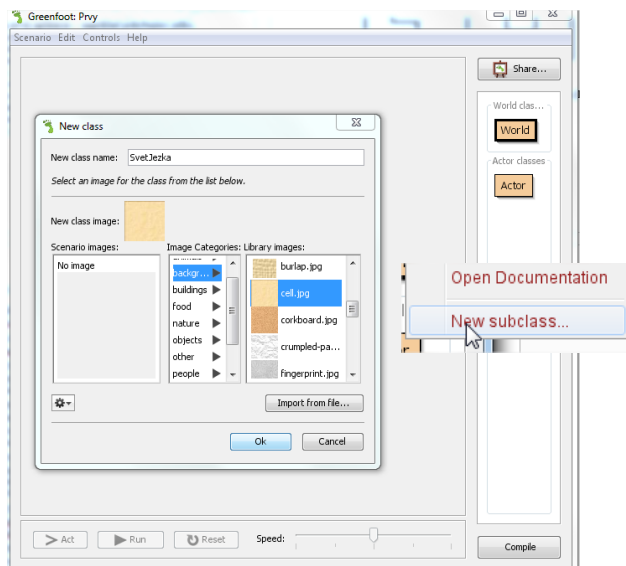
### 3. ZÁKLADNÉ POJMY OOP A PRÁCA V PROSTREDÍ GREENFOOT

#### 3.1 Základné pojmy OOP

Na začiatku študentom, ktorí sú začiatčníci v programovaní je potrebné vysvetliť základné pojmy OOP. Základné pojmy sú pojmy objekt, trieda, metóda, prípadne konštruktor. Študenti by na začiatku mali pochopiť základné vnímanie OOP, kde sa premieta vnímanie sveta, ktorý sa skladá z objektov. Trieda je abstraktným pojmom pre všetky objekty rovnakého typu a prostredníctvom nej vzniknú všetky tieto objekty. Napríklad trieda „Jezko“ bude triedou z ktorej môže vzniknúť niekoľko objektov - niekoľko ježkov. Každý objekt má svoje vlastnosti – atribúty, ktoré sú jemu vlastné a často chránené – slovo *private* a má svoje správanie – vyjadrené metódami. Tieto metódy pre každý objekt sú často všeobecne používané – *public*. Na začiatku výučby OOP tento pojmový aparát zatiaľ postačuje. Neskôr sa postupne dopĺňa a aj rozširuje.

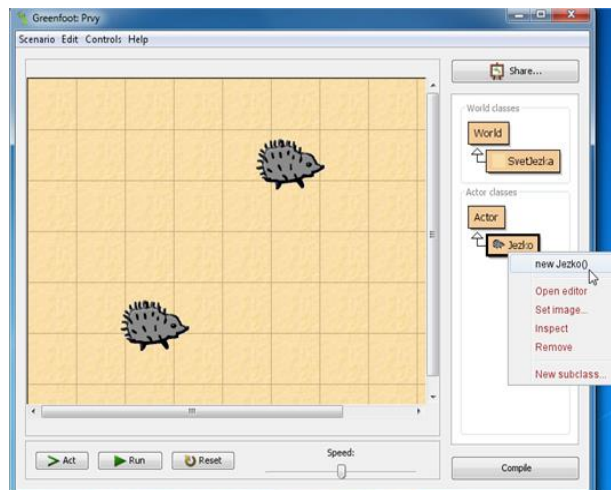
#### 3.2 Vytvorenie nových tried a ich objektov

Pri otvorení nového projektu sú aktivované dve základné triedy - *World* a *Actor*, od ktorých všetky ostatné triedy dedia metódy. Podobne pri návrhu prvej aplikácie - napríklad pohybujúci ježko, ktorý je ovocie je potrebné navrhnuť tzv. svet aplikácie - *World* kde je možné na začiatku navrhnuť aj pozadie a druhá trieda sú samotní účastníci hry, *Actors* tzv. postavy. Pri vytváraní je možné vybrať z viacerých pozadí sveta a viacerých postáv z rôznych kategórií. V našom prípade si môžeme zvoliť za pozadie napr. hnedú plochu (neskôr ju upresníme) a ako hlavného aktéra v aplikácii zvolíme napr. ježka (nazveme ho *Jezko*). Študentov upozorníme, že vytvárajú šablóny budúcich tried, ktoré neskôr naplnia kódom z jazyka JAVA. Vytváranie nových tried je znázornené na obr. 1, kliknutím na triedu *World* alebo triedu *Actor*. Túto činnosť veľmi rýchlo zvládnu všetci študenti.



Obrázok 1 Vytvorenie tried v prostredí Greenfoot

Po vytvorení zatiaľ len šablón tried je vhodné ukázať čo je to objekt a ako si vytvoríme nový objekt. Predtým je ale potrebné projekt skompilovať. Tu máme ďalší pojem – kompilácia. Vysvetlíme ho ako preklad programovacieho jazyka do strojového kódu, tak aby tomu počítač rozumel, teda do jednoduchšej formy ako je kód jazyka JAVA. Kompilácia je zrealizovaná prostredníctvom tlačidla *Compile* - kompilácia. Jednotlivé triedy zmenili farbu a z tzv. šrafovaného tvaru vzniká trieda (obdĺžnik) s ružovou farbou. Prostredie dokáže vytvoriť, resp. zobrazíť na ploche objekt nášho sveta, resp. zatiaľ iba pozadie a podobne aj objekt ježka. Na názornej ukážke študenti spustia konštruktor – *new Jezko()* a vzniká nový objekt. Je možné vytvoriť aj niekoľko objektov triedy typu „*Jezko*“. Pozadie a nové objekty sú znázornené na obr. 2.

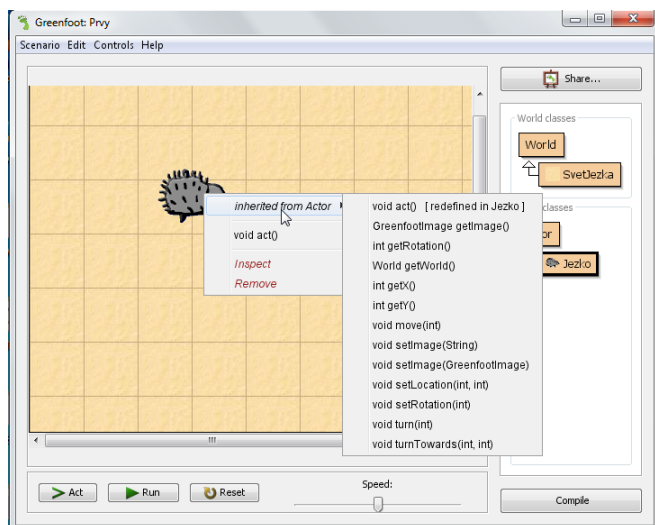


Obrázok 2 Vytvorenie objektov v prostredí Greenfoot

#### 3.3 Práca s metódami

Vytvorené objekty, ale aj objekt tzv. sveta – *World* a náš zdedený objekt z triedy „*SvetJezka*“ môže využívať zatiaľ iba zdedené metódy z triedy *World*. Tieto metódy napríklad vedú: zmeniť veľkosť, obrázok v pozadí, vymazať objekty v tomto prostredí atď. Z triedy *Actor*, aktérov nová trieda teda v našom prípade napr. trieda „*Jezko*“ dedí a teda aj jej objekty metódy, prostredníctvom ktorých sa môže objekt napríklad: pohybovať, otáčať, alebo zmeniť súradnice objektu, alebo dať informáciu o týchto súradniciach a pod. Jednotlivé zdedené metódy sa predstavujú študentom, ktorí ich majú možnosť následne vyskúšať. Vytvorený objekt triedy „*Jezko*“ má aj svoju metódu s názvom *act()*, ktorá je prázdna, ale má slúžiť na vytvorenie konkrétnej aktivity, činnosti aktéra, resp. postavy. Napríklad pohyb, v našom prípade objektu „*Jezko*“, alebo otočenie a zjedenie jablka. Metódy sa zobrazia kliknutím pravého tlačidla myši na konkrétne objekty. Prácu s metódami znázorňuje obr. 3.





Obrázok 3 Používanie zdedených metód v objektoch

#### 4. ZDROJOVÝ KÓD A VYTVORENIE HRY

Scenár prvej hry, resp. interaktívnej aplikácie, ktorú má učiteľ pripravenú musí byť jednoduchý. Napríklad úlohou je aby sa objekt z triedy „Jezko“ pohyboval a jedol jablká. Študenti, ktorí začínajú s programovaním pochopiteľne nie sú schopní vyriešiť túto úlohu samostatne. Vhodným spôsobom je postupne riešiť tento problém s učiteľom na vopred pripravenom príklade aplikácie, ktorú si učiteľ vopred vytvoril.

Prvým krokom je rozobrať scenár hry, resp. aplikácie podrobnejšie. Ako už bolo spomenuté okrem pohybu chceme aby jezko jedol aj jablká, to znamená, že je potrebné vytvoriť Triedu Jablko, ktorá bude vytvárať objekty jablka. Ďalšia časť vytvorenia aplikácie sa rieši už v zdrojovom kóde, t. z. učiteľ zobrazí zdrojový kód tried. (pravým tlačidlom myši na triede). Prvý otvoríme kód triedy sveta. Ak chceme aby sa na začiatku na zobrazovanej ploche tzv. „SvetJezka“ zobrazil objekt jezka a potom na rôznych miestach objekty niekoľkých jablk je potrebné vytvoriť tieto objekty napríklad prostredníctvom metódy priamo v triede sveta. Zároveň pre ľahšie riešenie je vhodné si zobrazovaný svet rozdeliť iba na 8x8 buniek, alebo 10x10 po 50 - 60pixelov. Jednotlivé časti kódu učiteľ ukáže a doplnuje priamo do triedy. Rozmery sveta sa zadávajú priamo do konštruktora.

Zložitejšou úlohou je naprogramovanie metód v triede postavy, napr. v triede „Jezko“. Podľa scenáru aplikácie máme vytvoriť napr. metódu pohybu objektu, otočenia objektu a zjedenia jablka v triede „Jezko“. Každá metóda sa vytvorí samostatne. Veľkou výhodou je, že v prostredí existujú už hotové metódy napr. presunu objektu - napr. `move(4)`, alebo otočenia objektu - napr. `turn(-1)`, vymazania objektu - napr. `world.removeObject(jablko)` a pod., ktoré môžeme pri použití pri vytváraní našich metód.

Podobne ako pri návrhu sveta - *World* učiteľ by mal najprv so študentami prediskutovať podrobnejšie činnosti v každej metóde, tak aby študenti porozumeli čo bude krok po kroku metóda robiť. Napríklad ak objekt nájde jablko – musí tam byť podmienka či ho

našiel a až následne ho zje, teda objekt Jablko sa vymaže zo sveta, prípadne následne si pripočíta k počítadlu, že zjedol ďalšie jablko.

Analýza problémov vedie študentov k rozvíjaniu nielen k analytickému ale aj algoritmickému myslenia, tak že v ďalších úlohách už sami začnú takto uvažovať. Zároveň ich motivuje k vytváraniu ďalších úloh a aj zložitejších zadanií.

Prvé metódy študenti vytvárajú spolu s učiteľom, ale od začiatku je vhodné vytvoriť priestor na samostatnú prácu, napríklad šikovnejší študenti, alebo študenti, ktorí už niečo programovali môžu vytvárať nové metódy. Učiteľ však vysvetľuje kód, resp. jeho úpravu od najjednoduchších metód po zložitejšie.

Na záver sa vytvára väčšinou metóda `void act()`. Metóda `void act()` je metóda, ktorá aktivuje objekt pri spustení aplikácie – kliknutím na tlačidlo **RUN** napr. aby sa objekt pohyboval, následne otočil pri náraze na hranicu tzv. sveta a zároveň ak nájde jablko, tak ho zje. Často v tejto metóde voláme ďalšie metódy a práve preto sa väčšinou vytvára ako posledná. Obr.4 znázorňuje príklad zdrojového kódu metód.

```
public void act()
{
    if(nasielJablko()) {
        zjedolJablko();
    }
    else if(mozePresun()) {
        move();
    }
    else {
        otocVlavo();
    }
}

// Ci nasiela jablko
public boolean nasielJablko()
{
    Actor jablko = getObjectAtOffset(0, 0, Jablko.class);
    if(jablko != null) {
        return true;
    }
    else {
        return false;
    }
}

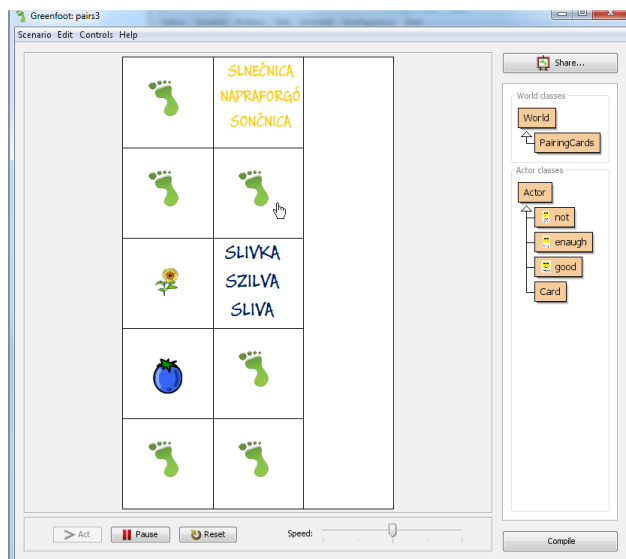
//Metoda na zjedenie jablka a jeho zmiznutie
public void zjedolJablko()
{
    Actor jablko = getObjectAtOffset(0, 0, Jablko.class);
    if(jablko != null) {
        // je jablko a spocita kolko..
        getWorld().removeObject(jablko);
        jablkoZjedol = jablkoZjedol + 1;
    }
}
```

Obrázok 4 Zdrojový kód s metódami

#### 5. ZLOŽITEJŠIE PRÍKLADY

Postupnosťou od jednoduchších aplikácií cez analýzu celkovej aplikácie, ako aj kódu je študent vedený k zložitejším aplikáciám tak, aby vedel neskôr samostatne navrhnúť svoju hru, resp. aplikáciu. Postupným rozborom kódov je študent prakticky vovedený aj do syntaxe jazyka. Naučí sa používať podmienky, cykly, tak, že vníma ich potrebnosť pretože cieľom je dosiahnuť konkrétny postup, algoritmus, ktorý si predtým zanalyzoval. Príkladom takejto hry, resp. aplikácie je napríklad aplikácia PEXESO, ktorá zobrazuje druhy ovocia v rôznych jazykoch. Študent mal vymyslieť metódy, ktoré dokážu obracať kartičky na ploche a rozhodnúť o tom kedy sú kartičky rovnaké, resp. kedy sa

hra končí, keď daný hráč dosiahol bodov a pod. Príklad hotovej aplikácie je na obr. 5.



Obrázok 5 Príklad hry Pexeso

## 6. ZÁVER

Hra, alebo učenie? Praktická skúsenosť práce so študentami pri výučbe OOP prostredníctvom hier a vopred vytvorených aplikácií je pozitívna, ako vyjadrujú aj viacerí učitelia [3]. Študenti rýchlejšie pochopia základné objektové pojmy, ale hlavne sú viac motivovaní k tvorivej činnosti. Veľký význam zohráva aj zvolené prostredie ako aj spôsob vedenia študenta k vytvoreniu jednotlivých hier, resp. aplikácií. Na základe skúseností aj viacerých učiteľov na školách by bolo vhodné zdokonaľiť metodiku výučby prostredníctvom hier, resp. simulačných aplikácií a rozpracovať ju pre širšiu verejnosť učiteľov stredných alebo aj vysokých škôl.

## 7. BIBLIOGRAFICKÉ ODKAZY

- [1] *Game-based learning* [online]. [cit. 2015-03-11].  
Dostupné na:  
<<http://www.nfer.ac.uk/publications/GAME01/GAME01.pdf>>
- [2] *About Greenfoot* [online]. [cit. 2015-02-07]. Dostupné na:  
< <http://www.greenfoot.org/overview>>
- [3] *Skúsenosti s metodikou výučby programovania pomocou hier* [online]. [cit.2015-02-08]. Dostupné na:  
<<http://wrack.ped.muni.cz/konference2011/prisp/nab1/prisp4.pdf>>

# Použití mechatronických systémů jako motivačního prvku nejen ve výuce informatiky

RNDr. Jan Krejčí, Ph.D.  
Přírodovědecká fakulta, UJEP  
České mládeže 8  
40096 Ústí nad Labem  
Česká republika  
jan.krejci@ujep.cz

Ing. Jiří Šretr  
Mechatronic Education spol. s r.o.  
Václava Řezáče 313  
434 01 Most  
Česká republika  
sretr@datacomsro.cz

## ABSTRAKT

V současné době, kdy se svět stává čím dál tím více závislý na elektronických technologiích, se zjišťuje, že zájem o technické předměty upadá. Proto je nemálo projektů a snah, které mají vzbudit zájem o tyto náročné obory. V rámci pedagogické praxe se však vyučující těchto předmětů potýká se základními otázkami: Jakým způsobem vzbudit zájem o problematiku technických oborů? Jak implementovat poznatky z průmyslové činnosti do edukačního procesu? Jak zajistit potřebnou techniku pro praktické vyučování? Motivaci studentů těchto disciplín je možné zajistit použitím mechatronických didaktických systémů. Kde jednotlivé studentem realizované aktivity jsou funkčním modelem reálných studovaných procesů. Vlastní softwarový algoritmus a jeho implementace je pak většinou identická jako ve vlastní průmyslové praxi.

## ABSTRACT

Today's world is getting more and more dependent on technologies but, however, we discover that there is less and less interest in technical courses of study. Thus there are many projects and attempts designed to raise interest in these demanding subjects. The teachers of these subjects face the following essential issues within their teaching practice: How to raise interest in the problems of technical studies? How to implement the knowledge from industrial activities into the educational process? How to ensure the necessary technologies for practice learning? It is possible to motivate students of such courses by using mechatronic didactic systems where each executed student's project is a functional model of the real studied process. Their algorithms and their implementations are usually identical to the ones in the industrial practice.

## Klíčová slova

mechatronika, senzory, motory, autonomní systémy, programování mikro-kontrolérů, programování android, měření neelektrických veličin, student střední školy, student univerzity

## Keywords

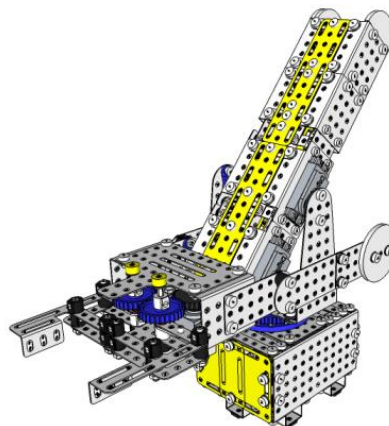
mechatronics, sensors, engine, autonomous systems, programing of microcontrollers, programing of android device, measurement of non-electrical quantities, high education student, student of university

## 1. Úvod

Mechatronika je další z široké škály oborů, kde absolventi oboru informatiky mohou uplatnit znalosti a dovednosti získané studiem. Tento mladý a vysoce perspektivní obor propojuje softwarové odborníky s reálným světem průmyslu, automatizace a robotizace. Autonomní robotické mechanismy mohou pro širší veřejnost působit dojmem něčeho velmi složitého, použitelného až v daleké budoucnosti a možná i pro budoucí profesi studentů až příliš

nadčasového. Opak je však pravdou. S mechatronickými výrobky se setkáváme v každodenním životě, počínaje ranní kávou z domácího kávovaru, naplánováním cesty do zaměstnání pomocí mobilního telefonu, automatizovanou platbou v platebním terminálu, či při relaxaci sledováním moderních interaktivních televizí.

Užitná hodnota těchto výrobků spočívá v důmyslných algoritmech a softwarových technologiích, kde interakce software s fyzickým a reálným světem přináší řešiteli, v našem případě studentům, řadu překážek. Tyto překážky dokáže zdolat pouze za použití kreativity, spojené se znalostí a dovedností z široké škály oborů a neotřelým či dokonce inovativním řešením. Naše katedra informatiky proto přistoupila k výuce, kde studenti učitelského studia informatických oborů pomocí didakticky připravených výukových setů od společnosti Mechatronic Education spol. s r.o. (1), mohou realizovat celou řadu funkčních modelů strojů, mechanismů a zařízení. Získají tím dovednosti a zkušenosti s měřením elektrických i neelektrických veličin, regulací a řízením. Nejen na



Obrázek 1: Model robotické ruky

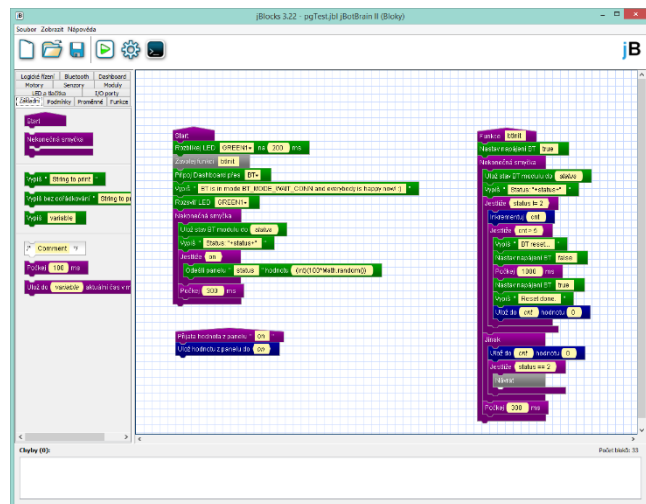
autonomních systémech student pochopí, proč musel absolvovat teoretickou přípravu na hodinách matematiky a fyziky. Progresivní na celém procesu je fakt, že student si vlastně ani neuvedomí, že tyto dovednosti využívá, nicméně bez jejich zvládnutí stroj, zařízení či mechanismus nezpřevodí.

Vize spočívá ve snaze vychovat generaci schopných specialistů, kteří v praxi budou schopni automatizovat výrobní procesy, osvobodit tyto procesy od stereotypní práce, zvýší produktivitu výroby a vrátí do Evropy konkurenceschopnost. Prvním krokem v celém procesu je v první řadě výchova odborně vzdělaných mladých pedagogů, kteří v budoucnu budou na odborných školách tyto specialisty připravovat. Současně s přípravou didakticky orientovaných studentů probíhá i výuka odborných studentů studijního programu Aplikovaná informatika na naší katedře. Vysoké procento technických škol, ať již středních či vysokých,

disponuje větší či menší řádkou sofistikovaných elektrických systémů, které byly pořízeny za značné finanční prostředky, jen málo škol však má těchto zařízení dostatek, aby na nich mohly vést řádnou výuku. Tento závažný problém se dá snadno vyřešit použitím levných a přesto operativně modifikovatelných systémů, které nahradí celou řadu jednoúčelových přístrojů.

## 2. MOTIVACE STUDENTŮ

Naši studenti mají obecně o problematiku informatických předmětů velký zájem, ale ve spojení s praktickou projektovou výukou, která je založena na principu jednoduchého programování ve speciálním vizuálním prostředí nazvaném jBlocks (podobné systému SCRATCH (2)), který může student používat i bez předcházející znalosti vlastní syntaxe, klíčových slov programovacího jazyka či jednotlivých objektů tříd, je zájem o to větší.

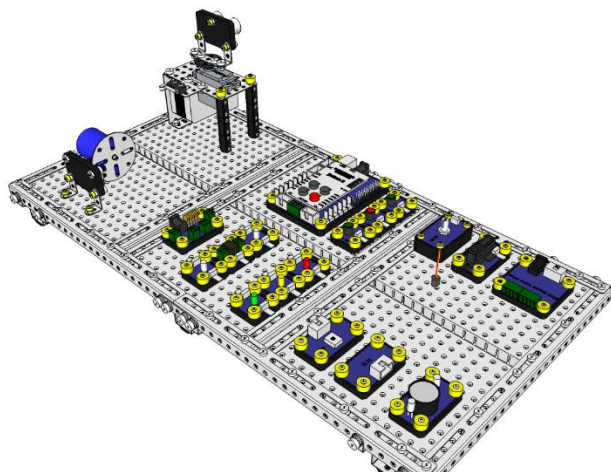


Obrázek 3: jBlock

Díky jednoduchosti programování není na studenty kladen v podstatě žádný počáteční požadavek na kompetence v oblasti programování a zdrojový kód mohou studovat jen v případě vlastního zájmu.

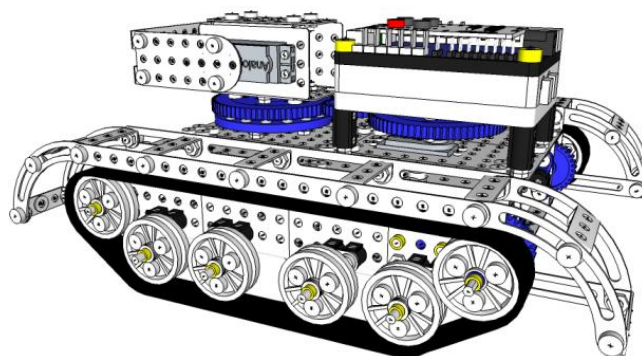
## 3. VÝUKOVÝ PROCES

Studenti v rámci výuky na jednoduchých úlohách napřed zjišťují vlastní možnosti systému na přehledné a praktické výukové desce, na kterou mohou připojit kromě centrální desky s mikro-



Obrázek 5: Pracovní deska

kontrolérem i velké množství senzorických komponent, nebo výstupních modulů.

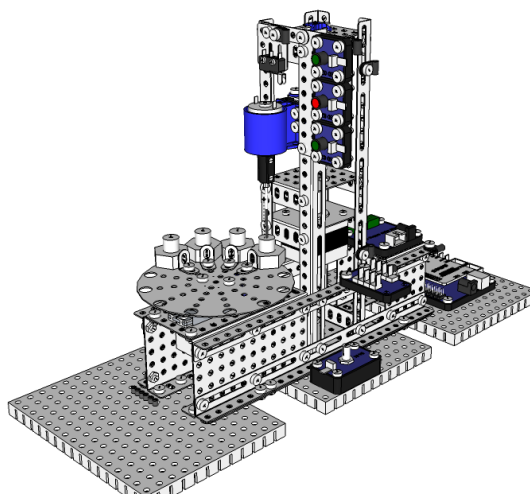


Obrázek 2: Model dálkově ovládaného vozítka

Součástí didaktických setů jsou výukové materiály, které dokáží studenta provést jednotlivými kroky výukového procesu a téměř v každém kroku mu pomoci a poradit. Součástí stavebnic jsou pochopitelně také návody, potřebné pro sestavení základních mechanických modelů a odpovídající kódy pro jejich oživení. Prvními kroky výuky jsou tedy kromě sestavení modelů spíše modifikace kódů, čímž se studenti seznamují s vlastními objekty a strukturami programování.

V případě výuky u nás na univerzitě pak v rámci seminárních či bakalářských prací i následně vytvářejí vlastní nové modely, ke kterým píšou ovládací kódy zahrnující nejrůznější automatizační algoritmy, bezdrátové komunikační protokoly a následně model ovládají pomocí mobilních vzdálených systémů např. prostřednictvím mobilních telefonů či tabletů, pro které si také naprogramují potřebné aplikace.

Model si student napřed nakreslí v 3D volně dostupné aplikaci SketchUp (3) do které jsou importovatelné objekty jednotlivých stavebních dílů. V rámci tohoto procesu se student seznamuje se základními podmínkami konstrukce mechanických strojů a vozidel. Při vlastní konstrukci pak téměř pokaždé přijde na to, že to, co si tak precizně vytvořil v rámci 3D výkresu, není reálné ve skutečnosti postavit. Proto studenti své projekty nejdříve upravují a optimalizují právě v procesu vlastní stavby fyzického modelu.



Obrázek 4: Model CNC vrtačky



Díky kovovým stavebním dílům je možno vytvořit realistický model skutečného systému. Modely mohou být vybaveny různými elektrickými motory, od klasických až po servomotory. Použitím servomotorů se například dá vytvořit model CNC stroje, který po doplnění o aplikaci přepisující G-kód na příkazy mikro-kontroléru dokáže simulovat drahé CNC přístroje v hodnotě třeba i několika milionů korun.

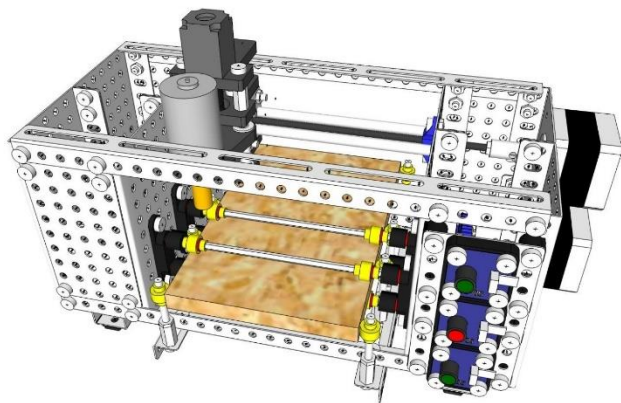
Student tak v rámci jednoho projektu projde vývojem mechatronického systému od návrhu a konstrukce až k jeho naprogramování a kalibraci včetně vytvoření ovládacího software.

#### 4. PŘÍKLADY STUDENTSKÝCH PRACÍ

V rámci prvních let implementace systému do výukového procesu u nás na katedře došlo kromě pravidelné výuky i k tvorbě závěrečných prací za použití výukových setů.

Na následující sekvenci obrázků jsou znázorněny návrhy nejnovějších modelů, které byly následně realizovány. V rámci prací byly ověřeny jejich funkcionality a součástí odevzdaných prací jsou i návody pro samotnou stavbu a oživení daných modelů.

V roce 2013 byly vypracovány bakalářské práce zaměřené hlavně na CNC řízení prostřednictvím reálně používaného G-kódu. V rámci bakalářských prací byly vytvořeny modely CNC gravírovací frézky (4) a CNC vyřezávačky z polystyrénu (5), další úlohou připravovanou pro případné využití v rámci přípravy středoškolských studentů může být model 2D skeneru (6), založeném na principu plotrového zapisovače ALFI od společnosti Merkur Toys spol. s r. o. (7). Součástí prací byla i aplikace, která dokáže konvertovat G-kód do instrukcí řídicí jednotky a tak jsou systémy připraveny pro výuku obsluhy CNC systémů právě použitím G-kódu, který se používá na průmyslových systémech (frézky, soustruhy, atp.).



Obrázek 6: Model CNC gravírovací frézky

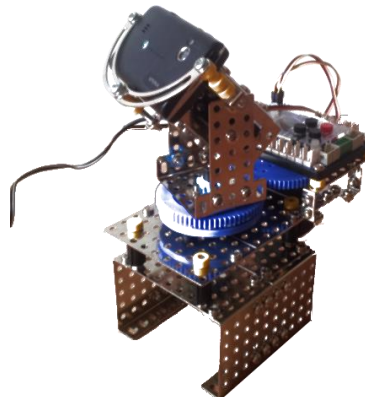
Na obrázku č. 7 je vyobrazen reálný model automatické zaměřovací věže z roku 2014 kombinovaný s mobilním telefonem, který zajišťuje kamerový systém po WiFi připojení, posílá obraz do počítače (notebooku), jehož software dále řeší zpracování obrazu a následně dává vstupní informace prostřednictvím komunikace bluetooth řídicí jednotce motorů. (8)

#### 5. SHRnutí

Výuka technických předmětů je velmi potřebná, avšak v rámci teoreticky zaměřené výuky se dá jen těžko ve studentech vzbudit zájem o problematiku. Jedním z možných směrů, kterým se vydat, je využití mechatronických systémů pro výuku, které studentům či žákům zpřístupní celý proces konstrukce, řízení a programování sofistikovaných zařízení. Zajištění skutečných přístrojů by bylo velmi finančně náročné, avšak použití stavebnicových systémů je i

několika řádově levnější a přesto v rámci laboratorní výuky umožňuje přesně stejné funkcionality jako stroj průmyslový.

Na naší katedře se výuka pomocí mechatronických setů společnosti Mechatronic Education spol. s r.o. velmi osvědčila a studenti při hodnocení výuky nešetří pochvalami. Navíc tito studenti velmi snadno najdou pracovní uplatnění a věnují se zajímavým profesím v odvětvích, které jsou i v současném propadu hospodářství stále na vzestupu.



Obrázek 7: Reálný model automatické zaměřovací věže (8)

#### 6. PODĚKOVÁNÍ

Poděkování patří především společnosti Mechatronic Education spol. s r.o. za úzkou spolupráci a v neposlední řadě pak děkuji projektu CZ.1.07/2.2.00/28.0296 za finanční pomoc při realizaci výuky.

#### 7. Bibliografie

1. **Mechatronic Education spol. s r.o.** Sety. *Mechatronic Education spol. s r.o.* [Online] Mechatronic Education spol. s r.o., 12. 12 2014. [Citace: 02. 02 2015.] <http://test.mechatroniceducation.cz/sety>.
2. **SCRATCH.** [Online] Lifelong Kindergarten Group, Massachusetts Institute of Technology. [Citace: 14. 03 2015.] <https://scratch.mit.edu/>.
3. **Limited, Trimble Navigation. Home. SketchUP.** [Online] [Citace: 08. 02 2015.] <http://www.sketchup.com/>.
4. **Bárta, Jan a Haberzettl, Petr.** *Vytvoření funkčního modelu CNC Gravírovací frézky.* Katedra informatiky, Přírodovědecká fakulta. Ústí nad Labem : Univerzita J.E. Purkyně v Ústí nad Labem, 2013. str. 64, Bakalářská práce.
5. **Rudolf, Jan a Haberzettl, Petr.** *Vytvoření funkčního modelu CNC Vyřezávačky z polystyrénu.* Katedra informatiky, Přírodovědecká fakulta. Ústí nad Labem : Univerzita J.E. Purkyně v Ústí nad Labem, 2013. str. 79, Bakalářská práce.
6. **Tichý, Pavel a Krejčí, Jan.** *Vytvoření funkčního modelu 2D skeneru.* Katedra informatiky, Přírodovědecká fakulta. Ústí nad Labem : Univerzita J.E. Purkyně v Ústí nad Labem, 2013. str. 49, bakalářská práce.
7. **Merkur Toys spol. s r. o. Plotrový zapisovač z MERKURu. MERKUR.** [Online] Merkur Toys spol. s r. o., 19. 04 2011. [Citace: 14. 03 2015.] <http://www.merkurtoys.cz/plotrovy-zapisovac-z-merkuru>.
8. **Radoň, Jan a Krejčí, Jan.** *Model zbraňové věže s automatickým zaměřováním.* Katedra informatiky, Přírodovědecká fakulta. Ústí nad Labem : Univerzita J.E. Purkyně v Ústí nad Labem, 2014. str. 77, Bakalářská práce.

# Tvorba kurikula pro výuku informatiky jako vědy pro česká gymnázia

Daniel Lessner

Katedra software a výuky informatiky, Matematicko-fyzikální fakulta, Univerzita Karlova v Praze

Malostranské nám. 25

118 00 Praha 1

Česká republika

lessner@ksvi.mff.cuni.cz

## ABSTRAKT

V České republice informatika dosud není zařazena do všeobecného vzdělávání. V rámci snahy o zlepšení tohoto stavu jsme realizovali rozsáhlejší projekt, který zahrnoval vytvoření konceptu výuky úvodu do informatiky, jeho ladění, úpravy a závěrečné vyhodnocení. Ukázalo se, že jde o téma sice náročné, ale zvládnutelné a jednoznačně prospěšné.

V tomto článku popisujeme dílčí část projektu, totiž tvorbu programu výuky. Není k dispozici tradiční a osvědčené řešení, o které by bylo možné se opřít, proto jsme začali s čistým štětem a na základě identifikace fundamentálních idejí didakticky transformovali obor informatiky pro gymnázium. Vybrali jsme pojmy informace, model, algoritmus a efektivita jako ty, na nichž vzdělávání vystavět. Lze je nahlížet na různé úrovni pokročilosti, objevují se opakovaně v různých oblastech místech informatiky, přitom mají svůj význam i mimo ni. Na závěr článku krátce komentujeme související výsledky pilotáže.

## ABSTRACT

Computer science is not a part of general education in the Czech Republic. We have carried out a larger project to help to improve this situation. It involved development of a new introductory computer science course concept, its iterative testing and adjusting and final piloting and evaluation. Computer science turned out to be difficult, yet manageable by our students, and computer science education showed some clear benefits.

In this article we describe a part of this project, the initial curriculum development. There was no traditional and well tested approach we could rely on. So we started with clear slate and carried out the didactic transformation of computer science for grammar school. Our approach was based on identifying fundamental ideas, such as information, model, algorithm and efficiency. These are the foundation of the resulting curriculum. They can be studied on different age levels, they appear in various areas of computer science and they are relevant also outside of it. We conclude with a short comment on the pilot run evaluation.

## Klíčové slová

Výuka informatiky, tvorba kurikula, didaktická transformace, fundamentální ideje, gymnázia.

## Keywords

Computer science education, curriculum development, fundamental ideas, grammar schools.

## 1. ÚVOD

V České republice informatika dosud není zařazena do všeobecného vzdělávání. Žáci se v souladu s Rámcovými

vzdělávacími programy (RVP) učí využívat digitální technologie a pracovat s informačními zdroji, prakticky vůbec se ale nesetkají s informatikou jako vědou. Obvyklá zdůvodnění takového stavu se opírají mimo jiné o názory, že to není potřeba a že to nejde. Teprve ve světle vývoje v zahraničí se česká vláda v roce 2014 rozhodla situaci změnit a informatiku vyučovat.

Abychom si odpověděli na otázku, jestli je výuka informatiky ve všeobecném vzdělávání možná a prospěšná, vytvořili a ověřili jsme program takového úvodu do informatiky pro gymnázium. V rozhodujících parametrech je srovnatelný s dalšími školními předměty jako matematikou či fyzikou. Je také určený všem žákům, nejedná se tedy o učivo pro výběrové semináře.

V tomto článku představíme proces, jakým jsme hledali vhodné pojetí informatiky, které by splnilo výše uvedené požadavky, a výsledek, ke kterému jsme došli. Práci jsme zahájili rozsáhlou rešerší, zahrnující jak vymezení oboru informatiky, tak různá vymezení informatiky jako školního předmětu. Pro naše potřeby jsme se rozhodli využít konceptu fundamentálních idejí. Po jejich identifikaci jsme přistoupili k jejich rozpracování na konkrétní vzdělávací cíle a dále k přípravě programu výuky. Nakonec se podařilo experimentální výuku informatiky realizovat i s úplnou třídou po dobu celého školního roku.

Na začátku článku popíšeme obor, který jsme identifikovali jako výchozí akademickou disciplínu pro experimentální školní předmět informatika. V následující kapitole se věnujeme postupu didaktické transformace oboru na školní předmět a souvisejícím aspektům. V kapitole 4 představíme výsledek transformace a v kapitole 5 výsledky uskutečněné pilotáže.

## 2. INFORMATIKA JAKO OBOR

Před dalším postupem bylo třeba si ujasnit, co má vlastně být předobrazem budoucího školního předmětu informatika. V češtině se k pojmu informatika váží různé významy, v dalších zemích je situace obdobná. V běžném životě tato mnohoznačnost příliš nevádí. V různých odborných kruzích mnohoznačnost eliminuje ustálený konsenzus o významu pojmů. Za takové situace ovšem nelze beze všeho přistoupit k tvorbě kurikula. Hrubou představu o výuce informatiky jako vědy (v protikladu k uživatelským či technickým dovednostem) bylo nutno značně zpřesnit a ujasnit, co do oboru patří, co je na hranici a co do něj nepatří.

Za tímto účelem jsme prozkoumali množství zdrojů jak českých, tak i zahraničních. Základní vymezení poskytují encyklopedie, zejména při srovnání významu hesel v různých jazycích. Významným zdrojem porozumění je pohled do historie pojmů, ale i samotných oborů. Konkrétnější obsah (a různá pojetí i totožně pojmenovaných oborů) představují sylaby vysokoškolských vzdělávacích programů. Nejpodrobnější pohled potom nabízí



přehledové monografie jako [1–3]. Průzkum nějakým způsobem souvisejících disciplín vymezení informatiky dále zpřesní. Jedná se např. o matematiku, informační technologie, softwarové inženýrství, informační vědu, kybernetiku, lingvistiku a mnohé další obory, které mají s informatikou nějaké styčné plochy.

Provedená rešerše se ukázala jako opodstatněná, na obor informatika existuje množství odlišných pohledů. Na druhé straně se také podle očekávání potvrdilo, že existuje určité společné jádro, ze kterého je možné vycházet. Kdybychom měli zvolit stručné vymezení oboru informatika, patrně bychom volili mezi formulacemi Philipa Guo „Efficiently Implementing Automated Abstractions“ [4], Jeanette Wing „Computing is Automation of Abstraction“ [5] a Petera J. Denninga „the study of information processes, natural and artificial“ [6]. Bližší prozkoumání ukazuje, že se všechna tři vymezení téměř bezezbytku překrývají. Rozdíl je tedy spíš v důrazu na jednotlivé aspekty, např. Guo poněkud zdůrazňuje aplikační stránku informatiky, naproti tomu Denningova formulace se zdá na člověku téměř nezávislá.

Jako velmi užitečná se ukázala související práce Petera J. Denninga [7], kterou zde krátce přiblížíme. Denning se pokusil informatiku kodifikovat. Tradiční struktura vysokoškolského kurikula postavená na jednotlivých technologiích se stávala nepřehlednou pro potřeby popisu informatiky jako oboru. Denning proto v průběhu let identifikoval základní principy, které technologie přesahují, a přitom jsou pro informatiku charakteristické. Výsledek se podobá přístupu v dalších vědách, které se také organizují podle nějakých obecnějších principů. V případě CS se (podle Denninga) jedná o:

- výpočet (computation),
- komunikaci,
- koordinaci,
- automatizaci,
- vybavování (z paměti, recollection).

Uvedené principy se samozřejmě leckde překrývají. Denning v této souvislosti hovoří o jednotlivých oknech, jimiž na informatiku nahlížíme. Uvedený model se v čase vyvíjí. K uvedeným mechanismům např. přibýlo pět kritérií kvalitního návrhu (jednoduchost, výkonnost, spolehlivost, přizpůsobitelnost a bezpečnost). Z nich potom plynou principy jako abstrakce, hierarchizace a další.

Navazující oblast Denningovy práce je snaha o klasifikaci informatiky. Na informatiku lze nahlížet jako na abstraktní vědu (podobnou matematice, s technologiemi coby pouhými aplikacemi), jako na technický obor, jehož úkolem je vytváření technologií a naopak abstraktní teorie jsou druhořadé, nebo jako na přírodní vědu. Pro třetí možnost argumentuje Denning [6, 7]. Ukazuje, jak informatika splňuje standardní požadavky na vědní disciplínu. Informatika nabízí organizovanou poznatkovou základnu, reprodukovatelné výsledky, využívá dobře popsané metody (vč. metod experimentálních), její teorie jsou vyvratitelné a mají prediktivní sílu (a nejsou triviální), zkoumá umělé, ale i přírodní jevy. Nutno podotknout, že většina informatiky a informatiků se zabývá umělými systémy. Myšlenka na přirozené systémy se zdá zprvu podivná. Denning nicméně uvádí přesvědčivé příklady (částicové jevy, zpracování DNA, lingvistika) a další odkazy.

Jedno z využití Denningových principů je tvorba vzdělávacích programů [8]. Výhodou je srozumitelnost (ve srovnání s kurikulem založeným na jednotlivých technologiích), vyváženější obtížnost, vyváženější poměr teoretických konceptů a

praxe a v neposlední řadě přesnější obraz informatiky na veřejnosti.

### 3. DIDAKTICKÁ TRANSFORMACE

Po prozkoumání výchozího oboru bylo možné započít s jeho transformací do školního předmětu. Zde popíšeme související okolnosti a samotný postup. Školní předměty (resp. vzdělávací obory v RVP) a akademické disciplíny nejsou totožné. Vztahy mezi nimi jsou předmětem zkoumání oborových didaktik [9]. Mezi základní otázky patří výběr vzdělávacích cílů a učiva a jejich zdůvodnění. Proces didaktické transformace lze rozdělit do dvou rovin [10]. Ontodidaktická transformace spočívá v převodu oboru do kurikula, např. ve formě RVP. Psychodidaktická transformace je o úroveň níž, převádí obsah kurikula na skutečnou výuku. Většinou ji tedy realizuje učitel. Školská informatika čelí ve srovnání s dalšími předměty značné neukotvenosti, neexistuje sdílená zkušenost ani názor na běžné a vhodné učivo a jeho uchopení ve výuce. Tento stav lze ale pojímat i jako výhodu, poskytuje totiž příležitost využít zkušeností z ostatních oborů.

Náš výzkum byl koncipován jako design-based research a plány výuky jsme na základě získaných zkušeností opakovaně upravovali. Úpravy se samozřejmě projevovaly na úrovni podrobného plánu, někdy ovšem i na úrovni kurikula. Vždy se nicméně jednalo o dílčí změny. Naopak jsme často hledali, jak dané cíle přeformulovat a dané učivo pojímat jinak, aby je žáci lépe zvládali, a přitom bylo v principu dosaženo původních vzdělávacích cílů. Přes následné úpravy má tedy smysl popsat i prvotní proces transformace.

#### RVP a realita gymnázií

Zcela zásadním hlediskem při vytváření pojetí informatiky pro gymnázia je samozřejmě situace na gymnáziích samotných. Výuka na nich vychází na jedné straně z Rámcového vzdělávacího programu pro gymnázia (RVP, [11]), na druhé straně z léty zažitých praxe. RVP předepisuje 4 hodiny pro vzdělávací oblast nazvanou Informatika a informační a komunikační technologie. Mnohé školy některé z předepsaných hodin tzv. integrují do výuky jiných předmětů. Oblast je zaměřená na práci s digitálními technologiemi a informačními zdroji. Jediné zřetelně informatické téma je algoritmizace, která představuje jednu položku ze 14 okruhů učiva, resp. jeden z 10 očekávaných výstupů. Čteme-li RVP kreativně, různé části informatiky jako oboru můžeme najít. V takovém případě lze ale jen těžko hovořit o nějakém konceptním pojetí.

Jednou z hlavních změn v souvislosti se zaváděním RVP jsou tzv. klíčové kompetence. Jsou v definovány jako „soubor vědomostí, dovedností, schopností, postojů a hodnot, které jsou důležité pro osobní rozvoj jedince, jeho aktivní zapojení do společnosti a budoucí uplatnění v životě.“ V gymnaziálním vzdělávání rozlišujeme kompetenci k řešení problémů, kompetenci komunikativní, kompetenci k učení, kompetenci sociální a personální, kompetenci občanskou a kompetenci k podnikavosti. Jejich bližší popis je uveden v [12]. Především popis klíčové kompetence k řešení problémů naznačuje, že informatika má v gymnaziálním vzdělání své místo, přestože ji obsahová stránka RVP prakticky úplně pominula.

Vzděláváním na čtyřletých gymnáziích a na vyšším stupni víceletých gymnázií se usiluje o naplnění těchto cílů [11, str. 8]:

- vybavit žáky klíčovými kompetencemi na úrovni, kterou předpokládá RVP G;

- vybavit žáky širokým vzdělanostním základem na úrovni, kterou popisuje RVP G;
- připravit žáky k celoživotnímu učení, profesnímu, občanskému i osobnímu uplatnění.

Přes snahu vymezit vzdělávací obor, který RVP v různých ohledech přesahuje, je namístě udržet přiměřenou shodu s těmito cíli a s celkovou koncepcí RVP. Cílem je tedy najít takové pojetí školské informatiky, které jednak umožňuje rozvíjet klíčové kompetence, dále na přiměřené úrovni seznamuje žáky s informatikou jako svébytným oborem a zároveň jako oborem, který zvyšuje možnosti uplatnění žáků (i mimo informatiku jako takovou). Takový přístup odpovídá již tradičním gymnaziálním předmětům, které sledují obdobné cíle.

Analýza RVP dále poslouží k lepší provázanosti informatiky s ostatní výukou na gymnáziu. Nemělo by např. smysl jen kvůli přidání informatiky nějakou látku zdvojovat. Budeme raději hledat to, co je informatice vlastní, a na gymnáziu chybí, jako např. snaha o efektivitu řešení problémů.

Současný stav výuky na gymnáziích lze využít, zaměřit se čistě na informatiku jako obor a předpokládat, že si žáci potřebné uživatelské dovednosti osvojí už dříve v rámci tradiční výuky, která se s informatikou téměř nepřekrývá.

Celý náš projekt samozřejmě směřuje k tomu, aby na gymnáziích výuka informatiky skutečně probíhala. Takový cíl je ovšem v dosti příkrém rozporu se skutečností jako obsah RVP a nedostatek vyučovacích hodin, které by bylo možno informatice věnovat. Snažíme se přitom o důstojné seznámení s oborem, nikoliv několikahodinový úvod. Přestože některé skutečnosti nezbude než ignorovat (informatika není v RVP), někde se kompromisu nevyhneme (rozsah výuky). V opačném případě by sice vznikla představa výuky informatiky ideální, ovšem zcela nerealistická.

Kurikulum jsme proto počítali pro výuku v jednom školním roce po dvě hodiny týdně. To je sice málo ve srovnání např. s přírodovědnými předměty, ale nepoměrně více než nic. Navíc bylo naším záměrem kurikulum konstruovat modulárním způsobem. Tedy tak, aby bylo možné použít byť jen jeho část, pokud podmínky na dané škole víc nedovolí. Při transformaci jsme nehleděli na to, jestli bude výuka třídy probíhat po polovinách, nebo jestli budou obě hodiny v rozvrhu přímo navazovat (být lze takovou organizaci předpokládat).

Mnohá témata informatiky vyžadují nějaké předchozí znalosti, nejčastěji z matematiky. Předpokládali jsme proto výuku v posledních dvou ročnících gymnázia.

## Další aspekty

Cennými zdroji pro transformaci oboru na školní předmět jsou již existující kurikula. Studovali jsme jak starší programy (např. izraelský [13] či slovenský [14]), tak i programy novější, např. ve Spojeném království [15] a USA [16].

Uvedená kurikula nicméně nelze přímo převést do podmínek českých gymnázií. V některém zásadním parametru totiž neodpovídají našim cílům či podmínkám. Některá předpokládají příliš vysokou časovou dotaci (resp. soustavnou výuku v období několika let školní docházky), některá jsou určena jiným věkovým skupinám, další se soustředí především na výuku programování nebo přípravu budoucích studentů informatiky. Cizí kurikula pochopitelně nejsou koncepčně provázána s našimi gymnaziálními předměty.

Nezastupitelný význam při tvorbě informatického kurikula má známý koncept informatického myšlení. Do značné míry totiž odpovídá právě na otázku, co má z informatiky vědět a umět každý člověk. Konkrétních pojetí informatického myšlení existuje větší množství, obvykle vychází z tradice v daném regionu nebo z potřeb místního kurikula a nelze je tedy přímo převzít do našich podmínek. Podrobnější informace a analýzu pojmu informatické myšlení jsme čerpali z článku [17].

Proces didaktické transformace ovlivňuje i obtížnost vznikajícího kurikula. Obtížnost pro žáky zvyšuje už novost předmětu, protože v případě obtíží mají méně možností pomoci než u zavedených předmětů. Informatika má navíc pověst náročného oboru (ovšem v kontextu přípravy na budoucí povolání!). Nakonec poukaz na obtížnost je jedna z častých námitek proti myšlence výuky informatiky ve všeobecném vzdělávání. Chceme-li sestavit kurikulum pro všechny, musíme na otázku obtížnosti brát zřetel.

Lze předpokládat, že hlavním důvodem obtížnosti informatiky bude abstraktnost učiva a nezvyklost řešených problémů. Na rozdíl od nezvyklosti, abstraktnost může být (a v několika případech byla) důvodem k nezařazení nějaké oblasti informatiky do výsledného kurikula. Zkušenost s výukou abstraktních informatických témat nebyla k dispozici, jako přibližné vodítko nám proto posloužilo intuitivní srovnání s abstraktní (či dokonce související), ale již prověřenou látkou ostatních předmětů. Právě srovnatelnost informatiky s ostatními předměty je nakonec jedním z cílů celého snažení. Další heuristikou pro rozhodování o obtížnosti byl předpoklad, že bude snazší případně později ubírat než přidávat, a ověřené pravidlo, že vysoká očekávání vedou k lepším výsledkům.

Dalším aspektem je otázka šířky a hloubky učiva. V různých předmětech se lze setkat s názory, že výuka nemůže jít do potřebné hloubky (nebo dokonce dosahovat uspokojivé kvality), protože množství předepsaného učiva je příliš vysoké. Čistý štít dává šanci v zájmu kvalitního osvojení nezahrnout obsahu příliš mnoho. Snažili jsme se proto omezit na skutečně nejdůležitější základy oboru.

## Teoretický rámec didaktické transformace

Teoretická informatika je disciplína velmi bohatá (zvláště s ohledem na hodinovou dotaci na střední škole). Lze ale najít pojmy, které se v různých oblastech informatiky znovu a znovu vracejí, což ukazuje na jejich důležitost. Mohli bychom je přirovnat k množině či indukci v matematice, popř. zákonům zachování ve fyzice. Tyto pojmy mají zásadní význam pro danou vědu, ale také význam, který ji dalece přesahuje. Toto pozorování rozpracoval psycholog Jerome Bruner do podoby *fundamentálních idejí* [18]. Zejména pro oblast výuky informatiky koncept dále rozvinul Andreas Schwill [19].

Fundamentální idea má splňovat několik kritérií. První je tzv. horizontální kritérium, idea má být rozeznatelná na rozličných místech v rámci dané oblasti. Příkladem je stromová struktura (se stromy se setkáme v souborových systémech a různých hierarchiích, v datových strukturách, v modelování alternativních průběhů výpočtů v teorii vyčíslitelnosti, v XML souborech...).

Vertikální kritérium vyžaduje možnost vyučovat ideu na nejrůznějších úrovních složitosti. Algoritmus jako posloupnost kroků je přístupný i nejmladším žákům, vyšší úroveň pak umožňuje zkoumání a porovnání abstraktních výpočetních modelů a hledání obecné definice algoritmu. Pojem NP-úplnosti se naproti tomu mladším žákům zpřístupňuje jen obtížně.

Fundamentální idea musí být rozeznatelná a významná v průběhu delšího historického vývoje dané disciplíny. Toto kritérium má sloužit k rozlišení dlouhodobě významných idejí. To je zvláště citlivá otázka v informatice, která je mezi ostatními poměrně mladou disciplínou. Nadto se velmi rychle vyvíjí, jak poukazují M. Černochová a J. Vaníček [20]. Časem prověřená je např. dvojková soustava, naopak unáhlit bychom se mohli v otázce hesel „big data“ nebo „augmented reality“.

Poslední kritérium vyžaduje, aby idea souvisela se všednodenním jazykem a uvažováním. Příkladem může být každodenně použitelné modelování závislých úkolů pomocí orientovaného grafu. Naopak dvojková soustava sice stojí v základu informatiky, ale (přímo sama o sobě) v běžném životě příliš nepomůže.

Koncept fundamentálních idejí je v různých variantách nadále využíván k rozvoji výuky informatiky. A. Schwill identifikoval jako fundamentální ideje informatiky algoritmizaci, jazyk a strukturovaný rozklad [19]. Z těchto tří potom odvozuje další. Z konceptu fundamentálních idejí ve své experimentální výuce informatiky vychází také Arno Pasternak [21]. V jeho *braided teaching* odpovídá fundamentální idejí pojem *strand* (protože se v průběhu výuky cyklicky vrací, v souladu s Brunerovým *spirálovým kurikulem*), na který jsou ovšem kladeny poněkud jiné požadavky. Hlavní rozdíl spočívá v požadavku, aby *strand* dokázali bez potíží rozpoznat samotní žáci. Kurikulum konstruované tímto alternativním způsobem více reflektuje pohled žáků a jejich vnímání oboru.

V naší práci jsme se rozhodli strukturovat cíle a učivo pomocí fundamentálních idejí. S ohledem na výše popsané aspekty jsme volně sledovali postup publikovaný v [19]. První krok spočívá v analýze obsahu výchozího oboru, jak jsme popsali v předchozí části. Vyhledávání vztahů a podobností mezi různými oblastmi oboru odhaluje první kandidáty na fundamentální ideje. Dále se využijí ostatní kritéria a kandidátské ideje jsou upravovány či vylučovány podle toho, jestli jsou přístupné na různých intelektuálních úrovních, dostatečně historicky podložené a prověřené a relevantní v běžném životě. V závěru práce je zkoumán vzájemný vztah mezi jednotlivými nalezenými idejemi: Jsou na obdobné úrovni abstrakce? Lze je nějak seskupit, uspořádat či jinak strukturovat?

Výsledek se stal základem našeho vymezení školní informatiky. Jednotlivé ideje jsme dále rozkládali, popisovali a přiřazovali k nim související vzdělávací cíle a učivo. Při tom se ještě výrazněji projeví výše zmíněné aspekty, jako předpokládaný rozsah výuky a požadavek srovnatelnosti s dalšími předměty. V této fázi jsme se opírali o monografii [22].

## 4. INFORMATIKA JAKO ŠKOLNÍ PŘEDMĚT

Nakonec jsme identifikovali čtyři fundamentální ideje, na kterých jsme školní informatiku postavili. Jedná se o informaci, model, algoritmus a efektivitu. Všechny tyto pojmy lze mj. rozpoznat v rámci obecného postupu řešení problému: získáme potřebné informace, vytvoříme vhodný model problému a následně hledáme efektivní, tedy pokud možno algoritmické řešení. Jednotlivé pojmy nyní popíšeme podrobněji.

### Informace

Informace jsou materiálem, který v informatice zpracováváme. Zároveň jde o ústřední pojem na úrovni hmotnosti a energie

ve fyzice. Pro žáky jsou relevantní zejména následující dva pohledy na pojem informace. Prvním je informace jako úbytek možností (něco nevím, musím tedy uvažovat několik možností; získáním informace se něco dozvím a některé možnosti vyloučím). Zobecněním tohoto přístupu je klasická informace jako úbytek entropie, což je ovšem pro běžného žáka gymnázia těžko dostupný pojem.

Informace jako úbytek možností je oproti úbytku entropie jednoduchý koncept. Je ovšem dostatečně mocný např. pro odvození algoritmu půlení intervalu. Úzce souvisí s dvojkovou soustavou a tím i druhým významným pojetím informace, totiž dat spolu s jejich interpretací. To vede mj. na problematiku kódování a jazyků.

Žáci by se měli naučit porovnávat informační obsah zpráv a pokládat efektivní otázky, posuzovat efektivitu různých způsobů kódování a provádět základní související výpočty (např. kolik ještě telefonů můžeme připojit do sítě, než začneme přidělovat desetimístná telefonní čísla). Porozumí problematice komprese a skutečnosti, že bezeztrátová komprese má své limity.

Na poněkud abstraktnější úrovni je poznatek, že každou informaci lze zakódovat pomocí posloupnosti znaků a že v případě počítačů k tomu využíváme dvojkovou soustavu.

### Model

Žáci samozřejmě mentální modely běžně využívají v ostatních předmětech. Nikde se ale modelům nevěnují explicitně. Pro informatiku je ovšem otázka modelování a jeho kvality stěžejní.

Žáci si rozšíří paletu modelů, které mohou v případě potřeby použít. Především se jedná o různé druhy grafů. Ty sice ve škole i v životě běžně používají (větný rozbor, taxonomie v biologii, vazby v molekulách), ale jen na intuitivní úrovni a neuvědomují si jejich souvislost.

Kromě samotných grafů a jejich typických aplikací se žáci seznámí se způsoby jejich kódování pomocí znaků (tedy nikoliv nakreslení).

Žáci postupně zjistí, že na nakreslení grafů nezáleží, tedy že strukturálně stejný graf (a model téže skutečnosti) může vypadat různě. To vede k jednomu z abstraktnějších témat, k isomorfismu grafů. I v dalších oblastech informatiky se opakovaně vynořuje otázka, jestli dvě struktury nejsou z nějakého hlediska totožné, jen např. jinak označené. V matematice žáci řeší rovnost čísel a geometrickou shodnost. Informatika umožňuje pojem rovnosti dále rozpracovat, např. otázkou na „rovnost“ dvou grafů či algoritmů apod.

Uvedené dovednosti zužitkují při práci s dalším užitečným modelem, kterým je stavový prostor. Žáci jsou sice v různých předmětech vedeni k systematickému řešení problémů, nemají k tomu ale žádný obecný nástroj. Jedním z takových nástrojů je být právě stavový prostor, který je možné systematicky procházet a hledat řešení spolehlivě, bez ohledu na talent či štěstí.

Žáci by se měli naučit rozlišovat různé úrovně abstrakce ve stejné situaci a to, jak jsou vhodné. S tím souvisí vědomí kladů i záporů modelování. K zodpovědnému použití každého nástroje patří znalost jeho limitů. V případě stavových prostorů se jedná především o jejich velikost.

### Algoritmus

Algoritmus je ve výuce často vnímán jako „to, co se má naprogramovat“. Takové pojetí ale dostatečně neodpovídá



potřebám gymnazisty. Souvislost s programováním samozřejmě nelze pominout. Algoritmus je ale především spolehlivý postup, který lze realizovat bez dalšího přemýšlení. V principu nezáleží na tom, jestli ho provádí stroj, nebo člověk. Lze předpokládat, že většina gymnazistů bude pracovat právě s postupy „pro lidi“. I ty ovšem často musí být spolehlivé. Typickým příkladem, kde se v principu pokoušíme splnit požadavky na algoritmy, jsou oblast medicíny a práva.

Žáci se seznámí s různými způsoby zápisu algoritmů. V životě se nejčastěji setkají s postupy popsány přírodním jazykem, jimž proto věnujeme zvláštní pozornost. Dále pracujeme s vývojovými diagramy (a rozhodovacími stromy jako jejich předstupněm).

Žáci by se měli setkat i se základy programování. Cílem je nicméně především podpora porozumění informatickým principům, nikoliv tvorba softwaru. Žáci se stručně seznámí se základními řídicími strukturami a s tím, jak jim programování může ušetřit práci. Pro většinu z nich je nakonec hlavním výsledkem právě schopnost rozpoznat rutinní úlohu, kterou nemusí řešit člověk, a zjištění, že mohou leckdy sami drobnou úpravou změnit chování různých programů.

Konkrétní algoritmy se žáci běžně učí a používají je v ostatních předmětech. Obvykle ovšem nedojdou až k jejich přesnému zápisu. Stejně tak nerozlišují, jestli jsou předkládané postupy algoritmické (typicky v matematice, fyzice) nebo nikoliv (např. v jazycích).

Žáci by měli umět (na přiměřené úrovni obtížnosti) rozpoznat, jestli je daný postup algoritmem a případně ho vhodně upravit, aby se algoritmem stal. Měli by umět algoritmus opravit nebo přizpůsobit algoritmus jinému účelu.

Žáci by měli umět algoritmická řešení i vytvářet. K tomu by měli být vybaveni řadou příkladů, kterými se mohou inspirovat a z nichž mohou vycházet. Mohou se naučit algoritmy jako vyhledávání půlením intervalu, řazení nebo systematické procházení grafu (stavového prostoru). Přitom nečekáme, že je v praxi použijí přímo a beze změny z paměti. Měli si ale osvojit a použít příslušné základní principy (např. „rozděl a panuj“).

Pod algoritmus řadíme i rekurze jako elegantní způsob řešení některých problémů. Těžko očekávat, že budou žáci sami rekurzivní řešení objevovat. Měli by si s nimi ale poradit, rekurze (ať už algoritmická, nebo strukturální) by je ale neměla zmást.

Tak jako v ostatních případech, i u algoritmů by žáci měli vědět, že mají své limity a v základních případech usoudit, kdy je nutno vyzkoušet jiný přístup k řešení. Jedná se jak o problémy s neurčitým zadáním, tak o problémy neřešitelné v principu, protože řešení vůbec neexistuje, nebo protože nelze algoritmicky najít.

## Efektivita

Otázka efektivity se prolíná takřka celou informatikou. Tradiční součástí (vysokoškolských) úvodních kurzů je problematika asymptotické složitosti. To je pro gymnazisty příliš formální, mohou se nicméně naučit řádově odhadovat prostředky potřebné pro vykonání toho kterého postupu v příznivém, průměrném i nepříznivém případě. Tím pádem by měli umět dva algoritmy porovnat a zvolit ten vhodnější. Zároveň by měli umět pracovat i s dalšími kritérii, nikoliv jen s v teorii oblíbeným počtem základních operací jako zástupcem měření času.

Efektivita staví další limit použitelnosti algoritmických řešení. Pojem NP-úplnosti je pro gymnázium příliš abstraktní a málo

potřebný. I gymnazista by měl nicméně vědět, že exponenciální algoritmus nejspíš použít nechce.

Pod efektivitu řadíme i rozličné heuristiky k jejímu zvyšování, tedy optimalizaci. Jako příklad uveďme předtřídění dat nebo okamžité ukončení práce ve chvíli, kdy máme dostatečný výsledek. Sem zapadá i otázka paralelizace. Žáci se seznámí s modelováním závislostí navazujících úkolů pomocí orientovaného grafu a se základními možnostmi použití, vč. zjišťování mezní doby splnění všech úkolů a počtu souběžně prováděných úkolů.

Už v souvislosti s algoritmem žáci zjišťují, na jaký typ úkolů se algoritmické postupy hodí, a na jaké se naopak hodí člověk. V konečném důsledku se jedná o otázku efektivní alokace zdrojů. Opět se vynořuje otázka pojetí rovnosti: pokud člověk i stroj odvedou stejně kvalitní práci, bude stroj nejspíš levnější. Žáci se proto musí seznámit také se stavem a potenciálem vývoje v oblasti umělé inteligence.

## Program výuky

Ve výuce není nutně nejvhodnější postupovat po jednotlivých idejích. Někteří učitelé by např. pojem algoritmus stavěli co možná nejvíc na začátek výuky, právě s poukazem na fundamentálnost pojmu algoritmu pro informatiku. Nám se nicméně osvědčilo posečkat, až se žáci přirozeně seznámí s několika příklady. Díky nim si pak mohou definiční vlastnosti algoritmů představit konkrétně, lépe jim tak porozumět, a lépe porozumět také tomu, proč jsou právě ony tak významné.

Jednotlivé fundamentální ideje se navíc překrývají v tom smyslu, že na totéž učivo lze pohlížet různě a zdůraznit jeho jinou stránku. Například rozhodovací stromy jsou *modelem* rozhodování, tedy nějakého *algoritmu*, který vede k získání *informace*. Snažíme se přitom, aby stromy nebyly zbytečně hluboké, tedy aby rozhodování probíhalo *efektivně*. S podobnou situací se setkáme u grafových notací. Představují způsob zakódování nějaké *informace*, která sama něco *modeluje*. Přitom bychom rádi, aby způsob práce s tou kterou notací byl *algoritmický* a abychom vybrali notaci v nějakém smyslu *efektivní* – např. jednoduše použitelnou, nebo datově úspornou.

Program výuky jsme tedy strukturovali do jednotlivých volně navazujících modulů, každý věnovaný jednomu hlavnímu tématu. Moduly ovšem nejsou izolované, základní pojmy se opakovaně vynořují a prolínají se všemi moduly, podobně jako v programu [21]. Pojmenování modulů je tak spíš pro orientaci a pro zdůraznění důležitosti jednotlivých témat z pohledů žáků. Jeden modul je plánován přibližně na jeden měsíc výuky.

Podrobné informace lze získat v on-line učebnici, kterou na základě získaných zkušeností tvoříme [23].

## 5. PILOTÁŽ

Pro úplnost článku je na místě doplnit, že uvedené výsledky nejsou čistě teoretické. Navržené pojetí informatiky jsme nejprve několik let testovali a upravovali v rámci různých volitelných seminářů, abychom jej nakonec jeden školní rok pilotovali ve výuce s celou třídou žáků.

Dostali jsme k dispozici kvintu osmiletého gymnázia, ve které jsme mohli vyučovat povinnou informatiku podle námi vyvinutého pojetí. Žáci měli jednu hodinu týdně, výuka probíhala po polovinách. Ze 32 žáků bylo 17 děvčat.

Oproti plánu se jednalo o mladší žáky a menší rozsah výuky, což pochopitelně způsobilo jisté komplikace. Na druhé straně takové nastavení umožnilo ještě přísněji určit skutečně základní vzdělávací cíle a formulovat je tak, aby nezávisely na úrovni znalostí z ostatních předmětů. Ukázalo se, že je to možné, a nevede to nutně k trivializaci látky.

Výuku jsme hodnotili na základě různých souborů dat, zahrnujících naše vlastní poznámky, práce žáků, data ze školního informačního systému a výsledky pololetních a závěrečných dotazníků, které žáci vyplňovali. Závěry z pololetních dotazníků jsme publikovali v [24], celkové závěry k publikaci připravujeme.

Sami žáci deklarovali přínos výuky ve schopnosti efektivního řešení problémů i efektivního chování obecně, v abstraktním (či logickém) myšlení a ve schopnosti jednoznačného a srozumitelného vyjádření. Tyto výsledky potvrzuje i srovnání úrovně žákovských prací na začátku a na konci školního roku. Lze tedy tvrdit, že jsme výukou pozitivně ovlivnili klíčové kompetence žáků.

Poněkud překvapivě uváděli žáci i zlepšení v oblasti uživatelských dovedností. Na to jsme se ve výuce přímo nezaměřovali. Postupovali jsme nicméně tak, jak to očekáváme i v jiných předmětech: když bylo pro výuku něco potřeba, naučili jsme se to. Důležitým faktorem je patrně také to, že množství vzdělávacích aktivit probíhalo na počítačích.

V parametrech jako zajímavost, užitečnost a oblíbenost (vše z pohledu žáků) se informatika bez problémů řadila mezi přírodní vědy a matematiku. Jednoznačně nejvýše v porovnání s ostatními předměty žáci hodnotili obtížnost pochopení informatiky. Tento výsledek byl kromě obtížnosti samotné látky nutně ovlivněn i věkem žáků, existencí domácích úkolů a jejich nezvyklou povahou (např. nebylo vždy hned zřejmé, jak postupovat) a použitím systému Moodle. Zajímavé je, že se obtížnost rozhodujícím způsobem neprojevila např. na oblíbenosti.

Další oblast zkoumání se týkala pojetí informatiky jako takové. Zatímco na začátku roku viděli žáci předmět téměř výlučně jako ovládání aplikací, po roce výuky se v jejich vlastních definicích objevilo 22krát slovo věda, 22krát slovo technologie a opakovaně se vyskytovaly vazby jako „co je uvnitř“ či „jak to funguje“. Představa těchto žáků o informatice se tedy zřetelně přiblížila skutečnosti.

## 6. ZÁVĚR

Na uvedené výsledky je samozřejmě třeba pohlížet jako na výsledky v dané třídě, v daném roce a na dané škole. Postačí ale jako ukázka toho, že výuka vědní informatiky podle výsledného programu je možná a může být pro žáky prospěšná. Zároveň je nutno dodat, že rozsah testování a pilotáže stěží mohl vést k optimálnímu nastavení vzdělávacích cílů a metod. Je tedy nadále na čem pracovat a co zlepšovat.

## 7. BIBLIOGRAFICKÉ ODKAZY

- [1] VANÍČEK, J., M. PAPÍK, R. PERGL and T. VANÍČEK. *Teoretické základy informatiky*. Praha: Kernberg Publishing, 2007.
- [2] AHO, AV and JD ULLMAN. *Foundations of computer science*. New York: Computer Science Press, 2000.
- [3] HAREL, David and Yishai FELDMAN. *Algorithmics: The spirit of Computing*. Addison-Wesley: Addison-Wesley, 2004.
- [4] GUO, Philip J. *What is Computer Science? Efficiently Implementing Automated Abstractions* [online]. 2010. Dostupné z: <http://pgbovine.net/what-is-computer-science.htm>
- [5] WING, Jeannette M. *Computational thinking*. B.m.: Carnegie Mellon University. 2011.
- [6] DENNING, Peter J. Computing is a natural science. *Communications of the ACM*. 2007, vol. 50, no. 7, pp. 13–18.
- [7] DENNING, Peter J. Is computer science science? *Communications of the ACM*. 2005, vol. 48, no. 4, pp. 27–31.
- [8] DENNING, Peter J. Great Principles in Computing Curricula. In: *SIGCSE '04 Proceedings of the 35th SIGCSE technical symposium on Computer science education*. ACM New York, NY, USA, 2004, p. 336–341.
- [9] DENG, Z. Transforming the subject matter of a secondary-school science subject. *Journal of Curriculum Studies*. 2007, vol. 39, no. 5, pp. 503–535.
- [10] JANÍK, Tomáš and Jan SLAVÍK. Vztah obor - vyučovací předmět jako metodologický problém. *Orbis Scholae*. 2007, vol. 1, no. 1, pp. 54–66.
- [11] *Rámcový vzdělávací program pro gymnázia*. Praha: Výzkumný ústav pedagogický v Praze, 2007. ISBN 978-80-87000-11-3.
- [12] ŠLEJŠKOVÁ, E. *Klíčové kompetence na gymnáziu*. Praha: Výzkumný ústav pedagogický v Praze, 2008. ISBN 9788087000205.
- [13] GAL-EZER, Judith, Catriel BEERI, David HAREL and Amiram YEHUDAI. A High-School Program in Computer Science. *Computer*. 1995, vol. 28, no. 10, pp. 73–80.
- [14] *Štátny vzdelávací program pre gymnázia v Slovenskej republike ISCED 3A – Vyššie sekundárne vzdelávanie*. Bratislava: Štátny pedagogický ústav, 2008.
- [15] COMPUTING AT SCHOOL WORKING GROUP. *Computer Science: A curriculum for schools*. Computing at School, 2012.
- [16] TUCKER, Allen B., Deborah SEEHORN, Stephen CAREY, Daniel MOIX, Brian FUSCHETTO, Irene LEE, Dianne O'GRADY-CUNIFF, Chris STEPHENSON and Anita VERNON. *CSTA K-12 Computer Science Standards*. 2011.
- [17] LESSNER, Daniel. Analýza významu pojmu „Computational Thinking“. *Journal of Technology and Information Education*. 2014, vol. 6, no. 1, pp. 71–88.
- [18] BRUNER, Jerome Seymour. *The process of education*. B.m.: Harvard University Press, 1977.
- [19] SCHWILL, Andreas. Fundamental Ideas of Computer Science. *Bull. European Assoc. for Theoretical Computer Science*. 1994, vol. 53.
- [20] VANÍČEK, Jiří and Miroslava ČERNOCHOVÁ. Didaktika informatiky na startu. In: STUHLÍKOVÁ and JANÍK, eds. *Oborové didaktiky v České republice: stav, vývoj a perspektivy*. Brno: Masarykova Univerzita, 2015, in press.
- [21] PASTERNAK, Arno and Jan VAHRENHOLD. Design and evaluation of a braided teaching course in sixth grade computer science education. In: *Proceedings of the 43rd ACM technical symposium on Computer Science Education*. New York, NY, USA: ACM, 2012, p. 45–50.
- [22] PASCH, Marvin, Trevor G. GARDNER, Georgea M. LANGER, Alane J. STARK and Christella D. MOODY. *Od vzdělávacího programu k vyučovací hodině*. Praha: Portál, 2005. ISBN 80-7367-054-2.
- [23] LESSNER, Daniel. *Informatika pro každého* [online]. 2015. Dostupné z: <http://ksvi.mff.cuni.cz/ucebnice>
- [24] LESSNER, Daniel. Jak žáci gymnázia vnímají výuku informatiky jako vědy. In: *Didinfo 2013 - Zborník príspevkov*. Banská Bystrica: Univerzita Mateja Bela, 2013.



# Hlasovacie systémy – interaktívna technológia na podporu aktívneho učenia sa

Gabriela Lovászová  
Katedra informatiky FPV UKF v Nitre  
Tr. A. Hlinku 1  
949 74 Nitra  
Slovensko  
glovaszova@ukf.sk

Viera Michalíková  
Katedra informatiky FPV UKF v Nitre  
Tr. A. Hlinku 1  
949 74 Nitra  
Slovensko  
vmichalickova@ukf.sk

## ABSTRAKT

V príspevku sa zaoberáme hlasovacími systémami ako modernou interaktívnou digitálnou technológiou vhodnou na podporu aktívneho a kolaboratívneho učenia sa. Na príklade kvalitného webového systému Socrative ilustrujeme viacúčelovosť, efektívnosť a flexibilitu nástrojov tohto typu. Automatizované zbieranie odpovedí od všetkých prítomných žiakov a ich spracovanie v reálnom čase umožňuje získať okamžitú spätnú väzbu cennú pri formatívnom hodnotení. Uvádzame rôzne kategórie hlasovacích aktivít ako aj konkrétne námety realizovateľné vo vyučovaní informatiky. Sumarizujeme výhody dobre navrhnutých učebných aktivít založených najmä na kladení otázok overujúcich úroveň porozumenia, motivujúcich žiakov k produktívnej diskusii, tvorivej činnosti a uvažovaní o vlastnom učení sa.

## ABSTRACT

In this paper, we deal with student response systems (SRS) – the modern interactive digital technology designed to enhance active and collaborative learning. To illustrate the versatility, efficiency and flexibility of such tools, we give an example of a high-quality web-based SRS called Socrative. The automatic recording and real-time processing of responses provided by every single student enable teacher to use the immediate feedback for purposes of the formative assessment. We describe various categories of SRS activities including concrete suggestions for teaching informatics. The benefits of the well-designed learning activities are summarized. We recommend mainly the activities based on questions verifying the level of understanding or having the potential to involve students in productive discussion, make them act creatively or reflect on their own learning.

## Kľúčové slová

Hlasovací systém, Socrative, aktívne učenie sa, kolaboratívne učenie sa.

## Keywords

Student response system, Socrative, active learning, collaborative learning.

## 1. ÚVOD

Hlasovacie systémy sú elektronické systémy na zhromažďovanie a automatické spracovanie odpovedí publika v reálnom čase. V anglofónnej odbornej literatúre sú označované termínmi: *Audience response system* (ARS) ako všeobecný termín pre systémy na získavanie a okamžité spracovanie odpovedí z (väčšinou) veľkého publika, *Classroom response system* (CRS)

alebo *Student response system* (SRS) pre systémy používané špeciálne vo vzdelávaní.

Hlasovacie systémy vo vzdelávaní sú technologickou obdobou hlasovania žiaka zdvihnutím ruky alebo hlásenia sa rukou k slovnej odpovedi na otázku učiteľa. Oproti hlasovaniu resp. hláseniu sa rukou má elektronický systém viacero výhod:

- možnosť anonymnej odpovede,
- hromadnosť odpovedí (odpovedajú všetci žiaci, nielen vyzvaný žiak),
- okamžité vyhodnotenie a zobrazenie výsledkov,
- uchovávanie dát o odpovediach žiakov na neskoršiu analýzu.

Prvotným cieľom používania hlasovacích systémov v škole (v období 70. rokov minulého storočia) bola automatizácia hodnotenia výkonu žiaka, tzv. *strojové skúšanie* [1]. Žiaci odpovedali na sériu testových otázok s možnosťou výberu odpovede pripravenú na hárku papiera a kódy otázok a odpovedí zadávali prostredníctvom hlasovacieho zariadenia pripojeného k centrálnej jednotke, ktorá odpovede zaznamenávala a okamžite vyhodnocovala. Testy pripravovali a hlasovacie zariadenia programovali učitelia, alebo sa používali profesionálne didaktické testy dodávané na školy spolu s hlasovacím zariadením.

Moderné hlasovacie systémy reagujú na potreby súčasnej generácie žiakov a celkovú zmenu vzdelávacej paradigmy. Od strojov na skúšanie sa posunuli k modernej interaktívnej digitálnej technológii, ktorá podporuje aktívne a kolaboratívne učenie sa. V takomto zmysle ich prezentujeme aj v nasledujúcich častiach článku.

V technických riešeniach moderných hlasovacích systémov sa využívajú buď špeciálne hlasovacie zariadenia – vysielace s tlačidlami na zadávanie odpovedí (niekedy aj s displejom), ktoré komunikujú bezdrôtovou technológiou na infračervenej alebo rádiovkej frekvencii s prijímačom, alebo sa použije existujúca infraštruktúra wifi na bezdrôtovú komunikáciu v počítačovej sieti a ako hlasovacie zariadenia slúžia vlastné mobilné zariadenia odpovedajúcich (smartfóny, tablety, notebooky). Prehľad typov hlasovacích systémov a ich vlastností z pohľadu technickej realizácie uvádzame v Tabuľke 1.

Tabuľka 1. Typy hlasovacích systémov

typ komunikácie	výhody	nevýhody
infračervený (IR) signál	signál neprestupuje cez steny, nižšia cena	jednosmerná komunikácia, nutný priamy výhľad vysielateľa na prijímač

rádiový signál	väčší dosah signálu, dvojsmerná komunikácia, nie je nutný priamy výhľad na prijímač	vyššia cena, možné interferencie signálu
wifi	variabilita zariadení, využitie existujúcej wifi infraštruktúry, webové rozhranie, variabilita aktivít	vlastné mobilné zariadenia na hlasovanie

V období, keď smartfón, tablet alebo notebook vlastní čoraz viac používateľov, je veľmi perspektívny koncept používania vlastných mobilných digitálnych zariadení pri práci alebo vzdelávaní Bring Your Own Device (BYOD). Na tomto koncepte je založený aj hlasovací systém Socrative. Využíva sa existujúca komunikačná infraštruktúra a štandardné mobilné zariadenia na odosielanie odpovedí bez nutnosti investície do špeciálneho hardvéru.

## 2. SYSTÉM SOCRATIVE

*Socrative* [2] je webová aplikácia implementujúca funkcionality hlasovacieho systému moderným a praktickým spôsobom (multiplatformovosť implicitne podporujúca politiku BYOD, intuitívne rozhranie, dôraz na edukačné ciele, sociálnosť, bezplatnosť). Jej serverová časť poskytuje zaregistrovanému učiteľovi vlastné úložisko vo forme virtuálnej triedy s jedinečným identifikátorom. Používateľ v roli študenta sa neregistruje, do virtuálnej triedy svojho učiteľa sa prihlási až v čase realizovania učebnej aktivity prostredníctvom osobného počítača alebo mobilného zariadenia s prístupom na internet. Pre tablety či smartfóny s operačnými systémami Android a iOS možno ako alternatívu k webovému prehliadaču používať mobilné aplikácie. K dispozícii je prehľadný používateľský návod, videotutoriály ako aj portál komunity používateľov [3] s odporúčaniami a konkrétnymi námetmi na efektívne učebné aktivity overené v praxi.

Učiteľ môže využívať Socrative rôznym spôsobom. Základné použitie bez nutnosti editovania a ukladania otázok vopred predstavuje voľba *Quick Question*, t. j. rýchla otázka, ktorú učiteľ položí študentom osobne v priebehu vyučovacej hodiny (otázka s výberom z viacerých možností, áno/nie alebo krátka odpoveď). Vopred pripravený kvíz (test) pozostáva z viacerých otázok (s možnosťou pripojiť obrázok a označiť správnu odpoveď spolu

s uvedením jej vysvetlenia).

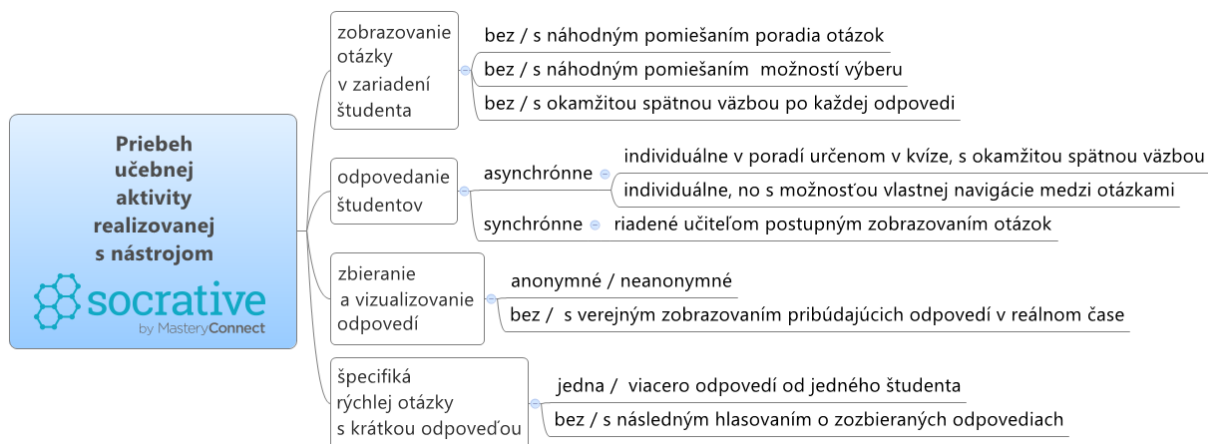
Kvízy je možné tagovať a sprístupňovať na zdieľanie (priradením a zverejnením ich sociálneho čísla). Kvíz importovaný zadáním sociálneho čísla alebo zo šablóny vo formáte *xlsx* si učiteľ ľahko upraví priamo vo webovom rozhraní. Ktorýkoľvek z kvízov v úložisku sa dá realizovať formou súťaže jednotlivcov alebo tímov spustením v tzv. *Space Race* móde so zábavným efektom zobrazovania pokroku súťažiacich vo forme letiacej rakety. Špeciálnym typom kvízu je tzv. *Exit Ticket* s tromi otázkami zameranými na zhodnotenie priebehu práve ukončeného vyučovania a výsledkov vlastného učenia sa (zabudovanú anglickú verziu rovnocenne nahradí vlastný kvíz s analogickým obsahom).

Priebeh kvízov, zbieranie odpovedí a ich okamžité vizualizovanie učiteľ ovplyvňuje nastavovaním viacerých voliteľných parametrov (Obrázok 1).

Reporty generované automaticky po každej z ukončených aktivít sa dajú zobraziť ihneď, ale tiež odoslať mailom alebo uložiť v cloude *Google Drive* (vo formáte *pdf* alebo *xlsx*) a analyzovať ich neskôr (tabuľku s výsledkami celej triedy, jednotlivca alebo súhrn odpovedí osobitne pre každú z otázok).

Socrative sa možnosťou vytvárať kvízy podobá iným softvérom špecializovaným na elektronické testovanie vedomostí. Nejde však o nástroj určený primárne na tvorbu testov, bol vyvíjaný predovšetkým ako nástroj na podporu motivácie, aktívneho učenia sa a pre potreby formatívneho hodnotenia. V článku [4] autori prezentujú použitie systému Socrative vo vysokoškolskom kurze fyziky pre budúcich učiteľov. Krátky test zaradený po výkladovej časti prednášky realizovali so zámerom získať spätnú väzbu o úrovni porozumenia jednotlivých študentov. Informáciu o rozdieloch v odpovediach študentov využili v ďalšom priebehu vyučovacej hodiny na zostavenie heterogénnych skupín s cieľom dosiahnuť produktívnejšiu diskusiu o správnych odpovediach a podporiť tak kolaboratívne učenie sa. Lepší výsledok opakovaného testu po diskusii v skupinách potvrdil efektívnosť zvoleného prístupu. Rôzni autori prezentujúci svoju skúsenosť s používaním hlasovacieho systému Socrative, napr. [4, 5, 6], hovoria o rovnakých pozitívach inovácie svojho metodického postupu:

- prednáška pre väčšie skupiny študentov vedená obvykle výkladovou metódou sa môže zmeniť na interaktívnu učebnú aktivitu prirodzene motivujúcu študentov



Obrázok 1. Voliteľné parametre pri používaní nástroja Socrative

k činnosti, neformálnej diskusii ako aj uvažovaní o vlastnom učení sa;

- použitie modernej technológie blízkej súčasným študentom zatriktívňuje vyučovanie, poskytuje učiteľovi efektívnym spôsobom okamžitú spätnú väzbu o napredovaní jednotlivcov, prípadných miskoncepciách a pod.;
- kľúčom k úspechu je kvalita položených otázok, medzi ktorými by mali prevažovať otvorené otázky rozvíjajúce vyššie myšlienkové procesy, spoluprácu a tvorivosť.

V nasledujúcej kapitole uvádzame vlastné príklady rôznych učebných aktivít realizovateľných pomocou nástroja typu Socrative a poukazujeme na ich prínos pre vyučovací proces.

### 3. STRATÉGIE POUŽÍVANIA HLASOVACIEHO SYSTÉMU V ŠKOLE

Implementácia technológie získavania a spracovania odpovedí prostredníctvom elektronického hlasovacieho systému do vyučovania závisí predovšetkým od cieľa, ktorý sledujeme položením otázky, tiež od publika, ktorému otázku kladieme, a od situácie a prostredia, v ktorých otázku kladieme. Cieľom hlasovania môže byť napríklad: aktívne zapojenie poslucháčov do prednášky vo veľkej prednáškovej miestnosti kladením otázok

prednášateľovi alebo zodpovedaním rýchlej kontrolnej otázky, testovanie vedomostí žiakov formou didaktického testu pre potreby formatívneho alebo sumatívneho hodnotenia, rozvoj kritického myslenia formou vzájomného hodnotenia prác žiakov, podpora tvorivého myslenia generovaním nápadov (brainstorming) a podobne. Hlasovanie môže prebiehať individuálne (žiak odpovedá samostatne), alebo ako skupinová aktivita (odpovedá skupina ako celok po diskusii o položenej otázke). Úprimnosť odpovedí a ochotu odpovedať môže zvýšiť anonymné hlasovanie. Neanonymné odpovede sú užitočné na monitorovanie triedy (dochádzka, výkony) hlavne pri dodatočnom spracovaní a analýze zhromaždených údajov. Použitie hlasovacieho systému môže byť vopred podrobne pripravené a naplánované v scenári výučby, alebo realizované neplánovane podľa situácie v triede. Výsledky hlasovania môžu ďalší priebeh výučby ovplyvniť rozhodujúco alebo mať zásadnejší vplyv na plánovaný scenár výučby. Hlasovací systém môže slúžiť len na prenos a zhromažďovanie dát a zbierané odpovede hodnotí učiteľ alebo žiaci navzájom, alebo môže plniť aktívnejšiu úlohu pri komunikácii so žiakom (automaticky hodnotiť odpovede, poskytovať automatickú spätnú väzbu).

V Tabuľke 2 sme sformulovali päť rôznych kategórií aktivít s využitím hlasovacieho systému. Ďalej uvedieme príklady aktivít z jednotlivých kategórií vo vyučovaní informatiky.

Tabuľka 2. Typy aktivít s hlasovacím systémom

Typ aktivity	Cieľ	Vlastnosti	Prínos pre žiaka	Príklady
Test	automatizovať hodnotenie výkonu žiaka	väčšinou asynchrónne a neanonymné odpovede, automatické vyhodnotenie a spätná väzba, nie sú potrebné podstatné zmeny v koncepcii a scenári výučby	zvýšenie vonkajšej motivácie učiť sa	didaktický test na preverenie vedomostí, vedomostná súťaž
Kontrolná otázka	zistiť úroveň porozumenia	synchronné, anonymné alebo neanonymné odpovede, projekcia odpovedí, okamžitá spätná väzba formou diskusie, priebeh výučby reaguje na zistenia z odpovedí žiakov	aktivizácia myslenia, odhalenie miskoncepcií, identifikácia podstatných prvkov učiva	kontrolná otázka počas výučby s projekciou odpovedí a diskusiou
Zber nápadov	zefektívniť učenie sa spoluprácou	synchronné, anonymné, viacnásobné odpovede, projekcia odpovedí, okamžitá spätná väzba formou diskusie, rozhodujúci vplyv na priebeh vyučovania, vysoký stupeň konštruktivismu	využitie kolektívnej inteligencie pri riešení problémov, podpora tvorivého myslenia	generovanie tvorivých nápadov (brainstorming), zdieľanie skúseností
Hlasovanie	podporiť kritické myslenie, angažovanosť	synchronné, anonymné odpovede, automatická spätná väzba (sumárne vyhodnotenie), malý okamžitý vplyv na priebeh vyučovania	posudzovanie kvality informácií, riešení	hlasovanie za najlepšiu nápad v brainstormingu, vzájomné hodnotenie žiackych prác
Prieskum postojov	zistiť postoje, záujmy žiakov	anonymné, asynchrónne alebo synchronné odpovede, bez alebo s okamžitou spätnou väzbou od učiteľa, neovplyvňuje okamžite scenár výučby	možnosť zaujať a vyjadriť osobný postoj, položiť otázky	vytváranie pozitívnej komunikačnej klímy v úvode ( <i>ice breaking</i> ), zisťovanie názorov v závere ( <i>back channel</i> )

### 3.1 Test

Automatické testovanie vedomostí je historicky najstarším typom aktivít s hlasovacím systémom. Jeho cieľom je zefektívniť meranie výsledkov vzdelávania okamžitým automatickým vyhodnocovaním odpovedí, prípadne aj automatickou spätnou väzbou. Žiak odpovedá na otázku v teste väčšinou neanonymne a individuálnym tempom. Odpovede sa zhromažďujú a automaticky vyhodnocujú.

Z pohľadu žiaka je testovanie vedomostí aktivita, ktorá zvyšuje jeho vonkajšiu motiváciu učiť sa. Jeho cieľom je získanie odmeny vo forme dobrého hodnotenia. Snaha o dosiahnutie dobrého výsledku v testovaní však môže na žiaka pôsobiť stresujúco. Preto moderné testovacie systémy ponúkajú rôzne hravé formy testovania, ktorých cieľom je tento stres odbúrať. V systéme *Socrative* je to testovanie formou tímovej vedomostnej súťaže.

Testovanie si vyžaduje starostlivú prípravu testu so súborom otázok. Zdrojom námetov na kvalitné testové otázky z informatiky môže byť napríklad súťaž *iBobor*.

### 3.2 Kontrolná otázka

V priebehu súvislého monológu učiteľa bez aktívneho zapojenia sa žiakov do vlastného procesu učenia sa nemožno očakávať hlbšie pochopenie podstatných prvkov učiva, súvislostí či zapamätanie si hlavných myšlienok výkladu. Pasívne počúvanie vedie prirodzene k strate pozornosti, a tým aj záujmu o vyučovanie. Zaraďovanie kontrolných otázok je len malou zmenou v scenári vyučovacej hodiny, ktorá však môže mať veľký efekt pre žiaka i učiteľa. Otázky si môže učiteľ premyslieť už v rámci prípravy na vyučovanie, prípadne položiť spontánne v reakcii na prejavy žiakov. V systéme *Socrative* nie je potrebné takúto otázku vopred editovať, používa sa len na rýchle zozbieranie a vyhodnotenie odpovedí. Otázka by mala priamo súvisieť s témou, o ktorej sa práve hovorí, a to tak, aby umožnila žiakovi zopakovať si nové informácie v inom kontexte, všimnúť si podstatné prvky preberaného javu, či aplikovať nový pojem alebo postup v praxi.

Cieľom kladení rýchlych kontrolných otázok na vyučovaní je udržiavať pozornosť žiakov, motivovať ich, aby aktívne premýšľali o tom, čo sa učia. Kontrolné otázky bývajú väčšinou uzavreté, lebo majú učiteľovi poskytnúť informácie o tom, ako žiaci rozumejú preberanému učivu. Sú kontrolnými miestami vo vyučovaní, v ktorých učiteľ môže pozmeniť svoj pôvodný plán, vrátiť sa k obsahu otázky, na ktorú žiaci odpovedali nesprávne, otvoriť diskusiu, zabrániť vzniku nesprávnych predstáv alebo doplniť naivné predstavy a prehĺbiť úroveň porozumenia.

Príklad otázky: *Na čo slúži príkaz znovu v jazyku Imagine Logo?*

Príklad odpovede: *„Tento príkaz slúži na vymazanie všetkého čo sme nakreslili.“*

Táto odpoveď predstavuje dobrý prekoncept (prvotnú predstavu) o význame príkazu, ktorý sa skúsenosťami dopĺňa, upresňuje. Na úvodnej hodine programovania v Logu môže byť takáto odpoveď postačujúca, nie však u pokročilého programátora. Podľa toho, aké skúsenosti s programovaním v Logu žiaci majú, by mali poznať aj niektoré z ďalších efektov príkazu *znovu*, napríklad: obnoví sa domovský stav korytnačky (teda domovská pozícia a smer), zmazanie znamená prekreslenie pozadia aktuálnou farbou pozadia, vyvolá sa vykonanie reakcie na udalosť *prizmazaní atď.*

Po zodpovedaní rýchlej otázky a projekcii odpovedí spravidla nasleduje diskusia. Žiaci môžu konfrontovať svoje odpovede s odpoveďami spolužiakov a upresňovať svoje chápanie pojmu. Úlohou učiteľa je moderovať diskusiu tak, aby jej závery viedli k takému vymedzeniu pojmu, aký zodpovedá skúsenostiam žiakom.

Iný variant diskusie: Žiaci diskutujú o zobrazených odpovediach sami v malých skupinkách bez učiteľovho moderovania a svoje závery formulujú v novej odpovedi na opakovane položenú tú istú otázku. Učiteľ vstupuje do diskusie až po druhom zodpovedaní otázky.

### 3.3 Zber nápadov

Jednou z aktivizujúcich metód vyučovania na podporu tvorivého myslenia žiakov je brainstorming. Podstatu brainstormingu vystihuje jeho názov („búrka v mozgu“). Po položení otázky či sformulovaní problému sa od žiakov očakáva čo najviac spontánnych, rýchlych odpovedí. Žiaci svoje odpovede opierajú o vlastné vedomosti a životné skúsenosti, niektorí môžu navrhnúť veľmi netradičné, neočakávané riešenia. Pomocou systému *Socrative* môžeme zaznamenávanie nápadov anonymizovať a automatizovať. Zozbierané nápady následne v rámci skupinovej diskusie podrobíme kritike s cieľom vybrať optimálne riešenie, príp. vylúčiť príspevky, ktoré nemožno považovať za vhodné odpovede. Učiteľ musí pri riadení takejto aktivity dbať na vytvorenie dobrej atmosféry, zabrániť tendenciám prvoplánovo zosmiešňovať nápady iných.

V učebnici informatiky pre ZŠ nájdeme vhodné námety na otázky do brainstormingu: *Čo všetko robíme s informáciami? Ako sa informácie (textové, grafické, číselné) dostanú do počítača? Čo robíme s informáciami (textom obrázkom, číslami) v počítači? Ako sa dostanú informácie (textové, grafické, číselné) von z počítača?*

Tieto otázky motivujú žiakov rozmyšľať o tom, čím sa zaoberá informatika a ako sa informácie rôzneho typu spracovávajú pomocou počítača.

Rovnakú metódu môžeme použiť nielen na zber tvorivých nápadov, ale aj na zber a zdieľanie skúseností.

Príklad: Žiaci experimentujú s nasledujúcim programom. Hľadajú riešenie úlohy: *Nájdí hodnotu vstupnej premennej a takú, aby výstup bol 5.*

```
procedure TForm1.Button1Click(Sender:
TObject);
var
  a, s: Integer;
begin
  a := StrToInt(Edit1.Text);
  s := 0;
  repeat
    s := s + a mod 10;
    a := a div 10;
  until a = 0;
  Edit2.Text:=IntToStr(s);
end;
```

Rovnako ako pri brainstormingu musí mať úloha veľa riešení a cieľom aktivity je, aby žiaci našli experimentovaním s programom čo najviac rôznych odpovedí. Zdieľaním výsledkov takéhoto experimentovania sa zefektívňuje fáza zbierania skúseností v poznávacom procese. Žiaci obohatia svoje vlastné

pozorovania o pozorovania spolužiakov. Spoločne vytvorila väčšiu množinu vstupov pre konkrétne výpočty s danou vlastnosťou, ako by získali vlastným experimentovaním. Cieľom diskusie po zbere skúseností je abstrakcia – zistenie, čo program počíta (ciferný súčet zadaného čísla). Námety na ďalšie otázky: *Pre aký vstup prebehne cyklus 3 (alebo iný počet) krát? Ktoré je najmenšie (najväčšie) číslo, pre ktoré cyklus prebehne 3 krát? Zopakovanie úloh s programom, v ktorom konštantu 10 nahradíme číslom 2.* (Počítať sa bude ciferný súčet daného čísla v dvojkovovej číselnej sústave.)

### 3.4 Hlasovanie

Hlasovaním dávame každému žiakovi príležitosť slobodne sa vyjadriť, stať sa dôležitým členom skupiny, na názore ktorého záleží. Hlasovaním sa často nazýva aj jednoduché rozhodovanie o pravdivosti tvrdenia, či rýchly výber z viacerých ponúkaných možností. My však máme na mysli hlasovanie ako zavŕšenie predchádzajúcej tvorivej činnosti zamerané na rozvíjanie kritického myslenia – schopnosti posudzovať kvalitu, robiť rozhodnutia, formulovať hodnotenia. Pomocou hlasovacieho systému žiak svoje hodnotenie alebo rozhodnutie zverejňuje väčšinou anonymne, aby sa zvýšila jeho objektivita.

Hlasovaním je možné napríklad ukončiť brainstorming s cieľom vyfiltrovať spomedzi väčšieho množstva odpovedí kvalitu, napr. najvýstižnejšiu definíciu pojmu alebo zaujímavý nápad na nový projekt a podobne. V systéme *Socrative* je možné zber odpovedí na rýchlu otázku s krátkou odpoveďou ukončiť hlasovaním, v ktorom žiaci označia jednu alebo viac odpovedí, ktoré považujú za dobré, zaujímavé. Systém dovoľuje učiteľovi v priebehu zbierania nápadov ukrývať opakujúce sa odpovede a sprehľadniť tak následné hlasovanie. Výsledky hlasovania sú vyhodnocované automaticky.

Iným príkladom aktivity, v ktorej žiaci vyjadrujú svoj hodnotiaci postoj hlasovaním, je vzájomné hodnotenie práce napríklad po prezentovaní projektu. Hodnotenie je možné v systéme *Socrative* realizovať ako:

- hlasovanie za jeden alebo viac najlepších projektov po odprezentovaní všetkých projektov, výsledné hodnotenie vznikne jedným hlasovaním,
- hodnotenie výberom z možností, ktoré predstavujú bodovú škálu, hlasuje sa viackrát zvlášť pre každý prezentovaný projekt,
- otvorená odpoveď na otázku s krátkou odpoveďou, v ktorej žiaci môžu pridať nielen body podľa vopred dohodnutých pravidiel, ale aj anonymne zverejniť svoj komentár či dopĺňujúcu otázku bezprostredne po odprezentovaní projektu.

Objektivita hodnotenia môže závisieť od toho, nakoľko sú žiaci na takýto spôsob hodnotenia pripravení. Pozornosť by sa preto mala venovať aj krátkemu tréningu konštruktívnej kritiky a pozitívneho, empatického prístupu pri vyjadrovaní sa o iných (napr. prostredníctvom príkladov vhodných a nevhodných formulácií vzájomného hodnotenia). Vizualizovanie celkových výsledkov hlasovania je navyše zmysluplnou ukážkou spracovania informácií pomocou počítača v praxi.

### 3.5 Prieskum postojov

Informácie o záujmoch žiakov a o ich postojoch k obsahu a spôsobu vyučovania sú pre učiteľa cenné pre motivovanie a vzbudenie záujmu o učenie sa. Pre žiaka sú otázky adresované

na zisťovanie jeho postojov príležitosťou na reflexiu a rozvoj metapoznávacích kompetencií.

Otázky zamerané na sebahodnotenie pomáhajú žiakovi rozvíjať schopnosť posúdenia vlastných kvalít, definovať svoj pokrok v učení, ale aj uvedomiť si a pomenovať problémy v učení. Systém *Socrative* ponúka štandardný test na prieskum postojov a sebahodnotenie s tromi otázkami s krátkou odpoveďou, ktoré sa kladú v závere vyučovania:

*Ako dobre si porozumel dnešnému učivu? Čo si sa naučil na dnešnej hodine? Odpovedz, prosím, na otázku učiteľa.*

Súčasťou takéhoto záverečného prieskumu realizovaného pomocou hlasovacieho systému môže byť výzva na polozenie otázok, predovšetkým keď sa jedná o prednášku pre veľké auditorium, kde prednášateľ nemá blízky kontakt s publikom. Otázky môžu byť zodpovedané okamžite, alebo byť námetom na domáce zadanie.

Iným príkladom aktivity na zisťovanie názorov je motivačný prieskum realizovaný pred riešením nejakého problému. Po zadaní problému sú žiaci vyzvaní, aby odhadli jeho riešenie. Uvažovanie nad možnou odpoveďou podnecuje žiakov najprv hlbšie pochopiť problém, až potom ho začať riešiť.

Príklad problému: *Dvaja hráči hádzu mincou. Prvý hráč vyhráva, keď bezprostredne za sebou padne štyrikrát písmo, druhý hráč vyhrá, keď bezprostredne za sebou padne dvakrát znak a potom dvakrát písmo. Kto má väčšiu šancu na výhru?*

Hlasovací systém zaznamená odpovede žiakov, ktoré na úvod formulujú ako odhady riešenia na základe intuície, presvedčenia vychádzajúceho z ich dovtedajších skúseností a vedomostí. Po vyriešení problému napríklad pomocou počítačovej simulácie veľkého množstva pokusov je možné porovnať odhady so skutočným riešením.

## 4. ZÁVER

Systém *Socrative* používame vo vyučovaní na vysokej škole v rámci prednášok a cvičení v predmetoch orientovaných na teoretickú informatiku, programovanie a didaktiku informatiky. Naše skúsenosti naznačujú, že použitie hlasovacieho systému na zadávanie kontrolných otázok na prednáškach vo veľkej miere prispieva k vytváraniu pozitívnej komunikačnej klímy, podporuje záujem študentov o obsah prednášky, podnecuje myslenie. Prednáška je viac interaktívna, aktivita študentov je vyššia, učenie sa je viac sociálne.

V detskom informatickom tábore sme systém *Socrative* použili pri hlasovaní o najlepšom z prezentovaných projektov a pri realizovaní záverečného dotazníkového prieskumu. Príležitosť vyjadriť svoj názor a hodnotiť bola pre deti pozitívnym motivujúcim zážitkom. Zaujímavé bolo pre ne tiež použitie tabletov a hlasovacieho systému ako nástroja na zber dát.

V dotazníkovom prieskume, ktorý sme uskutočnili so 140 učiteľmi informatiky a informatickej výchovy, uviedlo 13 % opýtaných, že ich škola má k dispozícii hlasovacie zariadenia, z nich asi polovica (7 %) uviedla, že ich aj aktívne používa vo vyučovaní. Tablety určené na vyučovanie má podľa tohto prieskumu až 46 % škôl. Žiadny z respondentov však neuviedol v odpovedi na otvorenú otázku: „Akú mobilnú aplikáciu by ste odporučili na použitie vo vyučovaní?“ softvér *Socrative* alebo inú aplikáciu pre tablety a smartfóny s funkcionalitou hlasovacieho systému.



Z výsledkov prieskumu vyplýva, že technológia hlasovacích systémov nie je medzi učiteľmi celkom neznáma, avšak špeciálne hlasovacie zariadenia a ich používanie na vyučovaní nie sú v školách bežné. Naproti tomu podiel škôl, ktoré sú vybavené tabletmi, sa blíži k polovici z oslovenej vzorky. To vytvára dobré predpoklady na rozšírenie používania hlasovacích systémov (založených na koncepte BYOD) ako modernej interaktívnej technológie vzdelávania.

Naše skúsenosti s používaním systému *Socrative* sú pozitívne. Poukazujú na to, že nové interakcie v triede, ktoré vznikajú ako dôsledok používania hlasovacieho systému, podnecujú aktívne a kolaboratívne učenie sa študentov. Bezprostredný pozitívny vplyv na vzdelávacie výsledky sme nezaznamenali. Myslíme si, že sú prívelmi ovplyvňované vonkajšou motiváciou študentov uspieť na skúške. Technológia hlasovacieho systému sa zameriava viac na zvýšenie vnútornej motivácie.

## 5. POĎAKOVANIE

Príspevok vznikol v rámci riešenia projektu *Mobilné technológie v škole pre 21. storočie* podporovaného grantovou agentúrou KEGA.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] PECINA, Václav, ŠRUBAŘ, Jiří. Poznámka ke strojovému zkoušení. In *Pokroky matematiky, fyziky a astronomie*, Vol. 18 (1973), No. 4, p. 212--215

- [2] Socrative [online]. [cit. 2015-02-04]. Dostupné na: <<http://www.socrative.com>>.
- [3] Socrative Garden. *Growing 21<sup>st</sup> Century Skills* [online]. [cit. 2015-02-04]. Dostupné na: <<http://garden.socrative.com>>.
- [4] MÉNDEZ-COCA, D., SLISKO, J. Software Socrative and Smartphones as Tools For Implementation of Basic Processes of Active Physics Learning in Classroom: An Initial Feasibility Study With Prospective Teachers. In *European Journal of Physics Education*, ISSN 1309-7202, 2013, Vol.4(2), p. 17-24.
- [5] DERVAN, P. Enhancing In-class Student Engagement Using Socrative (an Online Student Response System): A Report. In *AISHE-J: The All Ireland Journal of Teaching and Learning in Higher Education*, 2014, ISSN: 2009-3160 Vol 6, No 3, p.1801-1812
- [6] LIU, D. Y. T., TAYLOR, C. E. Engaging students in large lectures of introductory biology and molecular biology service courses using student response systems. In *Proceedings of the Australian Conference on Science and Mathematics Education (ACSME) (Australian National University, 19–21 September 2013)*, pp. 154-162

# Vyučovanie základov programovania v jazyku Python na niektorých SŠ a VŠ

Eva Mészárosová

Katedra základov a vyučovania informatiky, FMFI  
Univerzita Komenského v Bratislave  
Mlynská dolina 1  
842 48 Bratislava  
Slovensko  
meszarosova@fmph.uniba.sk

Monika Tomcsányiová

Katedra základov a vyučovania informatiky, FMFI  
Univerzita Komenského v Bratislave  
Mlynská dolina 1  
842 48 Bratislava  
Slovensko  
tomcsanyiova@fmph.uniba.sk

## ABSTRAKT

V článku popisujeme úvodné kurzy programovania na dvoch stredných školách a na dvoch univerzitách na Slovensku a v zahraničí, ktoré prebiehajú v jazyku Python. Skúmame postupnosti preberaných tém z výukových materiálov týchto kurzov, pričom sa sústreďujeme na ich porovnanie s učebnicou programovania pre stredné školy, ktorá sa v súčasnosti používa na Slovensku. Výsledkom analýzy výukových materiálov sa snažíme vyšpecifikovať prvé poradie tém pre metodiku vyučovania základov programovania v jazyku Python pre stredné školy.

## ABSTRACT

In this paper we describe introductory programming courses at two secondary schools and two universities in Slovakia and foreign countries, which utilize the Python programming language. We explore the sequence of topics in the materials used by these courses, while we also compare them with the programming coursebook most commonly used in Slovakia. We strive to specify a primary sequence of topics for introductory programming courses in Python at secondary schools.

## Kľúčové slová

vyučovanie programovania, Python, výukové materiály

## Keywords

Teaching programming, python, coursebooks

## 1. ÚVOD

Vyučovanie základov programovania na stredných školách a gymnáziách je zamerané na vyučovanie algoritmov, rozvoja algoritmického myslenia a schopnosti riešenia problémov. Jazyk vhodný pre tento účel by mal umožniť žiakom a učiteľom sústrediť sa na algoritmy a získanie programátorských zručností a kompetencií žiakov a mal by žiakov čo najmenej zaťažovať vlastnosťami samotného programovacieho jazyka alebo prostredia, v ktorom programovanie prebieha. V minulosti sa žiaci na stredných školách učili programovať v jazyku Pascal a vznikla aj učebnica pre stredné školy. Zo zahraničných zdrojov vieme, že do popredia sa v súčasnosti dostáva jazyk Python a čoraz viac univerzít a stredných škôl prechádza na vyučovanie základov programovania v tomto jazyku. Existujú učebnice jazyka, ale tie sa často sústreďujú viac na syntax, resp. upozorňujú na zmeny, ktoré používa jazyk Python oproti iným programovacím jazykom. Väčšina učiteľov však potrebuje na vyučovanie také materiály, ktoré by učili žiakov jazyk počas riešenia úloh. Preto si mnohí učitelia vytvárajú vlastné materiály na vyučovanie programovania. Naším cieľom bolo vyhľadať niekoľko výukových materiálov a porovnať postupnosť tém a úloh, ktoré

využívajú na vyučovanie základov programovania na stredných, resp. v úvodnom kurze na vysokých školách.

## 2. PROGRAMOVACÍ JAZYK PYTHON

Jazyk Python vznikol v roku 1989 a jeho autorom je Guido van Rossum. Je to interpretovaný jazyk, ktorý je vyvíjaný ako open source projekt. Jazyk Python je multiplatformový a v praxi veľmi rozšírený.

Stoffova, pozri [1], uvádza, že vyučovanie programovania na gymnáziách prebieha najmä v programovacom jazyku Pascal, v prostredí Delphi a Lazarus, a na stredných odborných školách v jazyku C, pričom v súčasnosti čoraz viac univerzít a stredných škôl prechádza na jazyk Python. Zatiaľ však nie sú známe učebnice programovania v jazyku Python pre stredné školy na Slovensku, práve preto učitelia používajú materiály stiahnuté z internetu alebo si vytvárajú vlastné materiály.

Za hlavné rozdiely v syntaxi jazyka Pascal a Python sa považujú:

- V jazyku Pascal (Delphi/Lazarus) sa označujú konce príkazov bodkočiarkou, v jazyku Python je zvykom písať každý príkaz do nového riadku.
- Bloky príkazov sa v Pascale označujú príkazmi begin a end, v jazyku Python sú bloky príkazov určené odsadením, pričom pred blokom príkazov sa uvádza dvojbodka na konci predchádzajúceho príkazu (if, for, while, def).
- V jazyku Pascal deklarujeme typy premenných, v Pythone sa typy premenných nedeclarujú.

**Tabuľka 1.** Ukážka funkcie `je_prvocislo`. Porovnajme zložitosť kódu v jazyku Python a Pascal.

```
def je_prvocislo(x):
    for i in range(2,x):
        if x % i == 0:
            return False
    return True

function JePrvocislo(cislo:integer):boolean;
var
    i:integer;
begin
    Result:=true;
    for i:=2 to cislo-1 do
        if (cislo mod i) = 0 then Result:=false;
    end;
```

Názory mnohých odborníkov a učiteľov programovania sa zhodujú v tom, že programovací jazyk Python je vhodným jazykom na vyučovanie základov programovania, pozri [2] a [3]. Python svojou jednoduchou syntaxou mnohým pripomína pseudokód, t.j. programovací jazyk, v ktorom sa dajú prehľadne zapisovať rôzne algoritmy. Samotný Python nezaťažuje žiakov

komplikovaným vývojárskym prostredím a medzi jeho výhody patrí aj mnohostranné využitie v praxi, jazyk Python je čoraz viac rozšírený v rôznych odboroch.

### 3. PREHĽAD VYBRANÝCH KURZOV PROGRAMOVANIA

V našom výskume sme vyhľadávali základné kurzy programovania na stredných a vysokých školách, ktoré používajú Python. Ďalej uvedieme a z pohľadu preberaných tém porovnáme dva kurzy z vysokých škôl a dva zo stredných škôl.

#### 3.1 Massachusetts Institute of Technology

Kurz Introduction to Computer Science and Programming je základný kurz programovania, ktorý vedie prof. John Guttag na MIT. Kurz je určený pre študentov, ktorí nemajú takmer žiadnu alebo len minimálnu skúsenosť s programovaním. Obsah kurzu je zameraný na porozumenie riešenia problémov z informatického hľadiska. Pre niektorých študentov je tento kurz základným kameňom k pokročilejšiemu kurzu informatiky (computer science course), avšak pre väčšinu študentov je jediným informatickým kurzom počas ich štúdií. Práve preto sa v kurze zameriavajú skôr na šírku preberaného učiva ako na hĺbku preberaných konceptov. Ich cieľom je, aby študenti získali všeobecný prehľad a základné zručnosti riešenia informatických problémov (computational problem solving). Z lekcii svojho kurzu prof. Guttag pripravil knihu, pozri [4]. Z obsahového hľadiska je jeho kniha spracovaná pomerne detailne a na konci každej jej časti sú uvedené príklady, v ktorých však viditeľne prevláda matematický kontext.

**Tabuľka 2.** Ukážka úlohy z učebnice [4].

*Napiš funkciu obsahujúcu, ktorá dostane dva znakové reťazce ako parameter a vráti hodnotu True, ak niektorý z reťazcov obsahuje druhý reťazec. Inak vráti hodnotu False.*

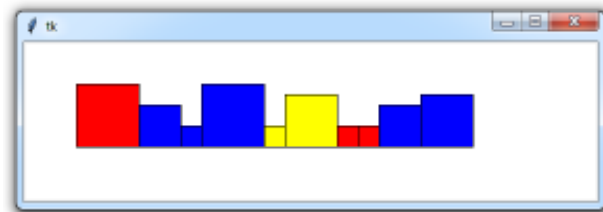
#### 3.2 FMFI Univerzity Komenského v BA

Vyučovanie úvodného kurzu programovania pre študentov v odbore aplikovaná informatika na Fakulte matematiky, fyziky a informatiky UK v Bratislave prebieha v jazyku Python (verzia 3.4.1) už druhý školský rok. Počas zimného semestra majú študenti cca 22 prednášok (90 min.) a 12 cvičení (90 min.). Počas letného semestra pokračuje základný kurz 12 prednáškami a 12 cvičeniami. Kurz bol organizovaný tak, že najprv (v pondelok a v utorok) prebehli dve dvojhodinové prednášky a s odstupom jedného dňa mali študenti jedno dvojhodinové cvičenie. Na cvičeniach mali študenti k dispozícii skupinu úloh, väčšinou 7 až 14, ktoré vypracovávali buď samostatne alebo vo dvojiciach, pričom si radili a pomáhali si navzájom. Na konci semestra pred skúškou mal každý študent naprogramovať semestrálny projekt, v ktorom mali použiť vedomosti a zručnosti nadobudnuté počas semestra. Kurz vyučuje RNDr. Andrej Blaho, PhD. a prednášky a cvičenia sú zverejnené na [5]. Jednotlivé prednášky sú rozsiahle a témy v nich sú rozpracované podrobne. Po obsahovej stránke sú prednášky spracované pomerne podrobne. Sú však skôr orientované na vysvetlenie detailu jazyka a ukážku jednoduchého použitia nejakého základného príkazu. Úlohy z cvičení sú potom orientované na využitie naučených príkazov pri riešení jednoduchých aj zložitejších algoritmickej problémov.

**Tabuľka 3.** Ukážka zadania úlohy z cvičení z kurzu Programovanie 1

*Funkcia `rad_kociiek(x,y,n)` pomocou funkcie `kocka(x,y)` nakreslí vedľa seba  $n$  farebných štvorcov tak, aby ležali tesne vedľa seba na myslenej čiare, prvý z nich nech má súradnice*

*napr. (50,100).*



Obrázok 1. Obrázok z riešenia úlohy z cvičení

#### 3.3 ŠpMNDaG v Bratislave

V Škole pre mimoriadne nadané deti a Gymnáziu sa vyučujú základy programovania v jazyku C. Ako pokračovanie vyučovania programovania sa už štvrtý rok žiaci následne oboznámia aj s jazykom Python. V ďalších rokoch štúdiá si žiaci sami zvolia, ktorý programovací jazyk budú používať. V súčasnosti začali experiment, v ktorom prebieha aj úvodný rok vyučovania základov programovania v jazyku Python. Vyučuje sa podľa materiálu vytvoreného pánom učiteľom Belanom, ktorý napísal internú učebnicu, pozri [6] pre žiakov svojej školy. Jeho učebnica je určená pre žiakov, ktorí už prešli úvodným kurzom programovania v jazyku C, a jazyk Python je pre nich už v poradí druhý vyšší programovací jazyk. Zrejme aj práve kvôli tomu sa autor v učebnici dlho nezaobereá vysvetľovaním základných pojmov programovania. Zaujímavosťou je, že autor uvádza najprv príklad a až potom vysvetlenie danej tematiky na príklade, pričom učebnica obsahuje relatívne málo úloh na precvičenie (1-4 úlohy). Text učebnice a výkladu je vtipný a pre žiakov určite veľmi pútavý. Po obsahovej stránke učebnica obsahuje požadované pojmy a termíny zo ŠVP.

**Tabuľka 4.** Ukážka zadania úlohy z materiálov zo ŠpMNDaG

*Napište program, ktorý šesťstokrát hodí kockou a vypíše vám, koľkokrát padla šestka.*

#### 3.4 Stredná odborná škola v Kiskunlacháza, Maďarsko

Na tejto strednej škole vyučujú základy programovania pre žiakov na informaticky zameraných odboroch. Žiaci sa oboznamujú so základmi programovania najprv v jazyku Python, neskôr sa učia aj Javascript a zoznámia sa aj s PHP. Učitelia na tejto škole si vytvorili interné materiály pre programovanie v jazyku Python, pričom využili učebnicu pre OOP v Jave [7] a [8]. Materiály vytvorené kolektívom učiteľov z tejto školy sú rozdelené na štyri polroky (štyri dokumenty) [9, 10], pričom v prvom polroku sa zaoberajú úvodom do programovania, v druhom polroku prácou s grafickou plochou. V ďalšom roku štúdiá venujú prvý polrok práci so súborami, triediacimi algoritmiami a prácou s databázami. V poslednom polroku, v ktorom vyučujú programovanie v jazyku Python, sa zaoberajú úvodom do tvorby aplikácií pre web v jazyku Python, pričom preberajú interaktívne weby, CGI interface, prácu s databázou a vytváranie fór. Tieto materiály sú prispôbené zameraniu školy. Podľa učiteľa informatiky, s ktorým sme komunikovali, sa ich žiaci väčšinou predtým nestretli s programovaním, ale niektorí už programovali v detskom programovacom prostredí Imagine Logo. Uviedli, že jedným z dôvodov, prečo si vybrali práve jazyk Python, patrí aj to, že tento jazyk motivuje žiakov viac ako napríklad Delphi alebo C++, a tiež to, že v Maďarsku od roku 2012 môžu žiaci používať

na maturitách z informatiky aj jazyk Python. Pre účely nášho výskum stačí, ak sa sústredíme na analýzu prvých dvoch častí z uvedených materiálov. Po obsahovej stránke sú tieto výučbové materiály spracované analogicky a väčšinou pomocou príkladov. Najprv je však vždy uvedený nový pojem alebo termín s vysvetlením a neskôr príklad ako ukážka. Po každej téme sú uvedené skupiny úloh na precvičenie novej tematiky. Ako zaujímavosť knihy by som uviedla, že príklady majú väčšinou matematický kontext a počas niekoľkých tém sa príklady opakujú a graduujú – prevažuje téma bankovníctva.

**Tabuľka 5.** Ukážka zadania úlohy z materiálov zo ŠpMNDaG

*Kolko si zarobí ročne Jozef, ak mu každý rok zvýšia výplatu o 12%? Výšku prvej výplaty zadávajú pomocou editovacieho riadka a rátať od roku 2009 do 2026.*

### 3.5 Učebnica Informatika pre stredné školy: Programovanie v Delphi a Lazaruse

V rámci pilotnej štúdie k nášmu výskum sme zrealizovali prieskum medzi učiteľmi na Slovensku, ktorí učia informatiku na gymnáziách alebo stredných odborných školách vo väčších mestách. Z ich odpovedí vyplynulo, že na vyučovanie programovania väčšinou používajú práve túto učebnicu [11]. Učebnica bola vydaná v roku 2012, a je už druhým upraveným vydaním, ktoré je schválené Ministerstvom školstva SR. Z obsahového hľadiska pokrýva všetky dôležité témy, pojmy a koncepty zo ŠVP. Z didaktického hľadiska je učebnica spracovaná detailne s vhodnými ilustratívnymi príkladmi použitia. Výklad je doplnený príkladmi a tiež ilustráciami a po každej časti sú uvedené úlohy na precvičenie nového učiva.

**Tabuľka 6.** Ukážka zadania úlohy

*Program by mal pri ťahaní myšou kresliť stále zmenšujúce štvorce. Prvý štvorec má dĺžku strany 50, každý ďalší je o 1 menší. Po kreslení štvorca so stranou 10 sa opäť pokračuje dĺžkou 50.*

V súčasnosti vyučujú základy programovania v jazyku Python aj na niektorých školách v Česku a v Anglicku. Komunikovali sme napr. aj s jedným učiteľom z gymnázia z Roudnice nad Labem, ktorý je hrdý na úspechy svojich žiakov v programátorských súťažiach. Podľa neho je dôležitým prvkom toho, že sa zúčastnili a uspeli na krajskom kole programátorskej súťaže aj jazyk, v ktorom sa žiaci učia programovať. Uviedol tiež, že Python vyučujú podľa vlastných materiálov a materiálov stiahnutých z internetu.

## 4. VÝSLEDKY PRIESKUMU

V predchádzajúcej kapitole sme opísali štyri kurzy programovania a ich materiály. Analýzou uvedených materiálov sme získali poradie tém v jednotlivých kurzoch. Tieto postupnosti tém porovnávame v nasledujúcej tabuľke, pozri Tabuľku 7. V tabuľke sú uvedené témy, ktoré sú určené ŠVP a ktoré v súčasnosti zastrešuje učebnica informatiky pre stredné školy, stĺpec uDL, a tiež témy, ktoré obsahovali kurzy, ktoré sme skúmali.

Čísla v tabuľke znamenajú poradie témy v danom kurze, pričom polia, n-tice, množiny a slovníky sme označili jedným číslom – prvá cifra v označení znamená spoločné poradie v kurze a posledná poradie v rámci tejto skupiny (7.2 – siedma téma a druhá v poradí medzi jednoduchými údajovými štruktúrami – pole, n-tice, množiny a slovníky).

Z tabuľky 7 vyplýva, že kurzy v Pythone začínajú úvodom do jazyka Python a vývojového prostredia IDLE, pričom hneď v úvode sa zaoberajú vytvorením premenných a priradeniu príkazu. Práca s grafickou plochou sa objavuje len v kurze z FMFI – v ďalších kurzoch/materiáloch preberajú grafickú plochu a prácu s grafikou buď medzi poslednými témami alebo sa týmto témam vôbec nevenujú. Podmienené príkazy prebrali v kurzoch MIT, FMFI UK a KSZI na druhom mieste hneď za úvodom do jazyka a premennými, pričom v kurze na FMFI UK sa táto téma prebrala až po príkazoch grafickej plochy. For cykly a while cykly väčšinou preberali za sebou v rôznom poradí, výnimkou je kurz z FMFI UK, kde sa najprv prebrali for-cykly, potom podmienky a logické výrazy a až na šiestom mieste sa zaoberali cyklami while. Z tabuľky vidíme, aký veľký rozdiel je medzi umiestnením tém, v ktorých sa preberali polia, n-tice, množiny a slovníky, pričom ich miesto v poradí tém vplyva na výklad učiva a ďalšie úlohy. Práci so súbormi, objektami a rekurziou sa všetky kurzy venujú na konci. Vieme, že posledne spomínané témy nie sú súčasťou ŠVP pre všeobecné vzdelávanie v informatike a uvádzame ich len pre úplnosť nášho výskumu. Uvedomujeme si aj to, že uvádzané kurzy nie sú určené pre rovnaké cieľové skupiny, keďže v ďalšom našom výskume sa budeme venovať všeobecno-vzdelávaciemu predmetu informatika a miestu programovania v ňom. S touto cieľovou skupinou čiastočne korešponduje učebnica Informatika pre stredné školy: Programovanie v Delphi a Lazaruse (v ktorej sa však objavujú aj témy k maturite) a kurz strednej odbornej školy v Kiskunháza v Maďarsku. Pre náš výskum je však dôležité aj to, aby sme získali prehľad o existujúcich výukových materiáloch a poradí preberaných tém.

**Tabuľka 7.** Poradie tém v kurzoch programovania [1]

Kniha / kurz	MIT	FMFI	ŠpMND	KSZI	uDL
premenné a priradenie	1	1	1	1	2
grafická plocha	-	3	11	10	1
riadiace štruktúry: podmienené príkazy	2	5	2	2	7
riadiace štruktúry: cykly (for)	5	2	5	4	4
riadiace štruktúry: cykly (while)	4	6	6	3	6
funkcie (procedúry)	6	7	8	5	5
práca so súbormi	10	9	-	15	11
náhodné čísla	-	4	7	8	3
polia	11.2	10.2	4.1	7.1	10
n-tice	11.1	10.1	4.2	7.2	-
množiny	11.4	10.3	-	-	-
slovníky, ar.polia	11.3	15.2	4.3	-	-
dvozmerné polia	15	15.1	-	-	-
objekty, triedy, dedičnosť	14	14	12	9	-
korytnacia grafika	-	11	-	-	-
rekurzia	8	12.1	-	11	-

## 5. PREDBEŽNÉ PORADIE TÉM

Z poradia preberaných tém v uvedených kurzoch s ohľadom na potreby a požiadavky učiteľov, ale aj s ohľadom na možnosti a špecifiká jazyka Python začneme postupne navrhovať poradie

tém pre metodiku vyučovania základov programovania v jazyku Python pre stredné školy. Zatiaľ sme sa rozhodli a zvolili sme prvé tri témy: premenné, grafická plocha, for-cykly.

Aj napriek tomu, že napr. v kurze MIT sa grafická plocha neobjavuje vôbec, z prieskumu a rozhovorov, ktoré sme zrealizovali s učiteľmi informatiky sme zistili, že väčšina učiteľov preferuje vyučovanie programovania práve na úlohách, ktoré sú graficky zaujímavé. Učitelia to zdôvodňovali tým, že pre žiakov je grafická spätná väzba omnoho motivujúcejšia ako textová, a úlohy, v ktorých sa využíva kreslenie do grafickej plochy sú pre žiakov atraktívne. Je teda žiadúce, aby sa téma, ktorá sa venuje grafickej ploche vyučovala na niektorých z prvých vyučovacích hodín programovania. Po téme grafická plocha navrhujeme v našej postupnosti tém zaradiť do vyučovania tému, v ktorej sa na grafických úlohách využije riadenie programu pomocou for-cyklu. Pomocou opakovania a zmenou, resp. využívaním riadiacej premennej cyklu, môžu žiaci kresliť rôzne grafické objekty a umiestňovať ich na grafickú plochu. Takto pripravené úlohy môžu byť navrhnuté tak, že majú gradáciu a žiaci sa pri ich riešení dôkladne oboznámia s využitím for-cyklov.

## 6. ZHRNUTIE

Predmetom ďalšieho nášho ďalšieho výskumu bude, aké témy jazyka Python sú vhodné pre všetkých žiakov strednej školy a aké úlohy sú vhodné na vyučovanie týchto tém. Pri výbere tém budeme sledovať, aby sme nimi pokryli obsahový štandard ŠVP ISCED 3A, pozri [12]. Už teraz vidíme, že samotné poradie tém sa nemusí (a zrejme ani nebude) veľmi líšiť od poradia tém v učebnici *Informatika pre stredné školy: Programovanie v Delphi a Lazaruse*, pozri [11], keďže sme presvedčení, že poradie navrhnuté v tejto učebnici je pre didaktické účely vyhovujúce a odvíja sa od tradície vo vyučovaní programovania na Slovensku. Jazyk Python a niektoré jeho špecifiká však môžu do značnej miery ovplyvniť výber a formuláciu jednotlivých úloh tej-ktorej témy.

## 7. ZÁVER

V článku sme popísali štyri kurzy programovania a ich výukové materiály. Poradie tém v týchto materiáloch sme porovnali navzájom a tiež s poradím tém z učebnice, pozri [11], ktorú používa väčšina učiteľov na vyučovanie základov programovania na stredných školách. Pre náš ďalší výskum sme spravili prvé závažné rozhodnutie: navrhnuť metodiku na vyučovanie programovania pre stredné školy v jazyku Python tak, že jednou z prvých tém, s ktorou sa žiaci stretnú, bude grafická plocha a možnosti kreslenia v nej. Ďalším našim výskumom budeme zisťovať, ako toto prvé rozhodnutie ovplyvní poradie ďalších tém v našej metodike a na akých úlohách sa budú žiaci oboznamovať s konceptami, ktoré vyžaduje ŠVP ISCED 3A.

## 8. BLIOGRAFICKÉ ODKAZY

- [1] STOFFOVÁ, V., CZAKÓOVÁ, K. Propedeutika programovania a nová školská reform, In *Journal of Technology and Information Education, Časopis pro technickou a informační výchovu*, 3/2012, vol. 4, issue 3, ISSN 1803-537X.
- [2] ZELLE, John M. *Python as a First Language*. Department of Mathematics, Computer Science, and Physics Wartburg College [online]. [cit. 2015-02-03]. Dostupné na: <<http://mcsp.wartburg.edu/zelle/python/python-first.html>>.
- [3] ATTEQ, M. a spol. *C++ or python? Which One to Begin With: A Learners Perspective*. In 2014 International Conference on Teaching and learning in Computing and Engineering, IEEE. ISBN 978-1-4799-3592-5/14.
- [4] GUTTAG, John V. *Introduction to Computation and Programming Using Python (Spring 2013 Edition)*. Massachusetts Institute of Technology. Cambridge: Press, 55 Hayward Street, MA 02142. ISBN 978-0-262-51963-2.
- [5] BLAHO, A. *Programovanie v Pythone* [online]. [cit. 2015-02-03]. Dostupné na: <<http://python.input.sk/>>.
- [6] BELAN, A. *PYTHON, učebný text pre septimu osemročného gymnázia*. Bratislava, 2013 [online]. [cit. 2015-02-03]. Dostupné na: <<http://www.smnd.sk/anino/moje/Python.pdf>>.
- [7] ANGSTER, E. *Objektumorientált tervezés és Programozás, JAVA*. 3. vyd. Martonvásár: Akadémiai Nyomda, 2003. ISBN 963-00-6263-1.
- [8] DOWNEY, A. *Think Python, How to Think Like a Computer Scientist*. O'Reilly Media, Inc. 1005 Gravenstein Highway North, Sebastopol. ISBN 978-1-449-33072-9.
- [9] PETO, L a kol. *Bevezetés a Python programozásba (Informatikai szakközépiskola – 10. évfolyam)* [online]. [cit. 2015-02-03]. Dostupné na: <<http://szerver2.lacszi.sulinet.hu/tananyag/informatika/python.pdf>>.
- [10] PETO, L a kol. *Bevezetés a Python Tkinter programozásába (Informatikai szakközépiskola – 10. évfolyam)* [online]. [cit. 2015-02-03]. Dostupné na: <<http://szerver2.lacszi.sulinet.hu/tananyag/informatika/tkinter.pdf>>.
- [11] BLAHO, A. *Informatika pre stredné školy: Programovanie v Delphi a Lazaruse*. 2 vyd. Prešov: Polygraf print, spol. s r.o., 2012. ISBN 978-80-10-02308-0.
- [12] Štátny pedagogický ústav. Štátny vzdelávací program, Informatika – príloha ISCED 3A. Bratislava, 2008.



# Videonávody na školní webové stránky

Ing. Bc. Václav Nádvorník  
Základní škola, Praha 2, Londýnská 34  
Londýnská 34  
120 00, Praha 2  
Česká republika  
nadvornik@londynska.cz

## ABSTRAKT

Videonávody uveřejňované na školní webové stránce mají potenciál být velmi efektivní vzdělávací pomůckou pro žáky školy. V článku jsou zveřejněny různé možnosti, jak se školní videonávody mohou tvořit a jaké jsou možnosti jejich využití. Autor vychází nejen ze svých mnohaletých zkušeností, ale také z průzkumu dalších výukových kanálů na serveru youtube.com a z literatury, která se tématem zabývá.

## ABSTRACT

Video tutorials published on the school website have the potential to be a very effective learning tool for pupils of the school. The article published various ways how videotutorials can be done and what are the possibilities for their use. The article is based on not only authors years of experience, but also from a survey of other educational channels on youtube.com and literature that addresses the issue.

## Klíčové slová

videonávod, videotutorial, školní webová stránka, převrácená třída

## Keywords

videotutorial, school web page, flipped classroom

## 1. ÚVOD

Školní webová stránka je prezentace činnosti školy a základní informační portál zveřejňovaný prostřednictvím sítě internet. Školní webová stránka nemá pouze prezentační cíl, ale má sloužit i jako zdroj informací pro žáky. Jednou z podstatných částí je možnost předávat skrz školní webovou stránku i informace týkající se výuky a samotné výukové materiály. Je zcela běžné zadávat přes stránku domácí úkoly a publikovat pracovní listy. Jako možná nadstavba je umísťování videonávodů, které žákům pomohou při domácí přípravě.

## 2. Funkce výukového videozáznamu

A. Nelešová v knize Pedagogická komunikace v teorii a praxi [1] rozděluje výukové videozáznamy dle jejich funkce následovně:

### Funkce motivační a inspirační

Videozáznam sleduje v tomto případě cíl motivovat, inspirovat k hledání progresivních didaktických postupů. Toho lze dosáhnout promyšleným seřazením krátkých výseků z rozsáhlejších videozáznamů a jejich vhodnou prezentací a interpretací.

### Funkce ilustrační

Videozáznam plní funkci ilustrace k výkladu.

### Funkce dokumentační

Videozáznam slouží k dokumentaci, uchování výjimečných, neopakovatelných nebo obtížně přístupných pedagogických jevů a procesů.

### Funkce instruktážní, demonstrační a imitační

Videozáznam slouží jako součást instruktáže při demonstraci a nácviku určité didaktické činnosti, vytváření dovednosti apod. Využívá se i simulovaných nebo režirovaných (hraných) ukázek.

### Funkce stimulační

Videozáznam vyvolává představy, ovlivňuje stanoviska a názory. Zdánlivě se tato funkce kryje s motivační, ale v tomto případě vycházíme zejména z příkladů didaktických postupů řady učitelů k určitému pedagogickému jevu.

### Funkce integrační, syntetická, zobecňující

Videozáznam v tomto případě sjednocuje, zobecňuje určité pohledy, názory a stanoviska. Přispívá k systémovému chápání sledovaného jevu, k uvědomění si jeho relevance či irelevance

### Funkce diferenciativní

Videozáznam respektuje, umožňuje a zajišťuje hledisko diferenciací v přenášených informacích i na straně uživatelů

### Funkce hodnotící

Videozáznam prezentuje požadované vzory, stanoviska, procesy a tím plní i funkci hodnotící, neboť předkládá "jak by to mělo být" a vybízí ke srovnání

### Funkce analytická

Videozáznam sleduje analytický přístup ke skutečnosti, orientuje se na vybrané jevy a procesy, které může diagnostikovat, fázovat, zpomalovat či zrychlovat apod.

### Funkce cílová

Videozáznam je jednoznačně orientován na dosažení cíle

## 3. Videozáznam a jeho využití ve výuce.

Velkým průkopníkem videonávodů a celého konceptu převrácené třídy je tzv. Khanova škola. Původcem myšlenky je Salman Khan, který v USA začal postupně tvořit videonávody z oblasti matematiky, které si žáci studovali před samotnou výukou. Výhodou tohoto konceptu je, že učitel má šanci v hodině již "stavět" na určitých znalostech. Postupem doby se rozšířilo množství takto vysvětlovaných předmětů a nyní jsou natočeny desítky videí v Khanově stylu. Proces učení každého žáka může být zdokumentován pomocí statistik a hodnocení. Žáci jsou také motivováni získáváním odznaků za všemožné činnosti. Některé dostanou snadno, ale u jiných se musí velmi snažit.

### Standardní způsob výuky

- Učitel vhodně zvolenou metodou ve škole s žáky vykládá látku. Metod a forem práce jsou desítky a mnohé jsou velmi účinné.
- Žáci si dělají poznámky pro domácí studium
- Žáci dostanou zadaný domácí úkol
- Žáci si doma látku procvičují a vypracovávají domácí úkol
- Učitel vše zkontroluje a žáky ohodnotí

### Princip převrácené třídy

- Učitel základní výklad připraví formou videonávodu
- Žáci předem doma videonávod nastudují
- Na hodinu přicházejí připravení a diskutují s učitelem, hledají rozšiřující vazby
- Učitel pomáhá žákům zvládat případné obtíže[2]

V metodě převrácené třídy nahrazuje přímý výklad učitele online vzdělávací video. Výuka probíhá tak, že se žáci nejprve seznámí prostřednictvím videa s probíranou látkou doma online, do školy přijdou již s konkrétními dotazy. Svoje dotazy mohou žáci vložit jako komentář k videu nebo do prostředí sociální, které sdílí s učitelem. Ten otázky setřídí a připraví výukové aktivity zaměřené na problematiku témat. Bezproblémová témata přitom vynechává. V hodině potom učitel využívá čas efektivněji, protože se věnuje hlavně těm pasážím probírané látky, kterým děti na videu nerozuměly. Během výuky se žáci učí diskutovat a vznášet dotazy k probírané látce. Tím si utříbí znalosti a získávají hlubší vhled do problematiky. Z pasivních posluchačů se mění na aktivní studenty. Při domácím sledování videa každý žák získává možnost postupovat v látce vlastním tempem. Zároveň se učí být zodpovědný za svoje vlastní vzdělávání. Metoda Převrácené třídy dává příležitost dostat se více do hloubky probírané látky a umožňuje každému žákovi dospět na maximum svých schopností.

Nejedná se však o nijak přelomovou a novátorskou metodu, vždyť učitelé často dávali před hodinami nastudovat témata o kterých poté v hodinách s žáky diskutovali. Rozdíl byl pouze v médiu, které pomohlo žákům v předchozím studiu. Autor se sám ve dvou případech inspiroval metodou převrácené třídy a pro určitá témata použil své videonávody pro předchozí samostudium. Výsledek, nebyl ale takový, jaký by očekával. V obou případech si z 50 žáků video pozorně prohlédlo pouze 40 a tudíž pro 20% žáků musel stejně výklad proběhnout. I když je internet zcela běžnou součástí cca 90% domácností s dětmi, stále musí učitel při plánování hodin počítat s možností, že žáci přístup na internet nemají.

Pro svoji další práci učitele pak autor zvolil kombinaci metod. Jak bude dále v článku uvedeno, autor tvoří pravidelně výuková videa, ale ta slouží žákům primárně k zopakování výkladu, který proběhl ve škole a k domácímu procvičování. Jako sekundární jev pak tato videa používají tisíce jiných uživatelů, než jsou autoři žáci.

## 4. Autorovo pojetí videonávodů

Z autorova pohledu se výukovou videosekvencí rozumí obrazový a zvukový záznam situace, která se váže k probíraným očekávaným výstupům. Videosekvence se liší od hraného filmu, případně výukového filmu v tom, že často nemá scénář, je technicky hůře zpracovaná je tvořena povětšinou amatéry. Pro natočení videosekvencí stačí pouze základní vybavení.

Výuková videa začal autor na stránkách školy publikovat v roce 2009. Tato videa jsou určena jeho žákům a odráží aktuální obsah výuky předmětu. Jelikož autor učí předmět Matematicko fyzikální

základ, jedná se o videa ukazující základní postupy při řešení typových početních úloh a dále o vysvětlení základních fyzikálních principů.

Za 6 let práce na videonávodech bylo autorem publikováno 30 videonávodů, které pokrývají matematiku základní školy. Video byla prohlédnuta celkem více jak 150 000 krát s tím, že nyní autorův kanál na [www.youtube.com/adlif57](http://www.youtube.com/adlif57) sleduje cca 900 odběratelů.

Jedním z hlavních důvodů, proč autor svá videa dále tvoří jsou reakce jeho odběratelů. Pro účely článku byly vybrány všechny komentáře k videonávodu publikovanému před 14 měsíci. Videonávod se týkal podmínek řešitelnosti lomených výrazů a byl prvním, kde autor použil záznam obrazovky. Komentáře jsou mírně upraveny – (nevhodná slova jsou vytečkována). U dalších autorových videí jsou reakce velmi podobné.

Agent Marty před 2 měsíci

*este ze tu ste bez vas bych byl asi ted hodne v p....i :D*

WillGamesChannel

*Výborné video, učil jsem se z posledních chvil před zítřejším testem a kamarádi mi poradili vaše video a musím říct že to umíte vysvětlit jako málokdo.*

Václav Nádvořník

*Děkuji za pochvalu.*

taxik007

*skvělá videa, jen tak dál. Líp to vysvětlit nejde :))*

Václav Nádvořník

*Děkuji za pochvalu...*

WillGamesChannel

*A smím se zeptat jak to bude v případě např. : X-1 lomeno 5 anebo B lomeno Z ?*

Karolína Zavřelová

*Natocte prosim video kdyz podminka nema smysl a kdyz je hodnota daneho vyrazu rovna nule*

Václav Nádvořník

*Malá rada. Pro příklad typu kdy je výraz roven nule použijte stejný postup, jako zde, akorát pamatujte, že zlomek je roven nule, když je ČITATEL roven nule.*

*Takže napište čítele=0 a dál řešíte rovnici. Nejbližší video asi bude na toto téma, a nebo na vytýkání mínus jedničky. Uvidíme, na co bude čas.*

Karolína Zavřelová

*Děkuji*

Tomáš Kazda

*Výborný návod. Hned jsem to pochopil.*

SkiNy Nao EsOn

*Děkuji, moc jste mi pomohl! ;)*

Depok

*Děkuju, moc mi to pomohlo... Takhle je to lepší než s obličejem*

..

Marek Stonawski

*Nic proti vašemu obličejí, jste sexy chlap, ale tohle bylo přehlednější z videí co jsme od vás viděl.*

Jiří Petranek

*Děkuji Václavu Nádvorníkovi za tento návod*

Stefka022

*Mě více vyhovuje tabule, ale tohle je taky v pohodě :) Další pomoc, kterou jsem potřebovala. :) Děkuji.*

evca4

*Velmi dobře to vysvětlujete ,děkuji*

Špageer Vašíček

*zase ta kamera bohužel ale jinak díky vám sem dostal 1 z matematiky :D*

richto01

*Moooc děkuji za pomoc :)*

Václav Nádvorník

*rádo se stalo*

Black\_Jacck

*Interaktivní tabule je super :) Jen by me zajímalo, kde jste vzal tu jedničku :D x se nesmí rovnat 1 . To sem opravdu nepochopil :)*

Václav Nádvorník

*díky za uznání tabule. tu jedničku беру jako ironii*

Vaclav Janek

*Tato metoda je asi nejlepší, vše je čisté a člověk vidí jen to důležité :). Děkuji moc za to, že obětujete pár hodin svého času a vytvoříte tento návod.*

## 5. Postup práce při tvorbě videonávodů

V následující kapitole je uveden jeden z možných postupů, jak tvořit videonávody a určitá krizová místa jejich tvorby.

### Zvolení vhodného tématu

Ne každé téma je vhodné pro tvorbu videozáznamu. Je třeba volit takové úlohy, které jsou pro žáky problematické a obtížně pochopitelné, v nich se využije největší výhoda videonávodů - možnost zastavení a návratu v čase. Dále je vhodné takto zaznamenávat různé pokusy a jiné praktické úlohy, které žák může obtížně vypracovávat doma.

### Potřebné pomůcky

#### Videokamera nebo digitální fotoaparát

Na tvorbu záznamu stačí dnes běžný digitální fotoaparát, který umí zaznamenávat video i zvuk v dostatečném rozlišení. Jako rozlišení stačí 640 x 480 pixelů a rychlost snímání 30 snímků za minutu. Toto je zcela běžné ve většině dnešních fotoaparátů i mobilních telefonů. Samozřejmě natáčení ve FullHD kvalitě videa jen vylepší.

#### Stativ

Bylo třeba najít vhodný způsob upevnění fotoaparátu. Pro tvorbu záznamu je třeba mít "volné ruce" Je samozřejmě možné použít lepenky, či jiného způsobu upevnění záznamového zařízení. Další možností je místo stativu požádat o pomoc spolupracovníka, ale poté se kvalita záznamu zhorší - záznam bude roztržes. Investice do stativu není tak velká a pro další práci je vhodné jeho pořízení.

### Počítač a program na sestřihání videa

Pro úpravu videa může být používán například program iMovie od firmy Apple. Jedná se o program, který je snadno ovladatelný a umožňuje mimo jiné přímý export na kanál youtube.com. Tento program je zdarma dodáván ke každému počítači firmy Apple a k mobilním zařízením s iOS. V případě použití platformy Windows, je možné použít program moviemaker, který je součástí instalace Windows XP, případně Vista. Pro Windows 7 a 8 je tento program nutné doinstalovat ze služby live.com. Jako další možnost je použití jednoho z tisíců programů volně stažitelných ze sítě internet, případně software dodávaný společně s digitálním fotoaparátem, nebo kamerou. Volba programu je zcela libovolná.

### Natáčení záznamu

Při natáčení záznamu je vhodné dodržet několik zásad.

#### Připravení příkladů a jejich vyřešení předem

Jelikož běžný učitel není zvyklý stát před kamerou a do ní mluvit je třeba počítat s určitou nervozitou. Při řešení úloh se musí učitel více soustředit na jiné věci, než pouze na řešení úlohy. Dále je dobré volit vhodnou skladbu a náročnost úloh. Rovněž pokus je správné si připravit předem. Jestliže učitel při natáčení udělá chybu, je možné ji vystříhnout, ale tím se prodlužuje doba potřebná na zpracování videa.

#### Práce v klidu

Pro práci je třeba si vyhradit klidné místo, kde nebude učitel rušen a kde bude zajištěn klid. Jako vhodné prostředí může být i domácnost učitele, případně prázdná učebna v době mezi zvoněními.

#### Srozumitelnost výkladu

Při tvorbě videozáznamu je nutné dodržovat srozumitelnou a pomalou mluvu. Kamera zaznamená i drobné nepřesnosti v mluveném projevu. Srozumitelnost pak zajistí i lepší výsledek - porozumění žáky. Pomalé mluvení je poměrně obtížné, ale nutné.

#### Čitelnost rukopisu

Pro pochopení je nutné aby byl rukopis čitelný. Na toto je třeba se zaměřit i proto, že velikost písma je větší než normálně.

#### Čtení toho co píše

Právě pro horší čitelnost v malém rozlišení je vhodné při psaní textu tento zároveň i číst. Tímto se zapojí více mozkových center a tím je i větší šance na pochopení výkladu žáky.

#### Publikace v síti internet

Pro publikaci v síti Internet je možné použít v zásadě dva postupy. Prvním je přímé nahrání souboru na webové stránky s tím, že si student tato videa stáhne do počítače a poté přehraje. Na první pohled jednoduché řešení má ale mnoho zásadních nevýhod. První z nich je problém s kompatibilitou systémů a různých kodeků. Dále je tento systém náročný časově, neboť uživatel musí většinou čekat na kompletní stáhnutí souboru. Na druhou stranu je zde zajištěno plné zachování autorských práv. Protože však nevýhody převažují, je v běžné praxi častěji používáno streamování videí na specializovaných serverech. Celosvětově nepoužívanější je server youtube.com. Ten umožní většině uživatelů nahrát video téměř v jakémkoli formátu a také bez časového omezení. Poté jej uživatel může sledovat přímo na internetu. Formát youtube.com není závislý na platformě a je stejně funkční na mobilních telefonech, tabletech i stolních počítačích. Určitou nevýhodou je nutnost připojení k internetu po dobu sledování, ale i tento problém může být vyřešen vhodným programem. Druhou nevýhodou je, že publikací na serveru

youtube.com se vzdáváte části svých autorských práv. Přesto toto řešení je z mého pohledu řešením nejpraktičtějším. Streamovacích serverů je mnoho jiných, většina pracuje na podobném principu, ale youtube.com je jednoznačně nejrozšířenější. Pro publikaci videonávodů je vhodné si založit pouze pro tyto účely nový účet. Zde platí stejný princip jako při užívání sociálních sítí a emailovou komunikaci - oddělit pracovní a soukromý účet.

Dále je vhodné videozáznam sdílet na školní webové stránce, nebo v rámci výukového systému, který škola používá.

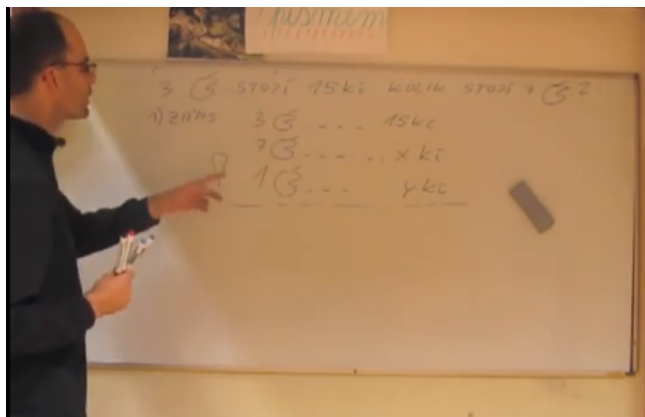
### Časová náročnost

Celková časová náročnost na vymyšlení, připravení natočení sestřihání a umístění výukového videozáznamu je přibližně 2 hodiny na 10 minutový záznam. Podle ohlasů žáků, rodičů i kolegů považují tento čas za vhodně vynaložený a pro žáky velmi přínosný.

### Zvolení technologie snímání

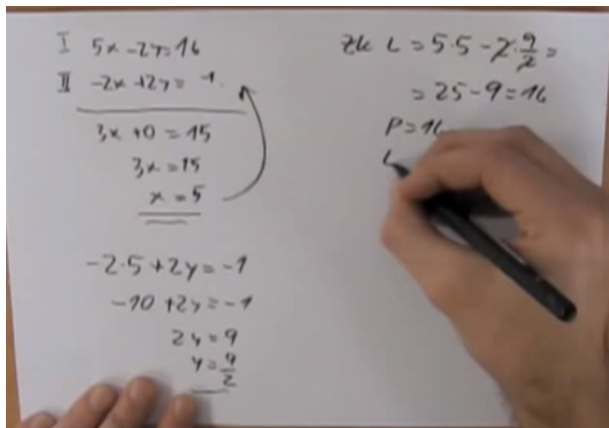
Pro videonávody lze použít mnoho různých metod snímání práce. V principu jsou možné následující metody:

Videozáznam školní tabule při psaní učitelem na ní. Zde je třeba si dát pozor na kvalitu záznamu a použité barvy a světlo. Některé fixy jsou špatně čitelné.[3]



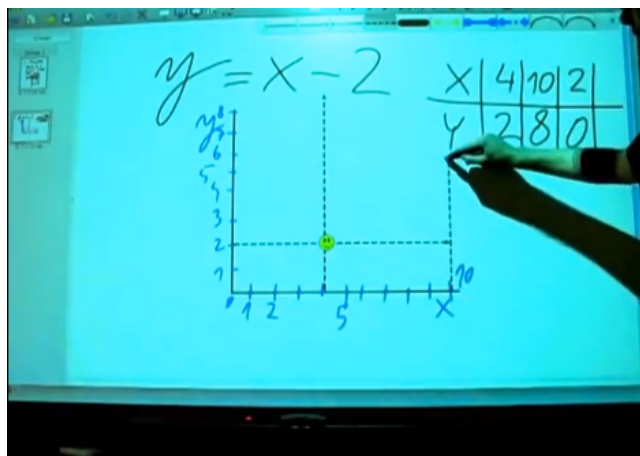
Obrázek 1 - Záznam výkladu u klasické tabule

Snímání papíru, na který učitel píše. Toto je nejjednodušší metoda, kde je třeba psát velmi čitelně a velkými písmeny. Zde se osvědčilo psaní tenkým černým fixem. Určitou nevýhodou je menší pohodlí při psaní.



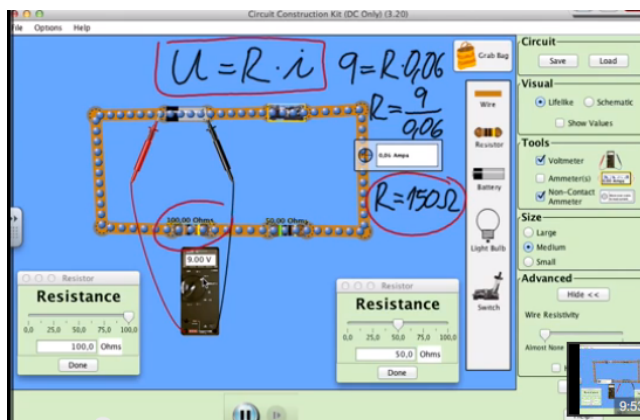
Obrázek 2 - snímání papíru s příkladem

Natáčení výuky učitele, který používá interaktivní tabuli.



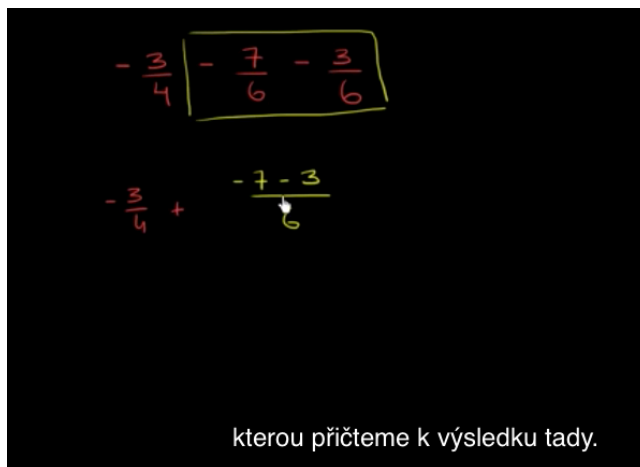
Obrázek 3 - záznam interaktivní tabule pomocí videokamery

Záznam obrazovky z interaktivní tabule, kdy učitel píše na tabuli a počítač zaznamenává veškeré činnosti na obrazovce.



Obrázek 4 - záznam obrazovky z počítače za použití interaktivní tabule

Záznam z obrazovky při použití grafického tabletu - metoda převážně používaná v Khan Academy.



Obrázek 5 - záznam obrazovky, psaní pomocí tabletu

Na základě vyzkoušení všech výše uvedených metod a reakcí v diskusích pod videi preferuje autor nyní snímání interaktivní tabule formou záznamu obrazovky. Je zde zachována možnost využití všech výhod počítače a také poměrně velká plocha pro

psaní učitelem. Tento způsob je pro učitele přirozenou činností, protože psaní na tabuli, oproti psaní na tabletu, je pro něj zcela běžné.

## 6. Statistické údaje autorova videokanálu youtube.com/adlif57

Při publikaci videí na serveru youtube.com lze pomocí integrovaných statistik vysledovat zajímavé souvislosti s návyky dnešních žáků a studentů. Zde je uvedeno několik čísel a dalších údajů. Stav údajů je ke dni 8.2.2015

*Celkový počet zhlédnutí* – 152 705

*Odhadovaný počet minut sledování* – 510 726 minut (cca 355 dní)

*Počet palců "líbí se"* – 1 256

*Počet palců "nelíbí se"* – 55

*Počet komentářů* – 493

*Počet odběratelů* – 851

*Nejúspěšnější videonávody*

Zlomky "násobení, dělení, rozšiřování, krácení" – 19 628 zhlédnutí

Sčítání zlomků – 13 073 zhlédnutí

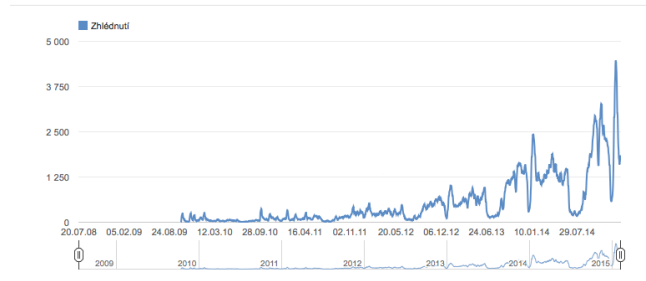
Sčítací metoda řešení soustav rovnic – 11 938 zhlédnutí

Rovnice s neznámou ve jmenovateli – 10 413 zhlédnutí

Výrazy – 9 762 zhlédnutí

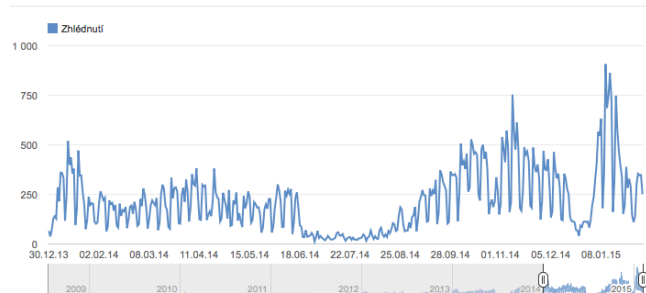
*Grafy sledovanosti videí*

První graf ukazuje, jakým způsobem rostla návštěvnost videokanálu. Při větším prozkoumání jsou zde patrné výchytky, které vznikají v průběhu prázdnin a hodnotících obdobích.



Obrázek 6 - počty zhlédnutí videí od začátku publikování

Na následujícím grafu je viditelná velká týdenní pravidelnost, která bude více specifikovaná v dalších grafech. Dále je zde vidět velký útlum sledování v průběhu hlavních prázdnin, kdy koncem srpna se počet zhlédnutí zvyšuje. Také v době vánočních prázdnin je sledování menší. Jako zajímavost je možné uvést 44 zhlédnutí videí 24. prosince 2014.



Obrázek 7 - počty zhlédnutí za rok 2014

V tomto grafu je možné vysledovat týdenní chování žáků a jejich přístup k učení. Pravidelně nejmenší počet zhlédnutí videí je v pátek, nebo v sobotu. Naopak nejvyšších čísel je dosaženo pravidelně v neděli. (Toto potvrzuje i výjimka 16.11.2014, kdy sice byla neděle, ale protože následoval státní svátek, byl počet zhlédnutí srovnatelný s běžnou sobotou a lokální maximum bylo dosaženo v pondělí. Po nedělní špičce následuje mírný pokles v pondělí, který pak v úterý nebo ve středu opět přechází v nárůst. Ke konci týdne opadá také zájem o videonávody.



Obrázek 8 - ukázka pravidelného chování návštěvníků videokanálu

## 7. Videokanály s výukovými videy v českém jazyce

Pro další orientaci v problematice autor doporučuje následující videokanály na serveru youtube.com.

**Khanova škola**

<https://www.youtube.com/user/KhanovaSkolaOS/videos>

**Masarykova ZŠ v Plzni**

<https://www.youtube.com/user/masarykovazs/videos>

**Marek Valášek**

<https://www.youtube.com/channel/UCLfEwNGlcQBpDiWbWnVaOPg>

**Václav Nádvorník**

<https://www.youtube.com/user/adlif57/videos>

**Honza Vlček**

[https://www.youtube.com/channel/UC6OV3bc5FiAN379QO\\_xvMAg](https://www.youtube.com/channel/UC6OV3bc5FiAN379QO_xvMAg)

**Lenka Fabiánová**

[https://www.youtube.com/channel/UCaTLOatXIhXMcvL\\_YiFGSA](https://www.youtube.com/channel/UCaTLOatXIhXMcvL_YiFGSA)

**ZŠ MŠ Březová**

<https://www.youtube.com/user/jancepo>

## 8. Závěr

Tvorba výukových videozáznamů je vhodným doplňkem při výuce žáků. V žádném případě nenahradí výklad ve třídě, ale může žákům pomoci při domácí přípravě. Náročnost pro učitele není velká. Video zvládne vytvořit běžný uživatel počítače, kterých je mezi učiteli mnoho. Limitujícími faktory by mohla být časová náročnost, určitá nechuť učitelů a v některých ojedinělých případech chybějící materiální podmínky. Tato metoda je vhodná k použití v případě, že žáci mají přístup k internetu i doma.



## 9. BLIOGRAFICKÉ ODKAZY

- [1] NELEŠOVSKÁ, A. *Pedagogická komunikace v teorii a praxi*. 1. aktual.vyd. . Praha : Grada Publishing, 2005. 172 s. ISBN 80-247-0738-1.
- [2] Převrácená třída. *Převrácená třída* [online]. 2015 [cit. 2015-02-08]. Dostupné z: <http://www.prevracenatrida.cz>
- [3] NÁDVORNÍK, Václav. Václav Nádvorník - YouTube. *Youtube.com* [online]. Praha, 2009 - 2015 [cit. 2015-02-08]. Dostupné z: <https://www.youtube.com/user/adlif57>

# Metodika *Architecture First*

Rudolf Pecinovský  
Vysoká škola ekonomická, Praha  
Nám. Winstona Churchilla 4  
130 67 Praha 3  
Česká republika  
rudolf@pecinovsky.cz

## ABSTRAKT

Většina současných učebnic programování i vývojových prostředí se soustředí především na výuku algoritmizace a zápisu kódu. Současné trendy však ukazují, že tuto oblast zanedlouho pokryjí automatické generátory kódu. Zkušenost ale ukazuje, že studenti, kteří se naučí myslet v hladině kódu, pak mají problémy s přechodem na hladinu architektury. Tento problém se snaží řešit metodika výuky *Architecture First*, která prosazuje začít s výukou architektury s tím, že navržené programy vytváří generátor kódu. Pro podporu této koncepce bylo upraveno prostředí *BlueJ* do podoby označované *BlueJ++*, jejíž generátor kódu umožňuje výrazně prodloužit dobu, po níž můžeme při výuce zůstat v hladině architektury a ponechat návrh kódu na použitím generátoru.

## ABSTRACT

Majority of contemporary textbooks of programming as well as development environments are concentrated above all to teaching the algorithms and coding. However, the present trends show that soon this area will be covered by automatic code generators. The experience indicates that those students who learn thinking in the code level have problems with transition to the architecture level. The *Architecture First* methodology tries to solve this problem by creating the proposed programs by the code generator. The *BlueJ* environment has been modified to support this conception called *BlueJ++*. Its code generator enables to prolong significantly the time when we can keep the architecture level for teaching and leave the code proposal to the used generator.

## Klíčová slova

Architecture First, metodika výuky programování, BlueJ, OOP

## Keywords

Architecture First, BlueJ, OOP, teaching methodology.

## 1. ÚVOD

Současné kurzy programování nepříjemně často nerespektují změny, k nimž v programování za posledních 30 let došlo, a už vůbec nepředjímají změny, k nimž v následujících dvou dekadách dojde. Někde se změnilo použité programovací jazyky a tu a tam se změnilo i vyučované paradigma. (Bohužel, řada kurzů učí sice programovat v nových programovacích jazycích, avšak stále podle starších paradigmat.)

Většina současných kurzů a učebnic programování se v počátcích výkladu věnuje především výkladu syntaxe použitého programovacího jazyka, v následujících etapách pak výuce algoritmických konstrukcí. Studenti začátečníci se tak musejí často natolik soustředit na detaily vyvíjeného programu, že jim začne unikat hlavní obsah přednášené látky.

Studenti často končí předmět s představou, že pro úspěšné zvládnutí předmětu je nejdůležitější zvládnout syntaktická pravidla a klíčové knihovní objekty. Zkušenosti programátorů však vědí, že syntaxe není ani zajímavou, ani obtížnou součástí programování

Někteří učitelé proto na počátek výuky zařazují speciální výukové programovací nástroje, které od studentů vyžadují pouze minimální znalosti syntaxe. Z těch známějších bych zde jmenoval nástroje Alice, Baltík a Scratch.

Tyto nástroje však pouze osvobozují od problémů se syntaxí (přesněji řečeno výrazně ji zjednodušují). Výuka se však nadále soustředí na probrání základních algoritmických konstrukcí a naprosto ignoruje současné trendy ve vývoji programování a jeho paradigmat.

## 1.1 Ohlédnutí do historie

Od svého počátku se programování vyvíjí tak, že se postupně automatizují další a další činnosti potřebné pro vývoj programu, takže se neustále zmenšuje sémantická mezera mezi způsobem, jak zákazník zadává požadavky, a způsobem, jak programátor tyto požadavky realizuje. Tím se současně zvyšuje hladina abstrakce, s jakou přistupujeme ke schopnostem počítače.

Nejprve bylo nutno programovat přímo ve strojovém kódu. Záhy se objevily jazyky symbolických instrukcí, resp. jazyky symbolických adres, které osvobodily programátory od nutnosti pamatovat si kódy jednotlivých instrukcí a adresy jednotlivých součástí programu.

Následně se prosadily vyšší programovací jazyky, které přiblížily zápis programu způsobu vyjadřování vědců a techniků. Paralelně se neustále zdokonalovaly knihovny, které programátorům umožňovaly sdílet vyvinutý kód a nemuset opakovaně vytvářet jeho často používané součásti.

V další etapě se začaly objevovat nejrůznější frameworky, které dále osvobodily programátory od vytváření řady nutných součástí, přičemž návrh a vývoj mnohých vyžadoval vysokou odbornost a zkušenost.

Paralelně se rodila nová paradigmatata umožňující více abstrahovat od způsobu práce počítače a přiblížit formulaci řešení daného problému formulaci vlastního problému. Formulace řešení se stále více přesouvá od imperativního stylu, při němž vysvětlujeme počítači **jak** má zadaný problém řešit, k deklarativnímu stylu, při němž počítači vysvětlujeme, **co** má řešit.

Již delší dobu se postupně rozšiřuje použití nejrůznějších (pověšinou grafických) nástrojů, které umožňují zadávat požadavky v podobě přístupné i osobám, které jsou sice odborníky ve své oblasti, ale s klasickým programováním nechtějí mít nic společného. Tyto nástroje pak vygenerují potřebný kód automaticky.

## 1.2 Pohled do budoucnosti

Výše zmíněné generátory kódu jsou schopny vytvořit potřebný program na základě čím dál abstraktnějších zadání. Nasazují se v oblastech, které jsou již dostatečně dobře zmapovány a bylo pro ně proto možno vytvořit algoritmy, které převádějí zadání odborníků na programy, které toto zadání řeší.

Počet a šíře záběru těchto oblastí se však neustále rozšiřuje a postupně se objevují v dalších a dalších oblastech. Těžiště potřeb-

ných znalostí vývojáře se tak postupně přesouvá z oblasti kódování do oblasti návrhu architektury.

Doba, kdy se programátoři už budou soustředit pouze na návrh architektury, protože veškeré kódování za ně přvezmou nejrůznější generátory kódu, není tak vzdálená. Vědci se shodují, že technologická singularita, což je okamžik, kdy stroj dokáže samostatně navrhnout a vytvořit jiný stroj, který bude dokonalejší než jeho tvůrce (tj. stroj, který jej vytvořil), nastane někdy mezi roky 2030 a 2045.

Odhaduje se, že v té době už programy budou schopny navrhovat jednodušší architekturu a budou se nadále zdokonalovat. Nedlouho potom pak vytlačí člověka i z oblasti návrhu architektury a začnou jej vytlačovat i z další oblasti, kterou je vyhledávání a specifikace problémů, které se mají vyřešit.

### 1.3 Co na to výuka?

Výuka programování a její metodika se bohužel na většině škol zastavila v době, která předcházela revolučním změnám odstartovaným na přelomu století. Studenti se nadále učí podle „osvědčených postupů“, které sice měly své opodstatnění na konci minulého století, ale v současné době ve studentech pouze zakotvují návyky, kterých se budou muset později zbavovat.

Vedoucí programátorských týmů, do nichž absolventi takovéhoto kurzů nastupují, si pak houfně stěžují, že školy opouštějí kodéři, kteří možná umějí dobře zakódovat problémy řešené na různých programátorských soutěžích, ale nejsou schopni navrhnout architekturu složitějšího projektu.

Při návrhu koncepcí výuky a její metodiky bychom proto měli lépe vnímat současné trendy a připravovat studenty tak, aby zvládali požadavky praxe v době, kdy budou školy opouštět. Hovoříme-li proto o výuce studentů základních a středních škol, měli bychom se je pokusit vybavit znalostmi a dovednostmi, které budou nezbytně potřeba za 10 let. Přiznejme si však, že k tomu je nejprve musíme získat sami.

## 2. METODIKA ARCHITECTURE FIRST

Metodika *Architecture First* staví na předpokladu, že v době, kdy budou naši studenti nastupovat do praxe, od nich bude vyžadována především schopnost navrhnout architekturu vyvíjeného programu. Schopnost návrhu architektury by proto pro ně měla být naprosto zásadní.

Pedagogická zásada *Ranní ptáče* říká, že to, co považujeme ve výuce za nejdůležitější, bychom měli umístit na začátek kurzu (nebo alespoň co nejbliže začátku), aby studenti měli dostatek času a příležitosti si tyto důležité principy osvojit.

### 2.1 První etapa

Protože se domníváme, že nejdůležitější dovedností, kterou si naši studenti musejí osvojit, je návrh architektury, začínáme s výukou základních architektonických principů a jejich použití.

S výukou základních architektonických principů začíná řada kurzů, ale většinou je to na počátku pouze povídání o programování, při němž studenti nemají možnost si použití některých principů vyzkoušet v praxi.

Základním rozdílem metodiky *Architecture First* je to, že studenti mají možnost si tyto konstrukce a principy zkusit od samého začátku kurzu. Studenti vždy navrhnou nějakou architektonickou konstrukci vhodnou pro řešení zadané úlohy a generátor kódu zabudovaný v použitém vývojovém prostředí pak vše zakóduje. Studenti si pak mohou vzápětí vyzkoušet, jak navržená konstrukce funguje.

Délka první etapy je úměrná schopnostem použitého generátoru kódu – v optimálním případě celý semestr. Protože se tato etapa vůbec nezabývá zdrojovým kódem a ze standardní knihovny používá jenom základní všudypřítomné minimum (třídy **Object** a **String**), lze po něm v druhé etapě pokračovat výukou některého ze široké škály objektově orientovaných jazyků a platforem.

### 2.2 Druhá etapa

V druhé etapě se vrátíme na počátek výuky. Znovu si projedeme jednotlivé programy a ukážeme si, jak je v předchozí etapě generátor kódu zakódoval. Při té příležitosti si studenti osvojí syntaxi použitého jazyka, jejíž znalost budou potřebovat ve třetí etapě.

### 2.3 Třetí etapa

Ve třetí etapě navážeme výkladem programových konstrukcí, které jsou mimo rámec schopností aktuálně použitého generátoru kódu. Látka probíraná ve třetí etapě proto závisí na kvalitě a schopnostech použitého generátoru kódu.

## 3. CO JE NÁVRH ARCHITEKTURY

Když řekneme, že by výuka programování měla začít výukou návrhu architektury programu, většina učitelů si pomyslí, že něco takového lze realizovat pouze v univerzitních kurzech, ale určitě ne na střední či dokonce základní škole. Opak je ale pravdou.

### 3.1 Postup klasicky školeného programátora

Jak už bylo řečeno výše, klasický způsob výuky programování se soustředí především na výuku algoritmizace, tj. popisu, **jak** zadanou úlohu řešit. Pro absolventi takovéhoto kurzů je řešením úkolu vytvoření kódu, který daný úkol vyřeší. Když si od zákazníka vyslechnou zadání úkolu, začnou ihned přemýšlet nad tím, jak by jej zakódovali.

Základní nevýhodou takovéhoho přístupu je to, že programátor okamžitě zmizí z mentálního obzoru zákazníka, který se v jeho kódu vůbec nevyzná a nemůže mu proto poskytnout jakoukoliv zpětnou vazbu do doby, než mu programátor předvede nějaké (byť provizorní) řešení.

### 3.2 Postup bližší obyčejnému člověku

Když má obyčejný člověk vyřešit nějakou úlohu, nezabývá se většinou detaily, které by odpovídaly kódu. Běžnému člověku bude bližší následující postup:

1. Nejprve si ujasníme účastníky řešení, tj. s kým a s čím můžeme, resp. musíme při řešení úkolu počítat. Potřebujeme-li např. pověsit obraz, mohou být účastníky daný obraz, skobička, kladivo a štafle.
2. Jsou-li jasní účastníci, měli bychom si ujasnit jejich schopnosti, abychom věděli, co od nich můžeme očekávat, resp. co od nich můžeme požadovat.
3. V následující etapě bychom si měli upřesnit i jejich klíčové vlastnosti, zejména pak ty, které můžeme (nebo potřebujeme) ovlivnit.
4. V závěrečné etapě začneme uvažovat nad tím, jak s pomocí výše zmíněných účastníků danou úlohu realizovat – naprogramovat.

Bude-li požadovaným řešením návrh vhodného programu, bude výše popsaný postup vlastně postupem návrhu architektury tohoto programu. Naučíme-li žáky řešit problémy tímto postupem, budou umět hned od počátku řešit mnohem složitější úlohy, než tomu bylo v době, kdy jsme při výkladu sice hovořili o nutnosti dekompozice a přístupu shora dolů, ale v praxi jsme většinu studentů naučili soustředit se především na detail.

Další výhodou tohoto přístupu je to, že mnohem déle zůstáváme v oblasti, v níž se zákazník dokáže orientovat a může nám proto poskytnout zpětnou vazbu nutnou pro případné korekce. Takovýto přístup tedy zákonitě vede k dalšímu zmenšování sémantické mezery a z toho plynoucí vyšší kvalitě vyvinutých programů.

#### 4. VÝVOJOVÉ PROSTŘEDÍ BLUEJ

V našich kurzech jsme začali s používáním vývojové prostředí *BlueJ*. To má pro výuku ve vstupních kurzech programování několik výhod:

- Je maximálně jednoduché, takže studenti nezápasí se zvládnutím vývojového prostředí a mohou se soustředit na vlastní programování.
- Nutí studenty začínat ve svých úvahách na hladině architektury, protože základním zobrazením vyvíjeného programu je diagram tříd.
- Nabízí zabudovaný generátor kódu, který okamžitě zanáší změny architektury z diagramu tříd do kódu a naopak.
- Umožňuje pracovat v interaktivním režimu, při němž studenti vystupují jako jeden z objektů programu, který komunikuje s ostatními objekty. Přitom předvádějí, co se má řešit, a zabudovaný generátor kódu následně vytvoří program realizující tuto demonstraci.
- Díky použité koncepci vede studenty přirozeně k tomu, že i třída je objekt, jenom má oproti objektům, které jsou instancemi některé třídy, jisté zvláštní vlastnosti.

Bohužel, schopnosti zabudovaného generátoru kódu nejsou příliš velké, protože prostředí bylo původně navrženo pro výuku podle metodiky *Object First*, která se mnohem dříve soustředí na tvorbu kódu, a dokonalejší generátor kódu proto nevyžaduje. Když proto autoři prostředí zveřejnili toto prostředí jako *open source*, začali jsme je upravovat, aby lépe vyhovovalo potřebám metodiky *Architecture First*.

#### 5. VÝVOJOVÉ PROSTŘEDÍ BLUEJ++

Upravenou verzi vývojového prostředí označujeme *BlueJ++*. Hlavní realizované úpravy jsou následující:

- *BlueJ++* ruší omezení na práci v jediném balíčku a umožňuje efektivně definovat komunikaci mezi třídami v různých balíčcích.
- V interaktivním režimu je možno definovat nejenom testovací třídy, ale libovolné třídy, interfejsy a výčtové typy.
- Ve třídě je možné interaktivně definovat její konstruktor i běžnou metodu.
- Místní nabídka objektů (tříd i jejich instancí) zobrazuje nejenom metody daného objektu, ale i jeho atributy.
- Prostředí umožňuje přímo měnit hodnoty viditelných proměnných atributů.
- Umožňuje definovat jak členy třídy, tak členy jejích instancí.
- Při vytváření atributů a lokálních proměnných umožňuje deklarovat jejich typ a nevnučuje uživateli skutečný typ přiřazované hodnoty.
- Umožňuje definovat lambda-výrazy.
- Zobrazuje kopenogram označené metody.

- Perspektivně bude umožňovat měnit definici metody v průběhu jejího provádění, přesněji vyjímát z definice ještě neprovedené příkazy a vkládat do definice příkazy nové. Studenti tak nebudou muset (na rozdíl od současného stavu) definovat chybné metody znovu od počátku, ale budou je moci spustit v interaktivním režimu a v průběhu jejich vykonávání upravit jejich definici.

#### 6. VÝSLEDKY APLIKACE METODIKY ARCHITECTURE FIRST

Metodika *Architecture First* byla testována paralelně v zájmových útvarech dětí základních a středních škol a v univerzitních kurzech programování. Zkušenosti s její aplikací v zájmových kurzech byly rozebrány ve [24]. V dalším textu se proto soustředím na zkušenosti z její aplikace na VŠE.

Zde byla metodiky použita u několika skupin studentů. Tito studenti měli v následujícím semestru předmět *Základy softwarového inženýrství*, v němž se ve skupinách setkali se studenty, kteří prošli klasicky koncipovanou výukou, a v němž mimo jiné navrhovali poněkud rozsáhlejší projekty, na nichž pak pracovali ve skupinách.

Po druhém semestru byl všem absolventům předmětu *Základy softwarového inženýrství* předložen anonymní webový dotazník, v němž měli odpovědět na otázku: „Domníváte se, že výuka podle metodiky *Architecture First* pomůže studentům v lepším chápání vazby mezi vyvíjeným programem a simulovanou skutečností a z toho plynoucích lepších schopností v návrhu odpovídající architektury?“

Dotazník vyplnilo 59 studentů, z toho 53 studentů, kteří v prvním semestru absolvovali výuku podle metodiky *Architecture First* a 6 studentů absolvovalých klasicky orientovanou výukou. Rozložení jejich odpovědí naleznete v tabulce

	<i>Arch.First</i>	Standard	Průměr
Výrazně méně	0%	0%	0%
Poněkud méně	0%	0%	0%
Zhruba stejně	11%	17%	12%
Poněkud více	62%	67%	63%
Výrazně více	17%	17%	17%
Neumím posoudit	9%	0%	8%

Z tabulky vyplývá, že 80 % studentů se domnívá, že použití metodiky *Architecture First* má pozitivní dopad na jejich schopnosti analyzovat problém a navrhnout architekturu požadovaného řešení.

#### 7. DALŠÍ VÝVOJ

Metodika *Architecture First* byla vyzkoušena v zájmových kroužcích studentů základních a středních škol ve věku 11 až 14 let. Ukazuje se, že děti nemají s akceptací objektově orientovaného paradigmatu problémy. Problémy se objevují pouze u studentů, kteří před příchodem do kurzu prošli nějakým klasicky orientovaným kurzem, který už jejich přirozený způsob uvažování „ohnul“.

Drobnou nevýhodou těchto kurzů je skutečnost, že základní koncepce prostředí *BlueJ++* vyžaduje od studentů jistou schopnost abstrakce, a ta toto prostředí předurčuje k použití ve vysokoškol-

ských kurzech a případně ve vyšších ročnících středních škol. Pro nasazení v nižších stupních škol mu chybí lepší provázanost s obrazem modelované skutečnosti.

Tu na druhou stranu nabízí systém Scratch, který se v poslední době ve školách stále více prosazuje. Jeho další výhodou je jeho přirozená objektová orientovanost a neméně přirozená tvorba událostmi řízených aplikací. Jeho nevýhodou je ale přílišné soustředění na kód a neschopnost zobrazení architektury vyvíjeného projektu, takže při tvorbě větších projektů studenti ve svých projektech bloudí a občas se v nich zcela ztrácejí.

Obdobně je na tom i systém *Alice*, který má také svoji skupinu zastánců a který nabízí podobné možnosti jako *Scratch*. Bohužel, tento systém má také podobné nevýhody, především pak přílišnou koncentrovanost na kód.

Dalším systémem se zajímavými vlastnostmi uplatnitelnými zejména při nasazování v nižších ročnících škol, je systém *Baltie*.

Ukazuje se proto, že by v dalším výzkumu bylo vhodné sloučit myšlenky výše zmíněných výukových nástrojů a pokusit se vytvořit takový, který by zahrnul jejich výhodné vlastnosti a eliminoval jejich nevýhody. Tím bychom získali nástroj, který by bylo možno efektivně použít i v základních a středních školách, a přitom by nám umožňoval aplikovat metodiky výuky, které se nesoustředí na výuku kódování, ale na návrh architektury projektu, protože starost o zakódování navrženého programu ponechávají na generátoru kódu, který je integrální součástí tohoto prostředí.

## 8. BIBLIOGRAFICKÉ ODKAZY

- [1] ACM/IEEE-CS Joint Task Force on Computing Curricula. 2001. *ACM/IEEE Computing Curricula 2001 Final Report*. <http://www.acm.org/sigse/cc2001>.
- [2] *Computer Science Curricula 2013 – Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. <http://www.acm.org/education/CS2013-final-report.pdf>
- [3] Barnes, D. Kölling, M. 2012. *Objects First with Java: A Practical Introduction Using BlueJ (5th Edition)*. Pearson. ISBN 978-0-13-249266-9
- [4] Bergin, J. 2000. Fourteen Pedagogical Patterns. *Proceedings of Fifth European Conference on Pattern Languages of Programs*. (EuroPLoP™ 2000) Irsee 2000. [Bergin 2000]
- [5] Bergin, J. 2012. *Pedagogical Patterns: Advice For Educators*. CreateSpace Independent Publishing Platform. ISBN 1-4791-7182-4. [Bergin 2012]
- [6] Cooper S. 2010. The Design of Alice. *ACM Transactions on Computing Education*. 10, 4, Article 15 (November 2010), 16 pages. DOI=10.1145/1868358.1868362 <http://doi.acm.org/10.1145/1868358.1868362> [Cooper 2010]
- [7] Dann, W., Cosgrove, D., Slater, D., Culyba, D. and Cooper, S. 2012. Mediated transfer: Alice 3 to Java. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12)*. ACM, New York, NY, USA, 141-146. DOI=10.1145/2157136.2157180 <http://doi.acm.org/10.1145/2157136.2157180> [Dann at al. 2012]
- [8] Dehnadi. S. 2009. *A Cognitive Study of Learning to Program in Introductory Programming Courses*. PhD thesis, Middlesex University [Dehnadi 2009]
- [9] Kofránek J., Pecinovský R., Novák P.: Kopenograms – Graphical Language for Structured Algorithms. *Proceedings of the 2012 International Conference on Foundation of Computer Science. WorldComp 2012 Las Vegas*. CSREA Press. ISBN 1-601-32211-9. [Kofránek at al. 2012]
- [10] Kölling, M. 1999. Teaching Object Orientation with the Blue Environment. *Journal of Object-Oriented Programming*, 12(2): 14–23. <http://www.BlueJ.org/papers/1999-11-JOOP4-blue-env.pdf>. [Kölling 1999]
- [11] Michael Kölling and John Rosenberg. 2001. Guidelines for teaching object orientation with Java. In *Proceedings of the 6th annual conference on Innovation and technology in computer science education (ITiCSE '01)*. ACM, New York, NY, USA, 33-36. DOI=10.1145/377435.377461 <http://doi.acm.org/10.1145/377435.377461> [Kölling and Rosenberg 2001]
- [12] Kölling, M. 2010. The Greenfoot Programming Environment. *ACM Transactions on Computing Education*. 10, 4, Article 14 (November 2010), 21 pages. DOI=10.1145/1868358.1868361 <http://doi.acm.org/10.1145/1868358.1868361> [Kölling 2010]
- [13] Kölling, M. 2014. Interactive Objects with BlueJ. *Java Magazine May/June 2014*. [http://www.oraclejavamagazine-digital.com/javamagazine/may\\_june\\_2014#pg34](http://www.oraclejavamagazine-digital.com/javamagazine/may_june_2014#pg34) [Kölling 2014]
- [14] Krejčí, J. 2014. *Požadavky podniků na profil absolventa – programátora: bakalářská práce*. (Employers demands on profiles of graduate programmers: Bachelor thesis) University of Economics Prague, Faculty of Informatics and Statistics. Department of Information Technologies. [Krejčí 2014]
- [15] LaLonde, W., Pugh, J. 1991. Smalltalk: subclassing subtyping is-a. *Journal of Object Oriented Programming*. 3, 5 (January 1991), 57-62. [LaLonde and Pugh 1991]
- [16] Liberman, N., Beeri, C. and Kolikant Y. B. D. 2011. Difficulties in Learning Inheritance and Polymorphism. *ACM Transactions on Computing Education*. 11, 1, Article 4 (February 2011), 23 pages. DOI=1921607.1921611 <http://doi.acm.org/1921607.1921611>. [Liberman at al. 2011]
- [17] Loftus, Ch. Thomas, L., Zander, C. 2011. Can graduating students design: revisited. In *Proceedings of the 42nd ACM technical symposium on Computer science education (SIGCSE '11)*. ACM, New York, NY, USA, 105-110. DOI=10.1145/1953163.1953199 <http://doi.acm.org/10.1145/1953163.1953199> [Loftus at al. 2011]
- [18] Malan, D. J. and Leitner, H. H. 2007. Scratch for budding computer scientists. In *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education (SIGCSE'07)*. 223–227. [Malan and Leitner 2007]
- [19] Matviichuk O. 2014. *Doplňení vývojového prostředí BlueJ o funkce využitelné v úvodních kurzech programování: diplom*



*mová práce*. (Enhancing of *BlueJ* IDE with functionality applicable in introduction courses of programming: Master thesis) University of Economics Prague, Faculty of Informatics and Statistics. Department of Information Technologies. [Matviichuk 2014]

- [20] Pecinovský R. 2000. *Baltie – The Programming TEXTBOOK not only for children*. SGP, Uherské Hradiště. <http://www.sgpsys.com/en>. [Pecinovský 2000]
- [21] Pecinovský R, Pavlíčková J., Pavlíček L. 2006. Let's modify the objects-first approach into design patterns first, *Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education*, pages 188–192. [Pecinovský at al. 2006]
- [22] Pecinovský, R., Kofránek, J. 2012. How to improve understanding of OOP constructs. Wroclaw 09.09.2012 – 12.09.2012. In: *Science Education Research Conference*. Wroclaw: PTI, 2012, s. 19–24. <http://fedcsis.org/proceedings/fedcsis2012/pliks/136.pdf> [Pecinovský and Kofránek 2012]
- [23] Pecinovský, R. 2013. *OOP – Learn Object Oriented Thinking & Programming*. Eva & Tomas Bruckner Publishing. ISBN 978-80-904661-8-0. <http://pub.bruckner.cz/titles/oop> [Pecinovský 2013]
- [24] Pecinovský, R., Kofránek, J. 2013. The Experience with After-School Teaching of Programming for Parents and Their Children. In *Proceeding of the 2013 International Conference on Frontiers in Education: Computer Science and Computer Engineering*. (FECS'13) Las Vegas. [http://edu.pecinovsky.cz/papers/2013\\_WC\\_Our\\_Experience\\_Teaching\\_After-School\\_Programming.pdf](http://edu.pecinovsky.cz/papers/2013_WC_Our_Experience_Teaching_After-School_Programming.pdf) [Pecinovský and Kofránek 2013]
- [25] Pecinovský R.: *Java 8 – Úvod do objektové architektury pro mírně pokročilé*. Praha, Grada Publishing, 2014. ISBN 978-80-247-4638-8. (*Java 8 – Object Oriented Architecture for the Pre-Intermediate Programmers*) [Pecinovský 2014]

# e-Skills for Jobs ve vzdělávání na Univerzitě J. E. Purkyně

Pavel PEŠAT

Univerzita J. E. Purkyně v Ústí nad Labem

Hoření 13

400 96, Ústí nad Labem

Česká republika

pavel.pesat@gmail.com

## ABSTRAKT

V roce 2014 proběhl třetí ročník celoevropské kampaně e-Skills for Jobs 2014<sup>1</sup>, do které se jakožto partner zapojila také Pedagogická fakulta Univerzity J. E. Purkyně v Ústí nad Labem. Součástí kampaně byl také IT Fitness Test 2014<sup>2</sup>, kterého se v měsících září a října 2014 zúčastnilo celkem 140 studentů denních i kombinovaných studijních programů sociální pedagogika, pedagogika volného času a sociálně pedagogická asistence. V příspěvku je popsána metodika testování a výsledky, kterých studenti UJEP dosáhli. Výsledky respondentů z UJEP jsou porovnány s údaji uvedenými v globální závěrečné zprávě IT Fitness test 2014 – výsledky, kterou publikoval Dům zahraniční spolupráce MŠMT České republiky.

## ABSTRACT

The third annual European campaign e-Skills for Jobs 2014 took place in the Czech Republic last year. The Faculty of Education, University of J. E. Purkyně in Ústí nad Labem (UJEP), joined the campaign as a partner. The IT Fitness Test 2014 was one of the important campaign activities. There were tested 140 students enrolled in the full time and combined bachelor study programmes social pedagogy, leisure time pedagogy and socio-pedagogical assistance in September and October 2014. The paper describes the methodology and results of testing achieved by UJEP students. Results from UJEP are compared with the survey data specified in the Final Report of the IT Fitness Test 2014 published by the Centre for International Cooperation in Education (DZS) established by the Ministry of Education, Youth and Sport of the Czech Republic.

## Klíčová slova

E-skills, IT Fitness test, digitální kompetence

## Keywords

E-skills, IT Fitness test, digital competence

## 1. ÚVOD

Znalosti a dovednosti v oblasti informačních a komunikačních technologií (dále jen ICT) bezpochyby patří mezi důležité kompetence lidí v moderní informační společnosti. V České republice tyto kompetence nejsou zařazeny mezi tzv. klíčové kompetence jmenovitě uváděné v Rámcových vzdělávacích programech, a to ani na základní, ani na středoškolské úrovni. ICT však ovlivňují většinu oblastí lidské činnosti a znalosti a dovednosti v oblasti ICT se jeví jako nezbytné pro uplatnění jak na trhu práce, tak i pro seberealizaci v soukromém životě. Trend v této oblasti je jasný – absolventi škol budou žít v informační

společnosti plně digitálních technologií a bez jejich funkčního zvládnutí se nebudou moci plně zapojit do života společnosti. Již dnes můžeme hovořit o tzv. digitálním znevýhodnění, při kterém jedinec nemůže nebo neumí smysluplně a efektivně využívat ICT při práci i v běžném životě. Člověk vybavený pouze „klasickým vzděláním“ (tj. před vstupem ICT do vzdělávání), i kdyby bylo sebedokonalejší, nebude mít šanci se plnohodnotně uplatnit v digitálním světě [1].

Věrohodné údaje o úrovni kompetencí v oblasti ICT jsou nezbytné pro posouzení aktuálního stavu uživatelů i pro formulaci budoucích cílů. Hodnocení znalostí a dovedností uživatelů v oblasti ICT je však všeobecně problematickou záležitostí a různá šetření vedou často k rozporným závěrům. Stejně problematické jsou závěry šetření zaměřených na hodnocení ICT kompetencí učitelů. Z nedávné doby existují studie, jejichž závěry jsou značně kritické a indikují nízkou úroveň funkčních ICT znalostí a dovedností pedagogické veřejnosti. Příkladem může být zjišťování *ICT dovedností u studentů učitelství*, které bylo realizováno na Pedagogické fakultě TU v Liberci v r. 2011. Pomocí testu informační gramotnosti TIGR fy SCIO bylo testováno 56 studentů prvního ročníku kombinovaného bakalářského studijního programu *pedagogika volného času* (27) a kombinovaného magisterského studijního programu *učitelství pro 1. stupeň základní školy* (29), tj. studentů, kteří již působili jako učitelé a doplňovali si potřebné vzdělání. Ukázalo se, že v „teoretické“ části testu dosáhli studenti kombinovaných studijních programů učitelství výsledků srovnatelných s výsledky studentů víceletých gymnázií ve věku odpovídajícím ukončení výuky ICT na ZŠ, tj. tercie a kvarty víceletého gymnázia. V „praktické“ části testu se však ukázalo, že v obou testovaných skupinách existují statisticky významné skupiny osob, jejichž skóre B bylo rovno nule, tj. tyto osoby neprokázaly základní praktické dovednosti a bylo by možné je označit za osoby bez *funkční ICT gramotnosti*. Počet těchto osob činil ve zkoumaných vzorcích cca 15 %. Současně se ukázalo, že v obou testovaných skupinách existují také statisticky významné skupiny osob, jejichž skóre B bylo větší než 30 (lepší než 90 % škály), tj. tyto osoby prokázaly výborné základní praktické dovednosti a bylo by možné je označit za osoby *funkčně ICT gramotné*. Počet těchto osob činil ve zkoumaných vzorcích cca 10 % (studenti primárního vzdělávání) a 15 % (studenti pedagogiky volného času) [2].

ICT kompetencemi učitelů informaticky orientovaných předmětů se zabývalo také rozsáhlé šetření *Rozvoj informačních technologických kompetencí na základních školách* realizované v letech 2012–2013 řešiteli z Pedagogické fakulty UK v Praze. Toto šetření bylo zaměřeno na částečně odlišnou cílovou skupinu učitelů (učitelé informatických předmětů) a pro hodnocení kompetencí učitelů při něm byly použity autoevaluační a evaluační deklarace, přesto jeho výsledky přispívají k dokreslení situace. Autoři konstatují, že deklarovaná a ověřená úroveň ICT gramotnosti učitelů informatiky se liší pouze minimálně. Autoevaluační přístup hodnotí jako dostačující aproximaci

<sup>1</sup> *e-Skills for Jobs 2014* je název kampaně EU, který se v České republice používá v původním anglickém znění.

<sup>2</sup> *IT Fitness Test 2014* je název testu IT dovedností, který se v České republice používá v původním anglickém znění.

skutečných ICT kompetencí, neboť dle výzkumů se respondenti se nesnaží své kompetence nadhodnocovat. Míry centrální tendence (modus, medián) aktuální úrovně ICT kompetencí učitelů informatických předmětů zjištěné při šetření odpovídají pátému stupni škály, který je charakterizován jako schopnost upravovat či vytvářet výukové materiály, přizpůsobit aplikace žákům a využít tyto dovednosti k podpoře výuky. Nejvyšší nabízená úroveň (st. 6) je přisouzena pouze 29 % učitelů. Jedná se o úroveň, která předpokládá nejen výběr a tvořivé využití technologií, ale i schopnost adaptace na nové technologie a zejména vedení žáků k používání pokročilých aplikací a tvorbě vlastních materiálů. Rozložení voleb ujištěné ve frekvenčních tabulkách dále ukázalo i nezanedbatelné obsazení nižších stupňů škály (2–4), které u učitelů informatických předmětů není možné považovat za dostatečné [3].

Překvapivé výsledky přineslo mezinárodní šetření *Počítačové a informační gramotnosti českých žáků ICILS 2013*. V šetření ICILS byly stanoveny čtyři dovednostní úrovně počítačové a informační gramotnosti. Nejvyšší, čtvrté dovednostní úrovně, na niž žáci disponují pokročilými znalostmi a dovednostmi, dosáhla 3 % českých žáků, což řadí Českou republiku na 6. místo. V České republice je klíčový podíl žáků na třetí dovednostní úrovni (34 %), který je nejvyšší ve srovnání s ostatními zúčastněnými zeměmi. Zároveň byl zjištěn vůbec nejnižší podíl žáků, kteří nedosáhli první úrovně (2 %). Největší podíl žáků v České republice tvoří žáci na druhé dovednostní úrovni (48 %). V národní zprávě se uvádí, že žáci České republiky dosáhli v testu počítačové a informační gramotnosti nejlepšího výsledku ze všech zúčastněných zemí s hodnotou 553 bodů. Bylo také zjištěno, že 66 % učitelů používá počítač při výuce minimálně jednou týdně (z toho 27 % každý den), což je nadprůměrný údaj. Z hlediska profesního vzdělávání učitelů jsou zjištěny ICILS dávány do souvislosti se zjištěními z mezinárodního šetření o vyučování a učení TALIS 2013. V TALIS bylo zjištěno, že nejvíce se učitelé v ČR účastní profesního vzdělávání zaměřeného na faktické znalosti a vědomosti v předmětech a hned na druhém místě profesního vzdělávání zaměřeného na ICT dovednosti potřebné pro výuku. Rovněž se ukázalo, že učitelé pociťují velkou potřebu profesního vzdělávání zejména v oblasti ICT dovedností potřebných pro výuku. V závěrečné zprávě je konstatováno, že cit.: „V každém případě závěry z obou šetření ICILS a TALIS podporují myšlenku, že má smysl věnovat více pozornosti dalšímu vzdělávání pedagogů v oblasti využívání ICT nástrojů při výuce a vzdělávání. Rovněž by bylo vhodné požadovat v rámci studijních programů fakult připravujících budoucí učitele, aby si každý absolvent osvojil alespoň určitou minimální úroveň schopností a dovedností využívat ICT nástroje při výuce a vzdělávání.“ [4].

Autoevaluační postupy byly použity také v pilotním šetření zaměřeném na využití ICT ve vzdělávání žáků se speciálními vzdělávacími potřebami, které bylo realizováno v Moravskoslezském kraji. Na základě zjištěných údajů bylo konstatováno, že přibližně tři čtvrtiny respondentů byly proškoleny v oblasti speciálně pedagogického využívání ICT, aktivně používají ICT ve výuce informatických i neinformatických předmětů, v přípravě na výuku a považují ICT za vhodnou učební pomůcku. Naopak přibližně jedna čtvrtina respondentů ICT ve své speciálně pedagogické praxi nevyužívala. Toto zjištění bylo celkem překvapivé a naznačuje pozitivní posun ve vztahu speciálních pedagogů k ICT, protože předchozí šetření spíše naznačovala opak, tzn. horší vztah speciálních pedagogů k ICT a malou míru využití ICT ve výuce. Ukazuje se také, že pedagogové aktivně spolupracují s Koordinátory ICT, kteří jim poskytují odbornou a metodickou ICT podporu. Pro speciálně pedagogické zařízení je

tedy výhodné, pokud má k dispozici odborníka s širším vzděláním v oblasti ICT [5].

Spektrum potřebných ICT znalostí a dovedností učitelů neinformatických předmětů je relativně široké. Je zřejmé, že témata, jejichž výuka vyžaduje specifické ICT znalosti a dovednosti, nelze bez těchto znalostí vyučovat. To se prokázalo u učitelů informatických předmětů např. ve výše zmíněném šetření Rozvoj informačně technologických kompetencí na základních školách, kde bylo zjištěno, že důležitá, odborně náročnější témata algoritmizace a základy programování a základy práce s databázemi respondenti neradi vyučují, nejsou příliš ochotni jejich výuku suplovat, preference těchto tematických celků jsou malé a jsou řazeny na konce časové osy výuky ICT předmětů [3]. ICT technologiemi používanými ve výuce speciálními pedagogy se zabýval Meier, který identifikoval 60 různých prostředků ICT používaných ve výuce a zjišťoval preference jejich používání učiteli vesměs neinformatických předmětů ve výuce žáků se speciálními vzdělávacími potřebami. Celkově nejvýše hodnoceným prostředkem ICT byly výukové programy pro žáky se SVP s průměrným hodnocením 8,83 na 10bodové škále. Na druhé pozici se umístily zdarma (legálně) dostupné programy pro žáky se SVP (7,95), následovaly interaktivní tabule (7,60) a stolní počítače (7,50). Naopak nejméně byly celkově hodnoceny sociální sítě typu Facebook (1,68), sociální sítě typu Twitter (1,80) a blogy (2,18). Mezi nejméně hodnocenými byly dále např. cloud computing a smartphony [6]. V zahraničí je situace v oblasti využívání ICT při vzdělávání dětí se speciálními vzdělávacími potřebami podobná. Lopúchová a Krajčí se zabývali výzkumem digitálních a asistenčních technologií v inkluzivní speciální pedagogice. Také oni použili autoevaluační dotazník, ve kterém měli učitelé hodnotit svou schopnost práce s počítačem. Na pětistupňové škále (1 je nejlepší) se stupněm 3 ohodnotilo nejvíce učitelů (54 %), 27 % si pak myslelo, že umí pracovat s počítačem velmi dobře (stupeň 2). Žádný z učitelů nepřiznal úplnou neznalost práce s počítačem, tj. stupeň 5 [7]. Podobně jako v ČR uvádějí relativně široké spektrum technologií a aplikací, které by měl speciální pedagog zvládnout. Všeobecně se ukazuje, že nízké preference mají i některá témata důležitá z hlediska edukace a bylo by žádoucí prohloubit ICT přípravu učitelů včetně speciálních pedagogů tak, aby se těmto tématům nevyhýbali z odborných důvodů.

12. listopadu 2014 schválila vláda České republiky *Strategii digitálního vzdělávání do roku 2020*, ve které se k roli učitelů uvádí následující, cit.: „Učitelé jsou bezesporu klíčovými aktéry, kteří implementují digitální technologie do školního vzdělávání a kteří mohou realizovat jejich potenciál přímo ve výuce. Jak ukazují šetření ČŠI i další výzkumy, velká část učitelů deklaruje zvládnutí základní či pokročilé úrovně dovedností z oblasti digitálních technologií. Zlepšují se rovněž schopnosti a dovednosti v práci s digitálními technologiemi u ředitelů škol. Učitelé zpracovali značné množství digitálních učebních materiálů a mnozí mají zkušenosti s využitím různých technologií i on-line služeb ve výuce. Data ukazují, že značná pozornost je věnována rozvoji dovedností v oblasti digitálních technologií i v dalším vzdělávání učitelů. Tato oblast patří v dalším vzdělávání učitelů k jedné z dominujících. Na druhé straně se tyto deklarované dobré znalosti učitelů příliš neprojevují v reálné výuce. Digitální technologie používají učitelé primárně k přípravě na vyučování, podstatně méně jsou využívány přímo ve vyučování.“ [1].

Na základě výše uvedených studií lze formulovat otázku, jaké jsou vlastně funkční ICT kompetence učitelů. Současně by bylo žádoucí ověřit, zda šetření založená na autoevaluačních

hodnocení učitelů neposkytují přinejmenším diskutabilní výsledky a zjistit, jak má být z hlediska funkčních ICT kompetencí učitelů interpretováno konstatování, že velká část učitelů deklaruje zvládnutí základní či pokročilé úrovně dovedností z oblasti digitálních technologií (viz citace ze Strategie digitálního vzdělávání do roku 2020 uvedená výše). Testování funkčních ICT kompetencí učitelů je obtížné stejně jako jakékoliv jiné odborné testování funkčních znalostí a dovedností u učitelů v praxi. Proto bylo provedeno šetření na vzorku studentů neinformatických oborů pedagogické fakulty, z jejichž funkčních ICT znalostí a dovedností lze s určitými omezeními extrapolovat také ICT znalosti a dovednosti učitelů.

## 2. e-SKILLS FOR JOBS 2014 & IT FITNESS TEST

V roce 2014 proběhla kampaň *e-Skills for Jobs 2014* iniciovaná Evropskou komisí, jmenovitě Generálním ředitelství pro podniky a průmysl. V České republice byla kampaň koordinována Domem zahraniční spolupráce. Hlavním cílem této iniciativy bylo pomoci motivovat mladé lidi, především žáky a studenty základních, středních a vysokých škol pro studium a profesní uplatnění v oblasti informačních a komunikačních technologií. Dalším cílem bylo podnítit zájem veřejnosti o zvyšování úrovně IT dovedností v každodenním i profesním životě (DZS 2015). Jednou z aktivit kampaně byl *IT Fitness Test 2014*, pomocí kterého si široká veřejnost mohla otestovat své znalosti a dovednosti týkající se informačních technologií. IT Fitness Test probíhal online v období 1. 5. – 30. 9. 2014 na stránkách [www.itfitness.cz](http://www.itfitness.cz).

IT Fitness Test byl primárně zaměřen na žáky a studenty, ale zúčastnili se ho i zájemci z vyšších věkových skupin, zejména učitelé. Celkový počet účastníků dosáhl 13 666, z nichž 12 822 se podrobilo testování, ostatní poskytli pouze základní informace o sobě a svém vztahu k digitální technice v úvodním dotazníku. Testování bylo řešeno interaktivním způsobem, prostřednictvím webové aplikace. Jednalo se vlastně o dva různé testy, první (dále označen jako ZŠK) byl určen pro základní školy, druhý (označen VVS) byl určen přednostně pro střední školy, případně i pro zájemce ve vyšších věkových skupinách. Mnozí účastníci zřejmě využili možnosti projít oběma testy, o čemž svědčí věková skladba jejich řešitelů. V souboru účastníků testu převažovali muži v počtu 7 932, žen bylo 5 734. Nejvíce účastníků patřilo svým věkem do desetiletí 11–20 let, modus byl 15 let. Další informace byly získány pro soubory řešitelů obou testů, tedy ZŠK pro základní školy, kterých bylo 7 546 a testu VVS pro vyšší věkové skupiny, primárně pro střední školy, kde řešitelů bylo 5 276. Před vlastním provedením testu vyplnili respondenti úvodní část dotazníku (pohlaví, věk atd.) a dále odpovídali na 15 otázek týkajících se jejich vztahu k digitálním technologiím, jejich možnostem tyto technologie využívat a realizovaným nebo preferovaným způsobům jejich využívání. Následovalo 24 otázek vědomostní a kompetenční části testu. Témata otázek a zjištěné úspěšnost respondentů jsou uvedeny v tabulce č. 1. Souhrnné výsledky testu jsou uvedeny ve zprávě IT Fitness Test 2014 – Výsledky [8].

**Tabulka 1. Témata otázek vědomostní a kompetenční části testu [8]**

IT Fitness Test 2014 – ZŠK	Úspěšnost [%]	IT Fitness Test 2014 – VVS	Úspěšnost [%]
datové formáty	68,60	přenos dat	24,13
bezpečné heslo	57,90	vstupní/výstupní zařízení	26,93
licence pro software	47,10	mobilitní zařízení	55,55
e-mailové adresy	78,00	komprimování souborů	34,74
výběr vhodného softwaru	62,80	přenos dat v síti	28,77
e-mail – skrytá kopie	31,10	základní informace o PC	36,88
e-mail – netiketa	76,00	e-mail	50,32
sociální sítě	73,30	hoaks	39,67
textový editor – pravopis	73,10	bezpečnost	67,63
textový editor – nástroje	26,90	open source software	45,34
práce s grafikou	36,40	licence – software	50,42
vzorce v tabulkovém procesoru	23,80	licence – práce s wikipedií	42,55
grafy v tabulkovém procesoru	62,20	práce s grafikou	21,99
soubory a složky	21,80	graf tabulky	22,63
adresářová struktura	51,10	jednoduché vzorce	50,44
vlastnosti složek	59,50	kopírování vzorců	18,54
vyhledávání – otevírací doba	66,40	formátování textů	45,13
vyhledávání – ceny	52,90	vyhledávání – internetový obchod	44,60
vyhledávání – mapa	49,70	vyhledávání – konference	29,89
vyhledávání – obrázky	18,80	vyhledávání – obrázky	33,30
vyhledávání – úřední hodiny	35,60	vyhledávání – texty	57,22
vyhledávání – jízdní řády	35,80	vyhledávání – Erasmus +	50,08
porozumění algoritmu	43,00	vyhledávání – kalkulačka kurzů	59,50
určení nesprávného algoritmu	32,10	vyhledávání – GPS a mapy	53,47

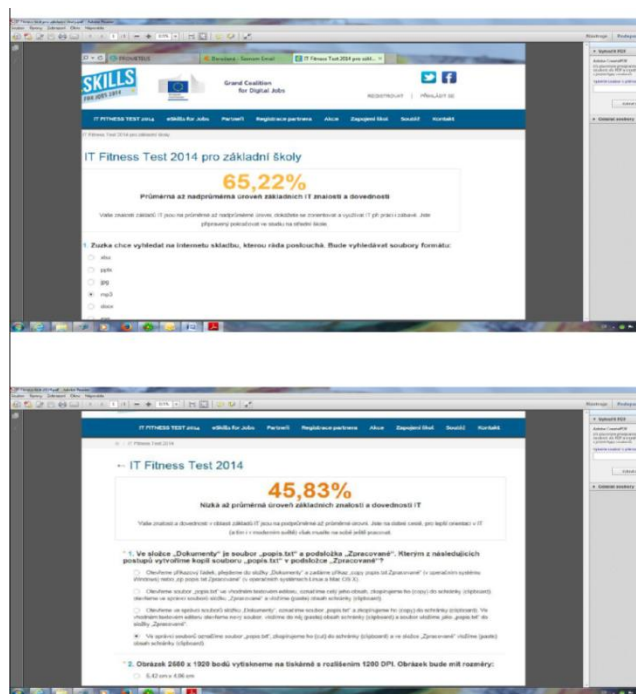
## 2.1 IT Fitness Test 2014 na Univerzitě

### J. E. Purkyně v Ústí nad Labem

IT Fitness test byl použit také ke zjištění ICT znalostí a dovedností posluchačů *Pedagogické fakulty Univerzity J. E. Purkyně v Ústí nad Labem* (dále jen UJEP). Po dohodě s *Domem zahraniční spolupráce* bylo období testování prodlouženo tak, aby přesáhlo do období začátku zimního semestru. Posluchači, kteří si v zimním semestru zapsali ICT předměty ve skupinách vyučovaných autorem tohoto šetření, dostali za domácí úkol vypracovat oba dva testy (ZŠK i VVS). Vzhledem k možnostem testovací on-line aplikace poskytované externím dodavatelem nebylo možno výsledky studentů UJEP přímo separovat. Proto studenti zkopirovali výsledky svého testu z displeje do obrázku a zaslali e-mailem vyučujícímu k dalšímu zpracování. Posluchači byli instruováni, aby test vypracovali samostatně s tím, že jedním z cílů ICT předmětu, který si zapsali, je opakování a upevňování digitálních znalostí a dovedností. Byli informováni, že výsledky testu mohou sami použít jako indikátor svých znalostí a dovedností a současně, že vyučující nepoužije jeho výsledky při hodnocení jejich studia jinak, než že pouze ověří, že mu byl soubor s výsledky byl doručen.

Příklad výsledků studentů je uveden na obrázku č. 1. Při zpracování výsledků bylo vizuálně kontrolováno, že v případě dosažení shodných výsledných numerických hodnot se v detailech liší snímek obrazovky, tj. že si studenti snímky navzájem nepřeposílali. Výsledky studentů byly rozříděny podle studijních skupin, tabelovány, statisticky zpracovány a porovnány s globálními souhrnnými výsledky. Z hlediska souhrnných výsledků uvedených ve zprávě *IT Fitness Test 2014 – Výsledky* jsou výsledky studentů UJEP součástí (podmnožinou) těchto souhrnných výsledků

Použitá metoda sběru dat neumožnila podrobnější analýzu výsledků v jednotlivých skupinách otázek, poskytla pouze podklady pro souhrnné hodnocení ICT znalostí a dovedností posluchačů. Testování se zúčastnili posluchači bakalářských studijních programů sociální pedagogika, sociálně pedagogická asistence a pedagogika volného času v denní i kombinované formě. Počty studentů jednotlivých studijních programů jsou



**Obrázek 1. Příklad snímku obrazovky s výsledky IT Fitness testu 2014, který studenti zasílali vyučujícímu k dalšímu zpracování**

uvedeny v tabulce č. 2, celkem 140 studentů odevzdalo výsledky testování v obou dvou testech (ZŠK i VVS). Vzhledem k tomu, že úkol byl zadán studentům hromadným e-mailem před zahájením semestru, je průměrná návratnost 86 % překvapivě vysoká. Přibližně 30 % studentů bylo upomínáno dva dny před vypršením lhůty, aby test vykonali a jeho výsledky zaslali. Do testování nebyli zařazeni ti studenti, kteří se ke studiu zapsali po termínu 1. 10. 2015 (byť ve lhůtě umožňující řádný zápis podle vnitřních předpisů UJEP).

**Tabulka 2. Rozdělení zkoumaného vzorku studentů do skupin podle studijního programu a návratnost testu**

Rozdělení studentů podle studijních programů						
Skupina = Studijní program – bakalářský	Studijní předmět	Ročník	Forma studia	Počet studentů, kteří zaslali výsledky [n]	Počet studentů zapsaných na předmět [n]	Návratnost [%]
A = Sociální pedagogika (SOCPED)	Základy výpočetní techniky	1	denní	29	37	78%
B = Sociálně pedagogická asistence (SPA)	Základy informatiky	1	denní	11	12	92%
C = Pedagogika volného času (PVČK)	Výpočetní technika	1	kombinovaná	27	32	84%
D = Sociálně pedagogická asistence (SPA)	Základy informatiky	1	kombinovaná	41	44	93%
E = Pedagogika volného času (PVČK)	Multimédia	3	kombinovaná	32	37	86%
<b>Celkem</b>				<b>140</b>	<b>162</b>	<b>86%</b>

## 2.2 IT Fitness Test 2014 pro základní školy

IT Fitness test určený žákům základní školy pokrývá široce témata v oblasti ICT. Vědomostní a kompetenční část, tedy vlastní test, určený žákům základních škol sestával z 24 otázek. Tvůrci testu odhadovali čas potřebný k jeho absolvování na 25–40 minut, skutečnost je blízká dolní hranici tohoto intervalu, modus činí 26 minut. Řešiteli byly předkládány otázky jedna po druhé, vždy s konečným počtem (nejméně čtyřmi) nabízených odpovědí,

z nichž právě jedna byla správná. Charakteristickým rysem použité metody testování bylo, že každá předložená otázka byla náhodně vygenerována ze čtyř připravených variant. Tím se dosáhlo toho, že variant testu bylo obrovské množství, takže výsledky mezi účastníky byly prakticky nepřenositelné. Přitom je ovšem důležité, že čtyři varianty otázek byly zhruba rovnocenné co do obtížnosti (viz níže) a že případy významně těžší varianty se vyskytly jen výjimečně.



Jednotlivá témata otázek jsou spolu s příslušnou úspěšností (přesněji procentní skóre, podíl správných odpovědí) uvedena v tabulce č. 2. Celková úspěšnost v testu se pohybovala od nuly (nejčastěji zřejmě u řešitelů, kteří se vzdali v rané fázi řešení) až do sta procent, dosažených jedním učitelem a jedním žákem 9. ročníku. Modus je 42 %, průměrná úspěšnost činí 49 %. Z kombinace s výsledky doprovodného dotazníku se potvrdily celkem očekávatelné rozdíly mezi skupinami charakterizovanými sociálním postavením nebo dosaženým vzděláním. Učitelé (pravděpodobně převážně učitelé IT) dosáhli v průměru 70% úspěšnost oproti 48% úspěšnosti žáků a studentů. U dospělých řešitelů byli nejúspěšnější absolventi gymnázia a absolventi vysokoškolského magisterského studia, shodně také se 70% úspěšností.

### 2.3 IT Fitness Test 2014 pro vyšší věkové skupiny

Test určený žákům střední školy (obecně vyšším věkovým skupinám), stejně jako paralelní test určený základním školám, pokrývá široce témata v oblasti informačních technologií. Vědomostní a kompetenční část, tedy vlastní test určený vyšším věkovým skupinám sestával z 24 otázek. Tvůrci testu odhadovali čas potřebný k jeho absolvování na 25–40 minut, skutečná doba spadá do tohoto intervalu, modus činí 30 minut. Podobně jako v případě testu ZŠK byly otázky náhodně generovány ze čtyř připravených variant.

Jednotlivá témata otázek jsou spolu s příslušnou úspěšností (přesněji procentní skóre, podíl správných odpovědí) uvedena v tabulce č. 2. Celková úspěšnost v testu se pohybovala od nuly (nejčastěji zřejmě u řešitelů, kteří se vzdali v rané fázi řešení) až

do 100 %, dosažených jedním učitelem VOŠ/SOŠ a jednou absolventkou bakalářského studia. Modus je 33 %, průměrná úspěšnost činí 42 %. Učitelé (pravděpodobně převážně učitelé IT) dosáhli v průměru 63% úspěšnost oproti 40% úspěšnosti žáků a studentů. U dospělých řešitelů byli nejúspěšnější absolventi vysokoškolského magisterského studia, zdanlivě paradoxně bakaláři s 68 % a magistři s 62 % úspěšnosti.

### 2.4 Výsledky IT Fitness Test 2014 na UJEP

Oba testy byly zadány posluchačům způsobem popsaným výše v kapitole 1.1. Výsledky jednotlivých respondentů byly tabelovány a byly pro ně vypočteny základní veličiny popisné statistiky. Průměrná úspěšnost a modus úspěšnosti ve vědomostní a kompetenční části testu jsou pro jednotlivé studijní programy uvedeny v tabulce č. 3. Nejhorší výsledky byly zjištěny u skupiny B = sociálně pedagogická asistence v prvním ročníku denního studia, nejlepší výsledky měli studenti skupiny E = pedagogika volného času ve třetím ročníku kombinovaného studia, u kterých se pravděpodobně projeví větší teoretické znalosti a praktické dovednosti získané v průběhu dvou ročníků studia. Jako relativně nejlepší lze hodnotit respondenty skupiny D = sociálně pedagogická intervence v prvním ročníku kombinovaného studia, kteří byli druzí nejlepší v absolutním pořadí se ztrátou 2 % v testu ZŠK, resp. 4 % v testu VVS na nejlepší skupinu E. Pokud bychom však uvedený rozdíl přičetli na vrub pouze výsledkům vzdělávání v ICT předmětech za dva roky studia, o které se tyto skupiny liší, jeví se by se přidaná hodnota ICT předmětů ke znalostem a dovednostem studentů jako relativně malá. Dále je třeba konstatovat, že tento rozdíl je srovnatelný s rozdíly mezi výsledky respondentů různých studijních programů v prvním ročníku (vzájemné rozdíly mezi skupinami A, B, C, D).

**Tabulka 3. Výsledky IT Fitness Testu 2014 – UJEP**

	IT Fitness Test 2014	ZŠK			VVS		
	úroveň vzdělání	počet řešitelů [n]	průměrná úspěšnost [%]	modus [%]	počet řešitelů [n]	průměrná úspěšnost [%]	modus [%]
UJEP	A = Sociální pedagogika (SOPED)	29	64%	42%	29	46%	38%
	B = Sociálně pedagogická asistence (SPA)	11	57%	58%	11	44%	42%
	C = Pedagogika volného času (PVČK)	27	67%	63%	27	48%	38%
	D = Sociálně pedagogická asistence (SPA)	41	69%	63%	41	48%	46%
	E = Pedagogika volného času (PVČK)	32	71%	75%	32	52%	46%
	celkem	140	67%	75%	140	48%	38%

Všichni respondenti zodpověděli úspěšně alespoň jednu otázku, minimální úspěšnost v celém souboru v testu ZŠK činí 35 %, v testu VVS pak jen 4 %. Pokud by tyto minimální výsledky skutečně odpovídaly ICT znalostem a dovednostem studentů vysoké školy zjištěným jinými metodami, mohli by tito studenti být označeni za osoby bez *funkční ICT gramotnosti*. Maximální úspěšnost v celém souboru pak činila 96 % v testu ZŠK, resp. 92 % v testu VVS, přičemž v každé skupině se vyskytovali studenti s úspěšností lepší než 80 %. Tito studenti naopak prokázali nadprůměrné ICT znalosti a dovednosti a lze je označit za osoby *funkčně ICT gramotné*. Dále bylo zjišťováno, zda existují významné rozdíly u jednotlivých respondentů mezi jejich výsledky v testu ZŠK a VVS. Střední hodnota rozdílů činí 19 %, což v souladu s metodikou výpočtu odpovídá rozdílu mezi celkovými průměrnými úspěšnostmi. Modus rozdílů pak činil 12 %, modus absolutních hodnot rozdílů jen 8 %. Maximum rozdílů činí 67 % (respondent pravděpodobně vyplnil zodpovědně jen jeden test, druhý jen náhodně odklikal), v každé skupině byl

zjištěn alespoň jeden rozdíl 50 % a větší, tzn. existují studenti, kteří vykazují výrazné rozdíly ve výsledcích testů ZŠK a VVS. Minimum rozdílů činí –14 % (výsledky v těžším testu VVS byly u daného respondenta lepší než výsledky v lehčím testu ZŠK).

Aby bylo možno porovnat výsledky studentů UJEP s výsledky celého souboru respondentů, byly obě sady výsledků uvedeny společně do tabulky č. 4. Zde se ukázalo, že výsledky respondentů UJEP v testu ZŠK jsou v průměru srovnatelné s výsledky respondentů aktuálně studujících vysokou školu, jsou však výrazně horší v testu VVS. Porovnání výsledků ve skupinách podle profesního zařazení vede ke značnému rozdílu, učitelé v průměru dosáhli lepších výsledků v testu ZŠK a výrazně lepších výsledků v testu VVS než studenti neinformatických oborů pedagogické fakulty UJEP. Pravděpodobnou příčinou je důvod uvedený v souhrnné zprávě *IT Fitness Test 2014 – Výsledky* [8] – učitelé, kteří se IT Fitness testu zúčastnili, byli převážně učitelé informatických předmětů, u kterých lze předpokládat vyšší úroveň ICT kompetencí.

**Tabulka 4 Výsledky IT Fitness Testu 2014 – souhrnné výsledky [8] a vlastní výsledky (UJEP)**

	IT Fitness Test 2014	ZŠK		VVS	
	úroveň vzdělání	počet řešitelů [n]	průměrná úspěšnost [%]	počet řešitelů [n]	průměrná úspěšnost [%]
Souhrnné výsledky podle dosaženého vzdělání	aktuálně žák základní školy	4795	45%	168	36%
	aktuálně žák střední školy	2341	57%	4712	41%
	aktuálně student vysoké školy	154	64%	167	53%
	absolvent základní školy	96	44%	21	36%
	absolvent VŠ – magisterské studium	63	70%	65	62%
	absolvent střední odborné školy	36	65%	57	52%
	absolvent středního odborného učiliště	15	56%	9	47%
	absolvent VŠ – doktorské studium	13	56%	16	56%
	absolvent 4letého gymnázia	12	70%	27	49%
	absolvent VŠ – bakalářské studium	9	63%	18	68%
	absolvent 8letého gymnázia	6	52%	7	47%
	absolvent střední zdravotní školy	6	64%	9	37%
	<b>celkem</b>	<b>7546</b>	<b>49%</b>	<b>5276</b>	<b>42%</b>
Souhrnné výsledky podle zařazení	student / žák	7021	48%	4872	40%
	učitel	269	70%	175	63%
	jiné	256	57%	229	54%
	<b>celkem</b>	<b>7546</b>	<b>49%</b>	<b>5276</b>	<b>42%</b>
UJEP	A = Sociální pedagogika (SOCPED)	29	64%	29	46%
	B = Sociálně pedagogická asistence (SPA)	11	57%	11	44%
	C = Pedagogika volného času (PVČK)	27	67%	27	48%
	D = Sociálně pedagogická asistence (SPA)	41	69%	41	48%
	E = Pedagogika volného času (PVČK)	32	71%	32	52%
	<b>celkem</b>	<b>140</b>	<b>67%</b>	<b>140</b>	<b>48%</b>

### 3. ZÁVĚR

Informační a komunikační technologie se v českých školách všeobecně nevyužívají na úrovni srovnatelné s moderní informační společností, v souladu s aktuálními výsledky světového pedagogického výzkumu a celosvětovými trendy. Existují výjimky v pozitivním, ale i negativním smyslu. Mezi respondenty on-line testů ICT dovedností, mezi které lze IT Fitness Test 2014 zařadit, lze identifikovat respondenty na celé škále ICT znalostí a dovedností, od osob bez funkční ICT gramotnosti až po osoby s výbornými ICT kompetencemi. Ukazuje se, že absolvování vysokoškolského studia sice obecně vede ke zlepšení úspěšnosti v IT Fitness Testu, ovšem toto zlepšení není dostatečně výrazné. To může být způsobeno relativně rychlým zastaráváním uživatelských ICT znalostí a dovedností, resp. „zakonzervováním“ ICT znalostí a dovedností získaných na střední, případně vysoké škole. Příkladem může být paradoxně slabá úspěšnost v IT Fitness Testu u absolventů doktorského studia.

Nelze než konstatovat, že podceňování systematického a komplexního vzdělávání v oboru ICT vede k horším znalostem a dovednostem uživatelů ICT a negativně se projevuje na jejich ICT kompetencích. Z tohoto hlediska je nutno podpořit aktivity vedoucí ke zkvalitnění ICT přípravy budoucích učitelů jak ve smyslu obecných ICT znalostí a dovedností, tak ve smyslu lepšího využívání ICT v oborových didaktikách. Výsledky IT Fitness testu jsou jen dalším argumentem pro plnou realizaci Strategie digitálního vzdělávání do roku 2020, ve které se konstatuje, cit: „Zdá se tedy, že se poměrně velké investice do vzdělávání učitelů v oblasti využití technologií částečně mjejí účinkem, protože nově získané dovednosti v oblasti využívání digitálních technologií nejsou aplikovány v praxi. To je zřejmě dáno i tím, že účelné využití digitálních technologií naše kurikulární dokumenty

i kariérní systém prakticky nepodporují. Ve výuce se doposud nevyužívá propojení různých učebních prostředí, kterými může být nejenom škola či třída, ale také domov, virtuální on-line prostředí, různé podoby informálního učení (ve vrstevnických skupinách, v oblasti trávení volného času apod.). Právě v mimoškolním prostředí jsou digitální technologie používány běžně a mnohdy jinými a současně inspirativními způsoby, z nichž by mohlo těžit i školní vzdělávání. Nad rámec výše uvedeného lze v souladu se výsledky IT Fitness testu doplnit další pravděpodobnou příčinu málo účinného aplikování technologií v pedagogické praxi, a to je formalismus ve vzdělávání učitelů v oblasti ICT, který vede ke zdánlivému zlepšování ICT znalostí a dovedností učitelů bez relevantního dopadu na jejich funkční ICT gramotnost.

### 4. PODĚKOVÁNÍ

Autor děkuje pracovníkům Domu zahraniční spolupráce za laskavé prodloužení *IT Fitness Testu 2014* pro veřejnost do konce září 2014, bez kterého by nebylo možno testování u studentů UJEP realizovat.

### 5. BIBLIOGRAFICKÉ ODKAZY

- [1] MŠMT ČR, 2014. *Strategie digitálního vzdělávání do roku 2020*. [online]. c2014-10-31. [2015-02-28]. <[http://www.msmt.cz/file/34429\\_1\\_1/](http://www.msmt.cz/file/34429_1_1/)>.
- [2] PEŠAT, P. SEIFERT, M., 2011. ICT dovednosti u studentů učitelství (TIGR pro budoucí učitele). In *Evropské pedagogické fórum 2011. Současná společnost a profese učitele. Sborník konference 5.–9. prosince 2011. Hradec Králové*. Magnanimitas: Hradec Králové, 2011. s. 324–331. ISBN 978-80-904877-6-5.

- [3] RAMBOUSEK, V., a kol., 2013. *Rozvoj informačně technologických kompetencí na základních školách*. Vyd. 1. Praha: České vysoké učení technické v Praze. 351 s. ISBN 978-80-01-05407-9.
- [4] BASL, J., BOUDOVÁ, S., ŘEZÁČOVÁ, L., 2014. *Národní zpráva šetření ICILS 2013: počítačová a informační gramotnost českých žáků*. Praha: Česká školní inspekce, 2014. 57 s. ISBN 978-80-905632-6-1. [online]. c2014. [2015-02-28]. <[http://www.icils.cz/articles/files/ICILS\\_2013\\_Narodni\\_zprava\\_CZE.pdf](http://www.icils.cz/articles/files/ICILS_2013_Narodni_zprava_CZE.pdf)>.
- [5] K PEŠAT, P., MEIER, M., 2014. Prostředky informačních a komunikačních technologií ve vzdělávání žáků se speciálními vzdělávacími potřebami. In: LOVÁSZOVÁ, G., ed. *DIDINFO 2014, Sborník 20. ročníka národnej konferencie s medzinárodnou účasťou. 2.–4. apríl 2014, Banská Bystrica*. Univerzita Mateja Bela: Banská Bystrica, 2014. s. 118–125. ISBN 978-80-557-0698-6.
- [6] MEIER, M., 2014. Informační a komunikační technologie a edukace žáků se speciálními vzdělávacími potřebami. In: CHRÁSKA, M., KLEMENT, M., SERAFÍN, Č., HAVELKA, M., eds. *Trendy ve vzdělávání 2014 [CD]*. 1. vyd. Olomouc: Gevak, s. 230–233. ISBN 978-80-86768-89-2.
- [7] LOPÚCHOVÁ, J., KRAJČÍ, P., 2014. *Teória a výskum digitálnych a asistenčných technológií v inkluzívnej špeciálnej pedagogike*. 1. vyd. Ostrava: Ostravská univerzita 2014. 200 s. ISBN 978-80-7464-675-1.
- [8] DŮM ZAHRANIČNÍ SPOLUPRÁCE (DZS), 2014. *IT Fitness test 2014 – Výsledky*. c2014. [cit. 2015-02-25]. <<http://www.dzs.cz/admin/file/2485/IT%20Fitness%20test%20V%C3%BDsledky.pdf>>.
- [9] AUTORSKÝ KOL. VÚP, 2013. *Rámcový vzdělávací program pro základní vzdělávání (verze platná od 1. 9. 2013) – RVP-ZV 2013*. c2013-05-28. [cit. 2014-01-15]. <<http://www.msmt.cz/vzdelavani/zakladni-vzdelavani/upraveny-ramcovy-vzdelavaci-program-pro-zakladni-vzdelavani>>.
- [10] DŮM ZAHRANIČNÍ SPOLUPRÁCE (DZS), 2015. *Webová stránka eSKILLS FOR JOBS 2014*. c2015. [cit. 2015-02-25]. <<http://www.itfitness.cz/cs/eskills-for-jobs-campaign/>>.
- [11] EVROPSKÝ PARLAMENT, 2006. *Doporučení Evropského parlamentu a Rady ze dne 18. prosince 2006 o klíčových schopnostech pro celoživotní učení*. c2006-12-29. [cit. 2011-14-10]. <<http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2006:394:0010:0018:CS:PDF>>.
- [12] EVROPSKÝ PARLAMENT, 2010. *Usnesení Evropského parlamentu ze dne 5. května 2010 o nové digitální agendě pro Evropu: 2015.eu, 2010*. c2010-05-05. [cit. 2011-11-15]. <<http://www.europarl.europa.eu/sides/getDoc.do?pubRef=-//EP//TEXT+TA+P7-TA-2010-0133+0+DOC+XML+V0//CS>>.
- [13] SBÍRKA ZÁKONŮ ČR, 2014. *Zákon č. 561/2004 Sb., o předškolním, základním, středním, vyšším odborném a jiném vzdělávání (školský zákon)*. [online]. c2008-08-280. [2014-01-10]. <<http://www.msmt.cz/dokumenty/uplne-zneni-zakona-c-561-2004-sb>>.

# Využívanie disponibilných hodín k aktualizovaniu školského vzdelávacieho programu

Peter Potocký

Stredná odborná škola informačných technológií

Hliníčka 1

831 52 Bratislava

Slovensko

peter1potocky@gmail.com

## ABSTRAKT

Aktuálny prínos pre didaktiku informatiky vychádza zo sledovania rozmachu vo využívaní počítačovej techniky prostredníctvom informačných technológií, ktorý zaznamenávame v súčasnosti. Daná skutočnosť kladie na učiteľa informatiky zvýšené nároky na jeho pedagogickú prácu. Na jednej strane musí sledovať daný rozmach svojho odborného predmetu a na druhej strane ho musí, v duchu toho rozmachu, aj aktualizovať v školskom vzdelávacom programe. Vyžaduje si to didaktickú zásadu spojenia teórie s praxou a školy so životom. Informujeme:

- Ako na súčasný rozmach využívania počítačovej techniky informačnými technológiami zareagoval zriaďovateľ našej školy svojimi požiadavkami.
- Ako sme pristúpili k riešeniu týchto jeho požiadaviek.
- A aké pedagogické výsledky sme pritom dosiahli.

## Kľúčové slová

robotika, humanoidný robot, disponibilné hodiny, štátny vzdelávací program, školský vzdelávací program, revidovaná Bloomová taxonómia vzdelávacích cieľov, programy Monitor 2.1.2 a Choregraphe 2.1.2.

## ÚVOD

Na začiatku 21. storočia sme svedkami výrazného rozmachu vo využívaní počítačovej techniky prostredníctvom informačných technológií v oblasti robotiky, ktorý v roku 2006 vyústil do narodenia roztomilého humanoidného robota. Do dnešného dňa „*bol vo viac ako 70 krajinách sveta použitý v počítačových a prírodovedných triedach, od základnej školy až po univerzitu*“ [1]. Humanoidný robot pomenovali NAO. Jedného dňa sa určite stane bežným, každodenným spoločníkom človeka. Treba však na ňom hodne popracovať, kým sa z tohto umelého 9 ročného chlapca stane dospelý jediniec s uvedenými vlastnosťami spoľahlivosti. Na Strednej odbornej škole (ďalej iba SOŠ) informačných technológií v Bratislave - centrum odborného vzdelávania a prípravy pre elektrotechniku a informačné technológie, sa v školskom roku 2015/2016 začne s výučbou NAO robota. Je to darček, ktorý sme na SOŠ dostali začiatkom decembra 2014 od Bratislavského samosprávneho kraja a Západoslovenskej energetiky Bratislava.

NAO robota som ukázal žiakom. Spočiatku sa o neho vážne začalo zaujímať 6 žiakov. V januári 2015 som z nich vytvoril **Klub priateľov humanoidného robota**. Začali sme sa stretávať

po vyučovacích hodinách. Najskôr sme NAO robota spoznávali formou hrania sa s ním. Po úvodných disponibilných vyučovacích hodinách [2] nasledovali náročnejšie úlohy, v ktorých už mali žiaci, okrem poznávania činností NAO robota, riešiť konkrétne príklady na jeho ovládanie. A výsledok? Zostala mi iba polovica z týchto žiakov. U nich rozvíjam programátorské znalosti i zručnosti s prácou NAO robota. Výučba zodpovedá záverom posledného prehodnotenia obsahovej reformy školstva v novom štátnom vzdelávacom programe [3]. Využívam ju na doplnenie školského vzdelávacieho programu o odborný predmet **Robotika**. V duchu uvedeného prehodnotenia obsahovej reformy školstva v každej téme vytvárame aktuálne učivo, ktoré nadväzuje na predchádzajúce. Učivo je zbavené prebytočných faktov. Takto na SOŠ začíname prípravu žiakov pre trh práce. Na tejto SOŠ je od minulého školského roku 2013/2014 zavedené, že žiaci s dobrým prospechom i dobrými výsledkami na odbornom výcviku sú priebežne začleňovaní do tímu pracovníkov aj v takých profesiách, v ktorých sa využíva práca s robotmi.

## 1. PREDSTAVENIE HUMANOIDNÉHO ROBOTA NAO

Predstavujeme vám humanoidného robota, ktorý sa volá NAO. Profesor Ki-sung Suh (Seo) o ňom napísal, že „*priťahuje omnoho väčšiu pozornosť ako ostatné roboty, pretože:*

- výzorom je podobný človeku, pôsobí priateľský a vie nám byť nápomocný pri plnení zadaných úloh;*
- rovnako ako ľudia, chodí po dvoch nohách a pri práci používa obe ruky*“ [4].

Humanoidný NAO robot pochádza z Aldebaran Robotics. V akademickom svete sa stal štandardom pre výskum a vzdelávanie.

Komunikácia medzi počítačom a NAO robotom je zabezpečená sieťovým káblom (drôtové pripojenie), no dá sa využiť i bezdrôtové pripojenie (WiFi). Funkčnosť hardvérového spojenia NAO robota s počítačom zabezpečuje softvér. Jedná sa o programy Monitor, NAOqi a Choregraphe, s ktorými vás oboznámime v nasledujúcej časti.

Program **Monitor** umožňuje zobrazit' videnie NAO robota cez dve kamery. Obe sú zabudované v jeho hlave (jedna v čele a druhá v ústach). Program **Choregraphe** obsahuje schránky, ktoré je možné využiť na riadenie pohybov i reči NAO robota. Schránky sú uložené v knižnici tohto grafického programu.

## 1.1 Užitočnosť pravidiel pre prácu s NAO robotom

Spočiatku som dúfal, že sa Klub priateľov humanoidného robota začne rozrastať. Väčšinu jeho prvých členov tvorili žiaci, ktorí sa v čase voľna intenzívne venujú bojovým hrám na počítači. Týchto ani moc nezaujala skutočnosť, že robotika je smer, ktorým sa bude uberať budúcnosť. Najviac ich oslovil fakt, že mnohým americkým pyrotechnikom v Afganistane zachránili roboty ich život. Keď však zistili, že práca s robotom si vyžaduje aj istý rozsah teoretických vedomostí, tak táto skupina žiakov sa svorne prestala zaujímať o prácu vo vzniknutom Klube priateľov humanoidného robota. Prestali ho navštevovať.

Ovládanie humanoidného NAO robota si vyžaduje určité pravidlá, ktoré je nevyhnutné si osvojiť. Okrem fyzického manipulovania je tu ešte manipulácia softvérovým produktom Choregraphe. Treba dohliadnuť, aby žiaci používali iba tie schránky, ktoré poznajú a vedia s nimi pracovať. Napríklad, pokiaľ nevedia ako sa v Choregraphe používa programovací modul (schéma) či programovanie a nastavenie pohybov NAO robota v časovej osi, tak im so zvládnutím tohto učiva pomôžeme. Pri nedodržaní daných pravidiel hrozí, že pri práci s NAO robotom môže dôjsť k jeho poškodeniu.

## 2. AKTUALIZOVANIE ŠKVP PODĽA REVIDOVANEJ BLOOMOVEJ TAXONÓMIE

Zostávajúca skupina žiakov, ktorí mali vážny záujem o robotiku sa naďalej dobrovoľne zúčastňuje voliteľných hodín, ktoré prebiehajú mimo času povinného vyučovania. Okrem odbornej stránky si na týchto hodinách vo výraznej miere upevňujú také charakterové vlastnosti, ako sú spoľahlivosť, prirodzený záujem o uvedenú oblasť informačných technológií a čo je najdôležitejšie, hravou formou si rozvíjajú vlastný intelekt a pestuje sa v nich zmysel pre porozumenie druhých. Vlastne som rád, že sa mi príchodom NAO robota konečne vytvoril priestor pre prácu so žiakmi podľa mojich vlastných predstáv.

Využívame uvedené disponibilné hodiny pre aktualizáciu školského vzdelávacieho programu, a to o odborný predmet robotika. Na týchto hodinách žiaci nadobúdajú zručnosti v ovládaní robota. Cielene vediem s nimi rozhovor na danú tému, počas ktorého zisťujem, do akej hĺbky si osvojili danú problematiku. V priebehu tohto rozhovoru sa objavujú pre nich ďalšie neznáme poznatky, ktoré im následne vysvetlím. Tým, že sa vytvára vzájomne pozitívna komunikácia medzi mnou a žiakmi si zabezpečím, aby dobre porozumeli tomu, čo budú robiť s NAO robotom.

Uvedeným spôsobom výučby sa žiaci, namiesto bifikovania a mechanického zvládnutia naučených postupov, dopracúvajú k logickému osvojovaniu si tejto časti učiva a zároveň nadobúdajú i praktické zručnosti v ovládaní NAO robota.

### 2.1 Logické osvojovanie učiva žiakmi počas disponibilných hodín

Z priestorového hľadiska tu nie je možné venovať sa úlohám, ktoré žiaci riešia pri jednotlivých tematických celkoch a ani to, ako ich žiaci zvládnu riešiť, respektíve s čím majú problémy. Preto sa obmedzíme iba na jednu takúto úlohu.

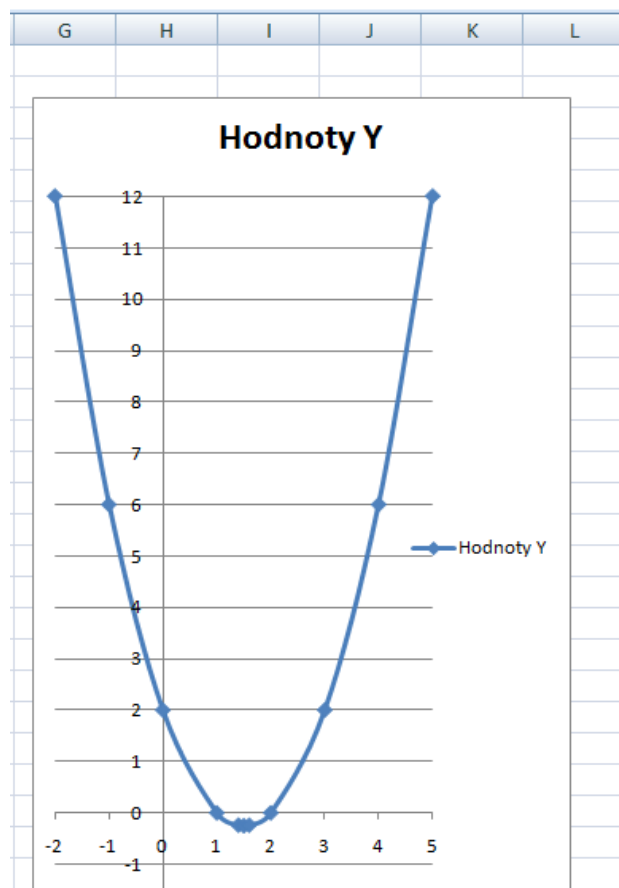
Pred vytváraním jednotlivých tém učiva o NAO robote som od žiakov vyžadoval, aby porozumeli týmto poznatkom o ňom:

- Hardvérová platforma (kľúčové komponenty).
- Pohyb (chôdza, pohyb celého tela a riadenie pádu).
- Videnie.
- Audio (lokalizácia zvukového zdroja a popis zodpovedajúcich programových aplikácií).
- Hmatové senzory a diaľkomer (horizontálny sonar).
- Pripojenie (drôtové a bezdrôtové).
- Otvorený zdroj (Open source) [1].

Pri pohybe celého tela NAO robota žiaci neporozumeli významu, že „symetria jeho pohybu je zabezpečená na princípe kvadratického programovania“ [1].

Určite by nebolo správne vysvetľovať žiakom princíp kvadratického programovania, ktoré sa vyučuje na vysokej škole až po lineárnom programovaní. Preto som k tomu využil názornosť s jednoduchou predstavivosťou.

Najskôr si žiaci v matematickom procesore MS Excel 2007 graficky vyriešili kvadratickú funkciu. Zadal som ju rovnicou  $y = x^2 - 3x + 2$ , pre  $x = (-2, -1, 0, 1, 1.4, 1.5, 1.6, 2, 3, 4 \text{ a } 5)$ . Graf tejto funkcie vidíme na obrázku 1:



Obrázok 1: Graf kvadratickej funkcie  $y = x^2 - 3x + 2$

Parabola je zobrazená v dvojrozmernom priestore  $E_2$ . Ak za stred paraboly budeme považovať strednú os NAO robota, tak pohyby jeho rúk budú v zrkadlovom zobrazení určitého výseku kužeľa, čo je trojrozmerný priestor  $E_3$ . Symetrické zabezpečenie pohybov oboch rúk sa rieši kvadratickým programovaním.

Po tomto vysvetlení si zvyšní traja žiaci uvedomili dôležitosť matematiky v práci profesionálneho programátora. Prejavili záujem o hlbšie štúdium tohto predmetu.

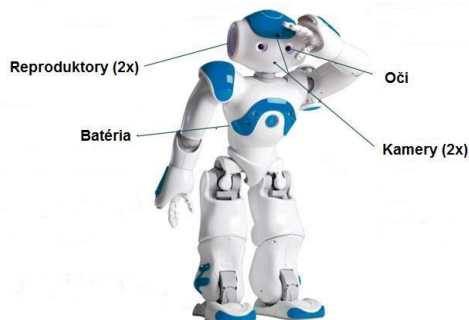


## 2.2 Koncentrické usporiadanie tematických celkov podľa revidovanej Bloomovej taxonómie vzdelávacích cieľov

Koncentrické usporiadanie tém o robotike podľa revidovanej Bloomovej taxonómie vzdelávacích cieľov [5] pozostáva z nasledovných štyroch tematických celkov:

Prvý tematický celok: Obsahuje zložky hardvéru NAO robota, ktoré sa budú využívať pri práci s programami Monitor a Choregraphe. Ostatné zložky som do učiva nezakomponoval. Takto som do potrebnej miery zredukoval obsah učiva o prebytočné poznatky o NAO robote, čo plne zodpovedá požiadavkám momentálne aktuálneho štátneho vzdelávacieho programu. Daný celok obsahuje jedno praktické cvičenie, v ktorom sa, okrem zapojenia NAO robota do zdroja, realizuje aj jeho fyzické pripojenie k počítaču (priama názornosť).

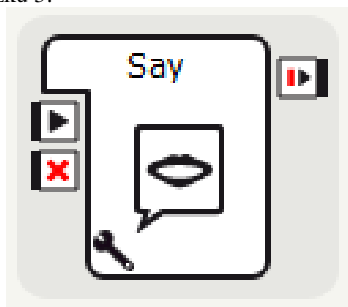
Na tejto prezentácii nechám NAO robota vykonať snímanie hornou a dolnou kamerou v programe Monitor. V programe Choregraphe mu prikážem hovoriť napísané slová. Pre túto prezentáciu vystačím s popisom hardvéru NAO robota, ktorý znázorňuje obrázok 2. Presne takýmto spôsobom vytváram prvý tematický celok tohto odborného predmetu, ktorý obsahuje iba tú časť jeho hardvéru, ktorá bude prakticky využívaná.



Obrázok 2: Popis využívaných častí hardvéru NAO robota, pri jeho prezentácii programami Monitor a Choregraphe

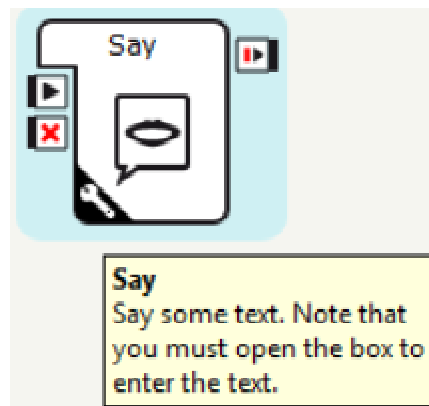
Druhý tematický celok: Zahŕňa poznatky o snímaní oboch kamier NAO robota a o vykonávaní záznamu z tohto snímania programom **Monitor**. Taktiež sa tu počíta s praktickými cvičeniami.

Tretí tematický celok: Je zameraný na prácu s programom **Choregraphe**, v ktorom aktivujeme nahrané aplikácie NAO robota cez program **NAOqi**. Táto práca si už vyžaduje väčší počet praktických cvičení. Na tejto prezentácii bude NAO robot hovoriť vety v slovenskom jazyku, ktoré mu napíšeme. K tomu využijeme schránku s názvom Say (Hovor) programu Choregraphe, ktorú je vidieť na obrázku 3:



Obrázok 3: Tvar schránky Hovor programu Choregraphe

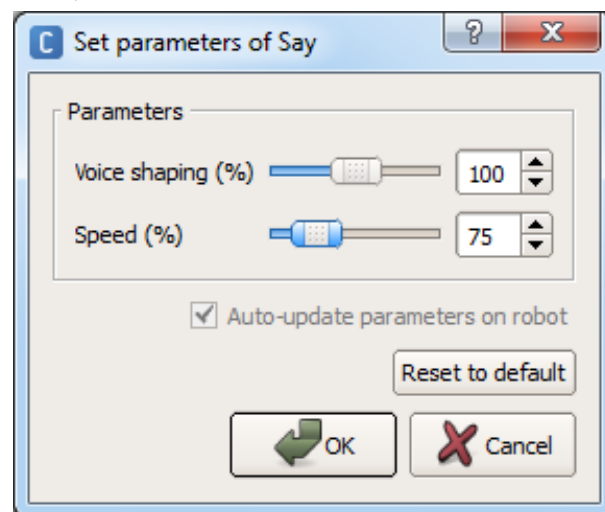
Ukážeme si aj zabezpečenie zrozumiteľnej výslovnosti NAO robota spomalením vyslovovania napísaných viet. Zabezpečíme to ťuknutím myšou na obrázok kľúča, ktorý je na obrázku 4 zobrazený bielou farbou v ľavom dolnom rohu schránky Hovor (Say).



Obrázok 4: Zobrazený kľúč bielou farbou v schránke Hovor

Všimneme si, že pri umiestnení ukazovateľa myši na kľúč sa najskôr zobrazí príkaz pre NAO robota: „*Hovor nejaký text.*“. Až potom nasleduje informácia pre programátora: „*Pamätajte si, že je potrebné otvoriť okno pre zadanie textu,*“ ktorý bude hovoriť NAO robot.

Uvedenému oznamu nevenujeme pozornosť, nakoľko má táto schránka štandardne zadaný text programom Choregraphe. Vykonáme iba uvedenú činnosť myšou. Na obrázku 5 vidíme zobrazené dialógové okno na úpravu hlasitosti a rýchlosti hovorených viet NAO robotom:



Obrázok 5: Dialógové okno na formovanie hlasu a rýchlosti hovorených viet NAO robotom v schránke Hovor (Say).

Štvrtý (posledný) tematický celok: Žiak si z ponúkaných nových schránok vytiahne jednu aj s obsahom jej činnosti. Daný obsah si preštuduje a podľa neho určí, ku ktorej časti hardvéru NAO robota ho treba priradiť. Pre vyriešenie tejto úlohy mu bude poskytnutý aj rozšírenejší popis hardvéru NAO robota. Novú schránku využije v programe **Choregraphe**. Danou úlohou skontrolujeme zvládnutie tohto odborného predmetu na úrovni 6. stupňa poznávacieho procesu revidovanej Bloomovej taxonómie vzdelávacích cieľov.

## ZÁVER

Týmto príspevkom demonštrujeme, ako sa dá skontrolovať koncentrické usporiadanie učiva revidovanou Bloomovou taxonómiou vzdelávacích cieľov, od najnižšieho stupňa poznávacieho procesu po najvyšší. Vďaka prebiehajúcim disponibilným hodinám so žiakmi môžeme už teraz potvrdiť, že jednohodinová týždenná dotácia vyučovacej hodiny z odborného predmetu robotika, na ktorej sa žiaci hravou formou naučia ovládať humanoidného NAO robota, bude rozhodne pozitívnym prínosom do výchovno-vzdelávacieho procesu na našej SOŠ.

Očakávame, že didaktické i odborné poznatky, ktoré som predložil na tejto konferencii, nezapadnú prachom, ale budú celoplošne zúžitkované v príprave žiakov na stredných odborných školách so zameraním na informačné technológie pre trh práce. Rovnako očakávame, že sa podstatné myšlienky tohto príspevku pozitívne využijú aj v príprave budúcich učiteľov robotiky na základných a stredných školách so zameraním na informatiku a informačné technológie.

Žiakov k výučbe humanoidného NAO robota motivuje najmä tá skutočnosť, že v zamestnaní ide o vysokokvalifikovanú prácu, ktorá je po finančnej stránke nadpriemerne hodnotená.

## POĎAKOVANIE

Touto cestou ďakujeme Bratislavskému samosprávnemu kraju a Západoslovenskej energetike, a. s. Bratislava za zakúpenie NAO robota SOŠ informačných technológií v Bratislave na Hlinickej ulici 1. Ďalej školiťovi zo Štátneho inštitútu odborného

vzdelávania za poskytnuté vedomosti k prezentovanej aktualizácii školského vzdelávacieho programu disponibilnými hodinami. A napokon vedeniu SOŠ a predsedovi predmetovej komisie za prejavenu dôveru a vykonané aktivity, ktoré viedli k výučbe odborného predmetu robotika na tejto SOŠ.

## BIBLIOGRAFICKÉ ODKAZY

- 1) Who is NAO? Dostupné na: <https://www.aldebaran.com/en/humanoid-robot/nao-robot>, prezerané dňa 25. januára 2015
- 2) Spracované podľa: JAKUBÍKOVÁ GABRIELA a JURKOVICHOVÁ ANNA: **Metodika tvorby školských vzdelávacích programov pre stredné odborné školy**. Vydal Štátny inštitút odborného vzdelávania. Bratislava 2011, strana 87 - 89. Bez ISBN
- 3) Spracované podľa: SALANCI ĽUBOMÍR: **Informatika v novom štátnom vzdelávacom programe**. In: **DidInfo 2014**. Vydala Univerzita Mateja Bela, Fakulta prírodných vied v Banskej Bystrici 2014, strana 143. ISBN: 978-80-557-0698-6
- 4) KISUNG SEO: **Using NAO: Introduction to interactive humanoid robots**. Vydal ALDEBARAN Robotics 2013. Vyšlo vo Francúzsku, strana 3. Vydateľský kód: ALDM400029\_P A00
- 5) Spracované podľa: TUREK IVAN: **Didaktika**. Vydala Iura Edition. Prvé vydanie. Bratislava 2008, strana 52. ISBN: 978-80-8078-198-9

# ANALÝZA INFORMATIKY V NOVOM ŠTÁTNO M VZDELÁVACOM PROGRAME

Ľubomír Salanci

Katedra základov a vyučovania informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského  
Mlynská dolina  
842 48 Bratislava  
Slovenská republika  
salanci@fmph.unbia.sk

## ABSTRAKT

Po inovácii štátneho vzdelávacieho programu pre prvý a druhý stupeň základných škôl vznikla nová verzia aj pre gymnáziá. V našom príspevku sa zameriavame na gymnáziá a maturitnú úroveň. Analyzujeme a porovnávame výkony a obsahy školskej informatiky naprieč rôznymi stupňami vzdelávania. Nadväzujeme tak na predchádzajúci príspevok, v ktorom sme sa venovali predovšetkým informatike na základnej škole.

## ABSTRACT

After upgrading of the national educational program for primary and lower secondary schools, it was created also a new version for grammar schools. In this paper, we focus on grammar schools and matura level. We analyse and compare the performance and content standards of school informatics across various levels of education. We continue on our previous post in which we focused mainly at elementary school.

## Kľúčové slová

Štátny vzdelávací program, základná škola, gymnáziá, maturita, analýza, hodnotenie

## Keywords

National educational program, elementary school, grammar school, matura, analysis, evaluation

## 1. ÚVOD

V predchádzajúcom príspevku [1] sme sa venovali novému štátnemu vzdelávaciemu programu z informatiky. V čase písania prechádzajúceho príspevku boli vypracované obsahové a výkonové štandardy pre prvý a druhý stupeň základnej školy. Na štandardoch pre gymnáziá sme práve pracovali<sup>1</sup>. Pripomeňme, že cieľom školskej informatiky je to, aby žiaci spoznali nástroje informatiky na spracovanie informácií a naučili sa pomocou nich riešiť problémy. Táto filozofia bola určujúca aj pri tvorbe nového štátneho vzdelávacieho programu [3]. V súčasnosti sú spracované štandardy aj pre gymnáziá. Tie v našom príspevku analyzujeme a hodnotíme.

## 2. PREHĽAD OBLASTÍ A TÉM

Gymnaziálne štandardy obsahujú nasledovné oblasti a témy:

- *Reprezentácie a nástroje* – práca s grafikou, práca s textom, práca s prezentáciami, práca s multimédiami,

práca s tabuľkami, informácie, štruktúry (informácie a štruktúry sú prierezovými témami).

- *Komunikácia a spolupráca* – prezentovanie informácií prostredníctvom webovej stránky, vyhľadávanie na webe, práca s nástrojmi na spoluprácu a zdieľanie informácií, práca s nástrojmi na komunikáciu.
- *Algoritmické riešenie problémov* – analýza problému, jazyk na zápis riešenia, pomocou postupnosti príkazov, pomocou nástrojov na interakciu, pomocou premenných, pomocou cyklov, pomocou vetvenia, interpretácia zápisu riešenia, hľadanie a opravovanie chýb (analýza problému, jazyk na zápis riešenia, interpretácia zápisu riešenia, hľadanie a opravovanie chýb sú prierezové témy).
- *Softvér a hardvér* – práca so súborami a priečkami, práca v operačnom systéme, počítač a prídavné zariadenia, práca v počítačovej sieti a na internete, práca proti vírusom a špehovaniu.
- *Informačná spoločnosť* – bezpečnosť a riziká, digitálne technológie v spoločnosti, legálnosť používania.

Vidíme, že gymnaziálne štandardy majú rovnaké členenie, ako štandardy pre základnú školu.

Je potrebné poukázať na to, že výkonové a obsahové štandardy nepredstavujú ani metodiku, ani tematický plán. Teda nie je to postupnosť, v akej sa jednotlivé oblasti a témy vyučujú počas školského roka na hodinách informatiky. Tie bývajú väčšinou zostavené tak, že sa v rámci každej hodiny pokrývajú viaceré ciele z rôznych oblastí. Napríklad, ak žiakov učíme používať nástroje typu Google Docs, nielen že tým naplníme niektoré ciele v oblasti *Reprezentácie a nástroje* – *práca s textom*, ale aj v oblasti *Komunikácia a spolupráca* – *práca s nástrojmi na spoluprácu a zdieľanie informácií* (výkon „*žiaci vytvárajú a upravujú zdieľané produkty pomocou nástrojov na spoluprácu*“), a aj niektoré ciele v oblasti *Informačná spoločnosť* – *bezpečnosť a riziká* (napríklad výkon „*žiaci zabezpečujú svoje údaje a komunikáciu proti zneužitiu*“). Takto sa dá na štátny vzdelávací program z informatiky nazerať aj pri iných témach.

## 3. POROVNANIE GYMNAZIÍ SO ZÁKLADNOU ŠKOLOU A MATURITOU

Porovnali sme obsahový a výkonový štandard pre základnú školu s gymnáziami (gym.) a maturitnou úrovňou (mat.). Prehľad výsledkov uvádzame v tabuľke 1.

<sup>1</sup> Členovia predmetovej komisie [2] v zložení: Z. Kubincová, M. Jurik, Ľ. Salanci, A. Blaho, E. Hanulová, V. Matisková, P. Kučera, E. Rausová, H. Cikatricisová, O. Székely.

**Tabuľka 1. Značkami je znázornená hustota / náročnosť konceptov v jednotlivých ročníkoch.**

Názov oblasti \ ročník, stupeň vzdelania	2. – 4.	5. – 6.	7. – 8.	gym.	mat.
<b>Reprezentácie a nástroje</b>					
– práca s grafikou	•	•	○	○	○
– práca s textom	•	•	○	○	○
– práca s príbehmi	•				
– práca s prezentáciami		•	○	○	○
– práca s tabuľkami		•	•	•	○
– práca s multimédiami	•		•	•	○
– hodnotiť nástroje informatiky					•
– informácie	•	•	•	•	○
– štruktúry	•	•	•	•	○
– typy údajov a operácie					•
<b>Komunikácia a spolupráca</b>					
– práca s webovou stránkou	•	•	○		
– prezent. inf. prostredníctvom web stránky				•	○
– vyhľadávanie na webe	•	•	○	○	○
– práca s nástrojmi na spoluprácu a zdieľanie inf.				•	○
– práca s nástrojmi na komunikáciu	•	•	•	○	○
<b>Algoritmické riešenie problémov</b>					
– analýza problému	•	•	•	•	○
– interaktívne zostav. riešenia	•				
– jazyk na zápis riešenia		•	•	•	•
– pomocou postupnosti príkazov	•	•	○	○	○
– pomocou cyklov		•	•	•	•
– pomocou vetvenia			•	•	•
– pomocou nástrojov na interakciu			•	•	•
– pomocou premenných			•	•	•
– pomocou záznamov					•
– pomocou jedno a dvojrozmerných polí					•
– pomocou súborov					•
– pomocou podprogramov s parametrami					•
– interpretácia zápisu riešenia	•	•	•	•	○
– hľadanie a opravovanie chýb	•	•	•	•	○
– posúdenie a hodnotenie riešenia					•
<b>Softvér a hardvér</b>					
– práca so súbormi a priečkami	•	•	•	○	○
– práca v operačnom systéme	•	•	•	•	○
– počítač a prídavné zariadenia	•	•	•	•	•
– práca v počítačovej sieti a na internete	•	•	○	○	○
– práca proti vírusom a špehovaniu		•	•	•	○
<b>Informačná spoločnosť</b>					
– Bezpečnosť a riziká	•	•	•	•	○
– Digitálne technológie v spoločnosti	•	•	○	○	○
– Legálnosť používania	•	•	•	○	○

Vysvetlenie značiek v tabuľke 1. Žiaci:

- ... Začínajú pracovať, zbierať zručnosti opatrne sa zoznamujú s informatickými konceptmi.
- ... Spoznávajú základy práce, kladie sa zvýšený dôraz na nové informatické koncepty, poznatky.
- ... Smelo používajú, trénujú nadobudnuté vedomosti (udržiavajú sa v kondícii, málo nových cieľov).

Do nasledujúcej tabuľky 2 sme vybrali tému *práca s textom* z oblasti *Reprezentácie a nástroje*. Tak môžeme porovnať výkony a obsahy, ktoré očakávame od žiakov na jednotlivých stupňoch vzdelania.

**Tabuľka 2. Reprezentácie a nástroje – práca s textom.**

Výkonový štandard:	Obsahový štandard
<p>Žiaci na konci 4. ročníka:</p> <ul style="list-style-type: none"> <li>• začínajú používať konkrétne nástroje editora na tvorbu a úpravu textu</li> <li>• zašifrujú a rozšifrujú text podľa jednoduchých pravidiel (reprezentujú znaky a slová)</li> </ul>	<p><i>Pojmy:</i> malé a veľké písmeno, znak, slovo, veta, symboly, číslica, znaky ako písmená, číslice, špeciálne znaky a symboly</p> <p><i>Vlastnosti a vzťahy:</i> slovo ako skupina písmen, veta ako skupina slov, odsek ako skupina viet, medzery a oddeľovače, obrázok a text, formátovanie textu, písmo + typ, veľkosť, hrúbka a farba písma (t.j. zvýraznenia), zarovnanie odseku, text ako postupnosť znakov</p> <p><i>Procesy:</i> písanie na klávesnici, opravovanie, mazanie, výmena slov, vkladanie obrázkov (cez schránku), vloženie a odstránenie medzery (z chybného textu)</p>
<p>Žiaci na konci 6. ročníka:</p> <ul style="list-style-type: none"> <li>• používajú konkrétne nástroje editora na tvorbu a úpravu textu</li> <li>• volia vhodný nástroj na prácu s textom,</li> <li>• používajú nástroje na vyhľadanie a nahradenie textu</li> <li>• posudzujú vplyv formátovacích nástrojov a skrytých znakov na výsledný text a operácie s textom</li> <li>• skúmajú nové nástroje v konkrétnom editore</li> </ul>	<p><i>Pojmy:</i> schránka, odrážky a číslovanie, tabuľka</p> <p><i>Vlastnosti a vzťahy:</i> slovo ako skupina písmen, veta ako skupina slov, odsek ako skupina viet, medzery a oddeľovače, obrázok a text, formátovanie textu, písmo + typ, veľkosť, hrúbka a farba písma (t.j. zvýraznenia), zarovnanie odseku, obrázok ako súčasť textu, skryté značky ako súčasť textového dokumentu, obrázok ako súčasť textu, vplyv skrytých znakov na štruktúru aj výsledný vzhľad textu, znaky ako písmená, číslice, špeciálne znaky a symboly, text a hypertext (napr. na internete, v encyklopédii)</p> <p><i>Procesy:</i> vytvorenie, presúvanie, kopírovanie a vkladanie textu, vkladanie obrázkov zo súboru, kontrola pravopisu</p>
<p>Žiaci na konci 8. ročníka:</p> <ul style="list-style-type: none"> <li>• volia vhodný nástroj na prácu s textom</li> <li>• používajú nástroje na vyhľadanie a nahradenie textu</li> <li>• rozlišujú neformátovaný a formátovaný text</li> <li>• posudzujú vlastnosti textovej informácie podľa stanovených kritérií</li> </ul>	<p><i>Pojmy:</i> odrážky a číslovanie, tabuľka</p> <p><i>Vlastnosti a vzťahy:</i> skryté značky ako súčasť textového dokumentu, obrázok ako súčasť textu, vplyv skrytých znakov na štruktúru aj výsledný vzhľad textu, znaky ako písmená, číslice, špeciálne znaky a symboly, neformátovaný a formátovaný text (t.j., obsahuje aj informácie o zvýrazneniach, písme, odsekoch, zarovnaniach)</p> <p><i>Procesy:</i> vkladanie tabuľky</p>
<p>Žiaci na gymnáziu:</p> <ul style="list-style-type: none"> <li>• kombinujú texty a iné objekty (grafiku, tabuľky, ...)</li> <li>• využívajú nástroje na prácu s textom pri realizácii svojich projektov</li> <li>• navrhujú a hodnotia postupnosť operácií pri práci s textom</li> </ul>	<p><i>Pojmy:</i> odrážky a číslovanie, tabuľka</p> <p><i>Vlastnosti a vzťahy:</i> skryté značky ako súčasť textového dokumentu, obrázok ako súčasť textu, vplyv skrytých znakov na štruktúru aj výsledný vzhľad textu, znaky ako písmená, číslice, špeciálne znaky a symboly, neformátovaný a formátovaný text (t.j., obsahuje aj informácie o zvýrazneniach, písme, odsekoch, zarovnaniach)</p> <p><i>Procesy:</i> vkladanie tabuľky</p>
<p>Žiaci na maturite sa vedia:</p> <ul style="list-style-type: none"> <li>• rozhodnúť sa pre nástroj</li> </ul>	<p><i>Pojmy:</i> textový editor</p> <p><i>Procesy:</i> rôzne spôsoby vytvárania, modifikovania, spracovania, prezentovania</p>



V ďalšej tabuľke 3 vidíme, ako sa mení dôraz na algoritmické riešenie vzhľadom na jednotlivé stupne vzdelania. Zlúčili sme tému *interaktívne zostavovanie riešenia* (3. – 4. ročník) a na to nadväzujúcu tému *jazyk na zápis riešenia* (5. ročník až maturita).

**Tabuľka 3. Algoritmické riešenie problémov**

Výkonový štandard:	Obsahový štandard
<p>Žiaci na konci 4. ročníka:</p> <ul style="list-style-type: none"> <li>riešia problém priamym riadením vykonávateľa (napr. robot, korytnačka)</li> <li>aplikujú elementárne príkazy daného jazyka (zo slovníka príkazov) na riadenie vykonávateľa</li> </ul>	<p><i>Vlastnosti a vzťahy:</i> priamy príkaz - akcia vykonávateľa</p> <p><i>Procesy:</i> riadenie vykonávateľa v priamom režime, používať jazyk vykonávateľa</p>
<p>Žiaci na konci 6. ročníka:</p> <ul style="list-style-type: none"> <li>používajú jazyk na popis riešenia problému – aplikujú pravidlá, konštrukcie jazyka</li> </ul>	<p><i>Vlastnosti a vzťahy:</i> chybný zápis, konštrukcie jazyka ako postupnosti príkazov</p> <p><i>Procesy:</i> zostavovanie programu v jazyku na zápis algoritmov, spustenie programu</p>
<p>Žiaci na konci 8. ročníka:</p> <ul style="list-style-type: none"> <li>používajú jazyk na popis riešenia problému – aplikujú pravidlá, konštrukcie jazyka,</li> <li>používajú matematické výrazy v jazyku na zápis algoritmov,</li> </ul>	<p><i>Vlastnosti a vzťahy:</i> algoritmus – programovací jazyk, vstup – algoritmus – výsledok, chybný zápis, konštrukcie jazyka ako: postupnosť príkazov, cyklus s pevným počtom opakovaní, podmienený príkaz, pomenovaná postupnosť príkazov</p> <p><i>Procesy:</i> zostavovanie programu v jazyku na zápis algoritmov, spustenie programu</p>
<p>Žiaci na gymnáziu:</p> <ul style="list-style-type: none"> <li>používajú jazyk na zápis algoritmického riešenia problému (používajú konštrukcie jazyka, aplikujú pravidlá jazyka)</li> <li>používajú matematické výrazy pri vyjadrovaní vzťahov a podmienok</li> <li>rozpoznávajú a odstránia chyby v zápise</li> <li>vytvárajú zápisy a interpretujú zápisy podľa nových stanovených pravidiel (syntaxe) pre zápis algoritmov</li> </ul>	<p><i>Pojmy:</i> program, programovací jazyk</p> <p><i>Vlastnosti a vzťahy:</i> zápis algoritmu a vykonanie programu, vstup – vykonanie programu – výstup/akcia</p> <p><i>Procesy:</i> zostavenie programu, identifikovanie, hľadanie, opravovanie chýb</p>
<p>Žiaci na maturite:</p> <ul style="list-style-type: none"> <li>rozpoznať zo zadania úlohy vstupné informácie, popísať očakávané výstupy, výsledky, výsledné akcie</li> <li>sformulovať formálny zápis algoritmu podľa kritérií rôznych vykonávateľov</li> <li>sformulovať, v čom spočíva informatický problém</li> <li>usporiadať príkazy do príkazových konštrukcií podľa pravidiel</li> <li>rozpoznať a odstrániť chyby vo formálnom zápise algoritmu</li> <li>vytvoriť riešenie pomocou pripravených podprogramov</li> <li>upraviť riešenie úlohy vzhľadom na rôzne požiadavky a obmedzenia (programu zmeniť vstupy, výstupy, spôsob spracovania, obmedzenie sady príkazov, premenných, typov)</li> <li>zostaviť program, v ktorom sa vyhľadáva, modifikuje, vytvára, vypisuje, resp. vykresľuje</li> </ul>	<p><i>Pojmy:</i> algoritmus, programovací jazyk, vývojové prostredie, deklarácie (deklaračná časť) a príkazy (príkazová časť), problém, podproblém, zovšeobecňovanie, kritické situácie, obmedzenia</p> <p><i>Vlastnosti a vzťahy:</i> program (spustenie, vykonanie zastavenie), pravidlá zápisu (syntax – rezervované slová, identifikátory, konštanty, oddeľovače, programové konštrukcie), opakujúce sa vzory</p> <p><i>Procesy:</i> postupnosti príkazov (mechanizmus vyhodnocovania), opakovanie (typy cyklov, podmienky, počítadlo, telo cyklu, vnorené príkazy), vetvenie (typy podmienených príkazov, mechanizmus vyhodnocovania, vnorené príkazy), podprogramy, parametre (mechanizmus volania, lokálne premenné, parametre, návratová hodnota)</p>

Z tabuliek 1, 2 a 3 môžeme usúdiť, že nižšie stupne sú zamerané na to, aby žiaci získali základnú digitálnu gramotnosť. Pre maturanta je zase dôležitá oblasť algoritmického riešenia úloh. Dodajme ešte, že smerom k starším žiakom sa kladie dôraz na porozumenie princípom, na objavovanie všeobecných vzťahov medzi informáciami a na ich popis na abstraktnej úrovni.

Popri novom štátnom vzdelávacom programe z informatiky pre gymnáziá vstúpili do platnosti v školskom roku 2013/2014 aj nové cieľové požiadavky na maturitné skúšky z informatiky. Nová maturita už zohľadňuje zmeny, ktoré sa pripravovali a realizovali vo vzdelávacích štandardoch. Maturitné požiadavky však majú iné členenie, ako štátny vzdelávací programy. Aby sme mohli tabuľky 1, 2 a 3 zostaviť tak, aby sa do nich premietla aj maturitná úroveň, museli sme maturitné obsahy a výkony transformovať do štruktúry štátneho vzdelávacieho programu.

#### 4. UKÁŽKY

Výkonový a obsahový štandard v téme *Algoritmické riešenie problémov – pomocou cyklov* vyzerá nasledovne:

**Tabuľka 4. Algoritmické riešenie problémov – pomocou cyklov**

<p>Žiaci na gymnáziu:</p> <ul style="list-style-type: none"> <li>rozpoznávajú opakujúce sa vzory</li> <li>rozpoznávajú, aká časť algoritmu sa má vykonať pred, počas aj po skončení cyklu,</li> <li>riešia problémy, v ktorých treba výsledok získať akumulovaním čiastkových výsledkov v rámci cyklu</li> <li>riešia problémy, ktoré vyžadujú neznámy počet opakovaní,</li> <li>riešia problémy, v ktorých sa kombinujú cykly a vetvenia</li> <li>stanovujú hranice a podmienky vykonávania cyklov</li> </ul>	<p><i>Pojmy:</i> opakovanie, počet opakovaní, podmienka vykonávania cyklu, telo cyklu</p> <p><i>Vlastnosti a vzťahy:</i> ako súvisí počet opakovaní s výsledkom, čo platí po skončení cyklu</p> <p><i>Procesy:</i> vyhodnotenie hraníc/podmienky cyklu, vykonávanie cyklu</p>
--	---

Nasledujúce vysvetlenia s ukázkami nám pomôžu pri interpretácii uvedených cieľov:

- Žiaci by mali rozumieť tomu, na čo cyklus v programoch slúži.
- Štátny vzdelávací program nepredpisuje programovací jazyk, v ktorom sa má učiť programovanie. Táto voľba ostáva na učiteľovi. Každopádne treba zvoliť jazyk, v ktorom sa dajú uvedené ciele realizovať. Žiaci by mali poznať syntax príkazu cyklu v tom jazyku, ktorý sa rozhodne učiteľ používať:

Napríklad, v prípade Pascal je to:

```
for premenná := od to po do telo_cyklu
```

V prípade jazykov C:

```
for (inicializácia; test; zmena) telo
```

Python:

```
for premenná in sekvencia:
    telo_cyklu
```

- Žiaci by mali rozumieť tomu, že telo cyklu môže byť jeden príkaz alebo aj viac príkazov. Tie však musia byť umiestnené medzi `begin ... end`, ak žiakov učíme

programovať v jazyku Pascal; v zložených zátvorkách `{ ... }` v prípade C-čkových jazykov; alebo správne odsadené v prípade jazyka Python.

- Zo zadania úlohy, resp. pri jej riešení by mali žiaci sami prísť na to, čo sa opakuje (opakujúce sa vzory), koľko krát sa to opakuje, alebo, čo sa mení. Napríklad, ak treba vypísať 10 pozdravov, mali by mať predstavu o tom, že v programe bude potrebné nejaký príkaz 10 krát vykonať. Uvádzame príklady pre rôzne jazyky:

Pascal: `writeln('Ahoj')`

Delphi/Lazarus: `Memol.Line.Add('Ahoj')`

C jazyky: `cout << "Ahoj\n"`

Python: `print("Ahoj")`

- Žiaci by mali vedieť zapísať riešenie v tom jazyku, v ktorom sa učia programovať. Na objasnenie opäť uvádzame príklady pre rôzne jazyky:

Pascal: `for i := 1 to 10 do writeln('Ahoj')`

C jazyky: `for (i=1; i<=10; i++)
 cout<<"Ahoj\n";`

Python: `for i in range(11):
 print("Ahoj")`

- Žiaci by mali vedieť riešiť aj iné typy úloh, v ktorých sa použije viac cyklov za sebou (napr., ak používajú vhodné grafické knižnice, dá sa pomocou úsečiek a dvoch samostatných cyklov nakresliť mriežka). Alebo úlohy, v ktorých sa kombinuje cyklus s podmienkou (vypísať magické čísla do 100 – magické čísla sú napríklad také čísla, ktoré sú deliteľné 7; alebo treba zistiť ich počet; červenou farbou vypísať párne a čiernou nepárne čísla) a ďalšie.

#### 5. ZÁVER

Pri tvorbe štandardov sme cieľavedome dbali na to, aby sme nevnucovali nástroje ani programovacie jazyky. Škola, učiteľ, tvorca metodík alebo autor učebníc sa tak môže slobodne rozhodnúť, aký textový editor použije (MS WORD, OpenOffice Write, alebo iný). Aj výkony a obsahy z oblasti *algoritmického riešenia problémov* sme špecifikovali takým spôsobom, aby sa pri vyučovaní dali použiť rôzne programovacie jazyky, napríklad: Pascal (FreePascal alebo Delphi, Lazarus), Python, C alebo iný. Cenou za to sú všeobecnejšie formulácie, ktoré sme museli pri tvorbe jednotlivých dokumentov použiť.

V tomto článku sme porovnali jednotlivé stupne vzdelávania. Vidíme, že oblasť algoritmického riešenia úloh je veľmi dôležitou zložkou pre žiakov, ktorí budú z informatiky maturovať.

#### 6. LITERATÚRA

- [1] SALANCI, E.: Informatika v novom štátnom vzdelávacom programe. Zborník konferencie DidInfo2014. Banská Bystrica : Univerzita Mateja Bela, 2014. - ISBN 978-80-557-0698-6. - s. 143-148
- [2] Predmetové komisie: [www.statpedu.sk/sk/Organizacia-SPU/Poradne-organy/Predmetove-a-odborne-komisie.alej](http://www.statpedu.sk/sk/Organizacia-SPU/Poradne-organy/Predmetove-a-odborne-komisie.alej)
- [3] Štátny vzdelávací program: [www.statpedu.sk/sk/Inovovany-Statny-vzdelavaci-program.alej](http://www.statpedu.sk/sk/Inovovany-Statny-vzdelavaci-program.alej)

# Meranie motivácie žiakov ku výkonu

Mária Spišáková  
KZVI FMFI UK  
Mlynská dolina  
842 48 Bratislava  
Slovensko  
spisakov@gmail.com

Ľubomír Salanci  
KZVI FMFI UK  
Mlynská dolina  
842 48 Bratislava  
Slovensko  
salanci@fmph.uniba.sk

## ABSTRAKT

*Prečo majú žiaci gymnázií rôzny vzťah ku programovaniu? Akú úlohu vo vyučovaní zohráva motivácia a stimulácia žiakov pri vyučovaní programovania? Aká je škála anxiezy podporujúcej výkon a brzdiacej výkon u žiakov gymnázia? Aká je škála motivácie k výkonu? Keďže už dlhší čas učíme programovanie a pozorujeme znížený záujem o programovanie špeciálne na gymnáziu, tak sme sa rozhodli skúmať položené otázky.*

## ABSTRACT

*In this paper, we describe causes reduced interest in programming. What role to play in teaching motivation and stimulation of students in teaching programming? What is the range anxiety supporting power and braking performance in high school students? What is the range of motivation to exercise? We answer these questions in our article.*

## Kľúčové slová

Motivácia, programovanie, anxiozita podporujúca výkon, motivácia k výkonu

## Keywords

Motivation, programming, codeing, supporting performance anxiety, motivation to exercise

## 1. ÚVOD

Počas nášho pôsobenia ako učiteľa informatiky sme pozorovali zníženie záujmu žiakov gymnázia o informatiku, špeciálne o algoritimizáciu a programovanie. Pri nástupe žiakov na strednú školu je v povedomí žiakov zakotvené, že programovanie je ťažké, že mu rozumejú len tí žiaci, ktorí majú prirodzený talent na programovanie.

Dávnejšie (asi 15 rokov), keď prístup žiakov na internet bol obmedzený, keď minimálny počet rodín malo pripojenie na internet, prístup na internet v škole bol silný motivačný aspekt, ktorý spôsobil záujem žiakov o informatiku, o programovanie, o tvorbu webových stránok a pod. Súčasný stav je úplne opačný. Žiaci automaticky využívajú všetky vymoženosti informačných technológií doma aj v škole, stávajú sa konzumenti obsahu IT a nie ich tvorcovia. Len malé percento z nich prirodzene skúma pozadie IT, fungovanie technológií a pod.

V našom výskume sa zameriavame na pôsobenie učiteľa na žiakov, pričom výsledkom výskumu má byť ich motivovanie ku tvorbe aplikácií prostredníctvom programovania. Myslíme si, že práve programovanie a algoritimizácia je najsilnejší nástroj pre riešenie problémov v IT, ktorý rozvíja u žiakov algoritmické myslenie nevyhnutné pre život. Vo verejnosti je všeobecne známe, aký veľký nedostatok pracovných síl je na v oblasti IKT, aké široké možnosti majú absolventi odborov zameraných na IKT, počítačové siete a pod. A predsa, ani tieto argumenty nie sú dôvodom, prečo by si mali zvoliť pre ďalšie štúdium odbory IKT.

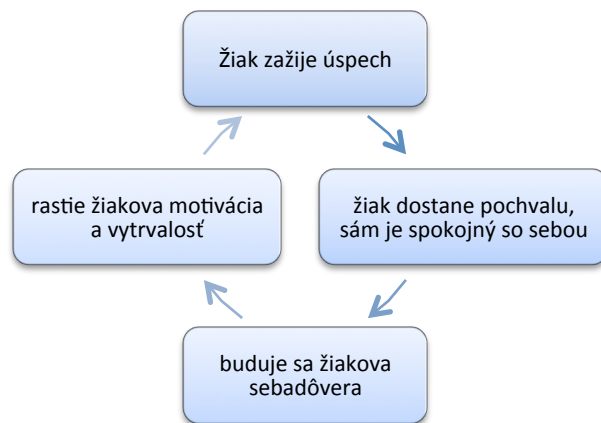
## 2. MOTIVAČNÉ FAKTORY

Všetci učitelia chápu, že ich pôsobenie na žiakov bude efektívne len vtedy, keď sa žiaci budú chcieť učiť. Ak sa nebudú chcieť učiť, tak ich učenie môže byť také neefektívne až do takej miery, že sa nič nenaučia.

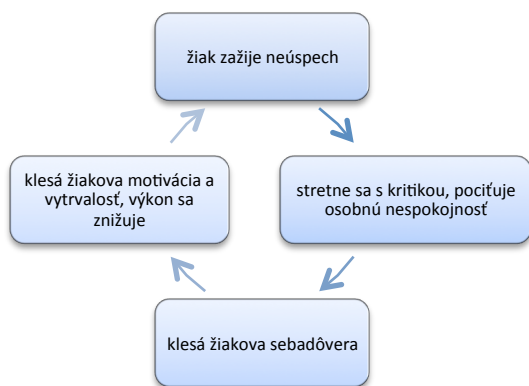
1. Aké sú teda motivačné faktory pre efektívne učenie? Podľa [1] sú nimi: užitočnosť poznatkov
2. získaná kvalifikácia
3. úspech pri učení (teda, prechádzajú cyklom: úloha – úspech – ocenenie – nová úloha)
4. pozitívna reakcia u učiteľa a spolužiakov
5. alebo nepríjemná reakcia u učiteľa a spolužiakov
6. zaujímavé veci – pri vyučovaní musíme vzbudiť žiakovu zvedavosť
7. vyučovanie je zábavné

Úspech pri učení je najsilnejší faktor, ktorý dodá sebadôveru a sebavedomie. Toto je spínačom, ktorý aktivuje ľudské schopnosti. Keď veríte, že niečo dokážete, alebo keď ste presvedčení, že niečo nedokážete, tak vždy máte pravdu.

Pri vyučovaní musíme zohľadniť dva začarované kruhy – dosiahnutie úspechu a neúspechu v štúdiu:



Obrázok: 1 Zobrazenie kruhu “dosiahnutie úspechu”



**Obrázok: 2 Začarovaný kruh neúspechu v štúdiu**

Podľa [1] Pokroky robia iba tí, ktorí si veria, ktorí zažívajú úspech.

Ďalšie motivačné faktory:

- Zvládnuteľnosť a primeraná náročnosť zadávaných úloh. Žiaci si nebudú vážiť pochvalu za splnenie úlohy, ak sa im zdá jej splnenie triviálne. Zložitost' úloh musí postupne narastať.
- Zábavnosť a zaujímavosť vyučovania, vyvolanie zvedavosti u žiakov, aby žiaci riešili úlohy, ktoré ich zaujímajú. príklady z ich života, z ich záujmov (futbal, šport, hudba),

Učiteľ si musí uvedomiť motivačné faktory, ktoré budú stimulovať žiakov podávať vysoké výkony pri štúdiu. Tieto faktory by mali byť prítomné na každej vyučovacej hodine:



**Obrázok: 3 Motivačné faktory používané učiteľmi [2]**

### 3. Dotazník motivácie k výkonu

Aj počas nášho výskumu plánujeme stimulovať výkon žiakov vyššie spomínanými motivačnými faktormi. Potrebujeme však na úvod nejakým spôsobom zistiť, čo motivuje našich žiakov. Ako vhodný sa nám javil dotazník výkonovej motivácie autorov Tomáša Pardela, L. Maršálkovej a A. Hrabovskej [3]. Tento dotazník obsahuje 52 položiek, obsahuje tri subškály na meranie všeobecnej výkonovej motivácie, stresovej situácie (anxiety) podporujúcej výkon a stresovej situácie brzdiacej výkon. Subškála výkonovej motivácie zachytáva aspekty výkonového správania, aspiračnej úrovne, vytrvalosti v práci a časovej orientácie do budúcnosti.

Dotazník sme dali v troch ročníkoch študentom Gymnázia Pavla Horova. Na dotazník odpovedalo 67 respondentov. Prvákov bolo 8, druhákov 34 a tretíakov 25.

Otázky podľa autorov sú rozdelené do troch škál

1. škála vnútornej motivácie k výkonu
2. škála stresových situácií brzdiacich výkon
3. škála stresových situácií podporujúcich výkon

Do škály motivácie k výkonu sú zaradené otázky typu: „Ľudia sa nazdávajú, že pracujem viac ako treba“, alebo „Keď po prvý raz nedosiahnem cieľ, ktorý som si vytýčil, vynaložím všetko úsilie na to, aby sa mi to predsa podarilo“.

Anxieta je úzkosť, strach, stres ktorý žiak cíti od okolia. Niektoré anxiety žiaka môžu paralizovať, reakcia na nich je nečinnosť, pasivita a niektoré môžu pôsobiť na žiaka naopak povzbudivo, zvišia jeho výkon, napríklad termín na odovzdanie práce, suťaž medzi spolužiakmi a podobne.

Do škály stresovej situácie brzdiacej výkon patria otázky ako sú: „Dostávala som sa takmer do panickej nálady, keď sa v škole bez prípravy písali triedne úlohy (písomky)“, alebo „Pri ľahkých pocitoch strachu klesá moja schopnosť reagovať“.

Do škály stresovej situácie podporujúcej výkon patria otázky napr.: „Môžem lepšie študovať, keď som v miernom napätí“ alebo „Zo skúsenosti viem, že pocity strachu pomáhajú“.

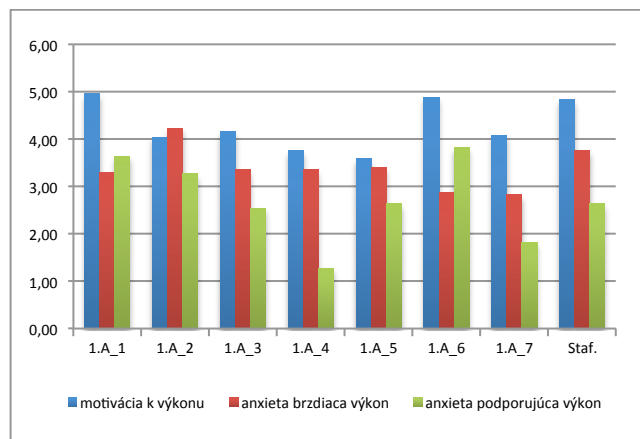
Aspekty výkonového správania, aspiračnej úrovne, vytrvalosti v práci a časovej orientácie do budúcnosti, ktoré boli zahrnuté v otázkach dotazníka sme špeciálne nevyhodnocovali.

Dotazník obsahuje 52 oznamovacích viet, pričom respondent si vyberá na základe číselnej stupnice od 1 do 6. Stupeň 1 znamená – výrok pre mňa vôbec neplatí, stupeň 6 znamená – výrok pre mňa úplne platí.

### 3.1 Vyhodnotenie dotazníka výkonovej motivácie

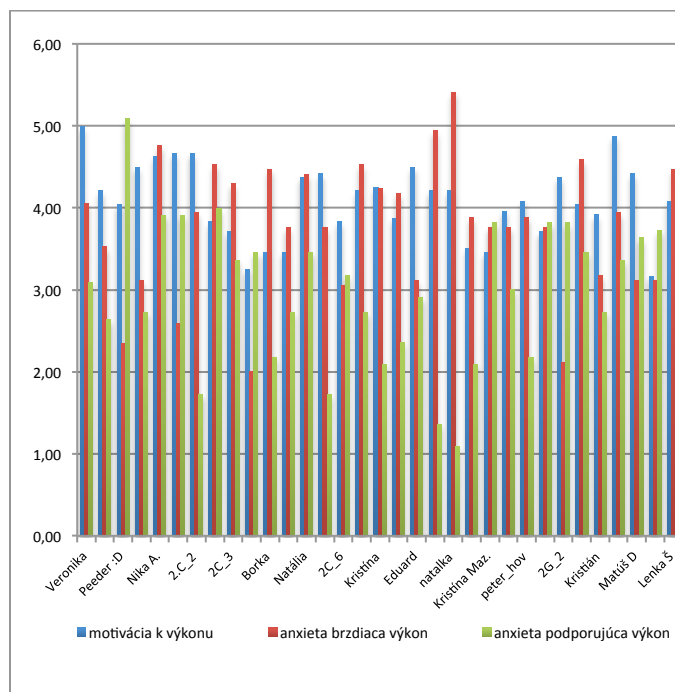
Dotazník sme pripravili v prostredí Google formulárov, vohodnocovali sme ho v prostredí tabuľkového kalkulatora. Respondenti sa mohli podpísať, alebo napísať iba prezývku.

Respondentov sme s výsledkami individuálne oboznámili. V grafe na osi x sú skratky, označenia jednotlivých žiakov.



**Obr.: 4 Prehľad výsledkov dotazníka žiakov 1. ročníka**

Podľa výsledkov dotazníka u žiakov 1. ročníka vidieť, že u nich prevážuje motivácia k výkonu, nad ostatnými škálami. Prváci väčšinou sú sami dobre motivovaní už prijatím na strednú školu, kde chcú dosiahnuť čo najlepšie študijné výsledky. U väčšiny z nich stresové situácie, ktoré podporujú výkon majú nižšie bodové hodnoty ako stresové situácie brzdiace výkon. Všetkých zapojených do hodnotenia nie je dostatočná vzorka, aby sa dalo toto zistenie zovšeobecniť. V ďalších rokoch ich štúdiu, kedy ich budeme stimulovať k výkonu špecifickými metódami, bude pre nás dôležité pozorovať, či sa tieto hodnoty posunuli k želaným výsledkom.

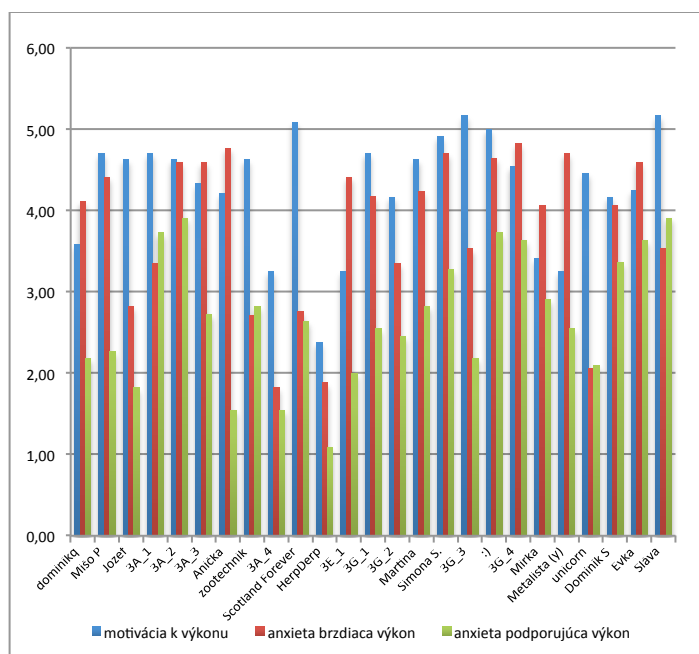


Obr.: 5 Prehľad výsledkov dotazníka žiakov 2. ročníka

Podľa výsledkov dotazníka u žiakov 2. ročníka sa dá usudzovať, že motivácia k výkonu je už potlačená anxiétami brzdiacimi výkon. To znamená, že žiaci sa viac boja a tento strach znižuje ich výkon. Zaujímavé sú výsledky žiakov *Natalka*, u ktorej vyskočili anxiety brzdiace výkon vysoko nad priemer bodovej stupnice a *Peeder :D*, u ktorého vysoko nad priemer vyšla škála anxiety podporujúcej výkon. To znamená, že najlepší výkon podáva pod tlakom, v napätí, resp. vtedy, keď má málo času, na poslednú chvíľu. A súčasne jeho škála anxiety brzdiacej výkon má podpriemernú hodnotu.

Výsledky v treťom ročníku vidieť na nasledujúcom grafe. V ňom sú zaujímavé výsledky konkrétnych žiakov: *Scotland Forever* – má vysokú motiváciu k výkonu, pričom všetky škály anxiety sú podpriemerné. Zdá sa, že tento žiak je imúnny proti stresorom, ktoré ho môžu buď kladne alebo záporne stimulovať. Keďže sme tohto žiaka identifikovali zo skúsenosti môžeme povedať, že tento výsledok ho charakterizuje. Mohli by sme ho nazvať tvrdohlavý žiak, ktorý si stojí za svojim názorom a všetko musí mať vyargumentované.

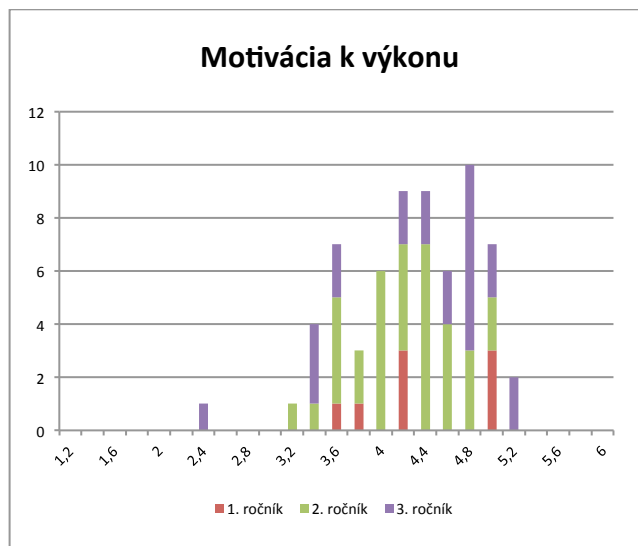
Ďalší zaujímavý výsledok je u žiaka *zootechnik*, avšak s menším rozdielom medzi motiváciou k výkonu a anxiétami. Tohto žiaka sme nedokázali identifikovať, takže nevieme povedať, či podobná charakteristika na neho platí.



Obr.: 6 Výsledky žiakov 3. Ročníka

### 3.1.1 Vyhodnotenie škály dotazníka k výkonu

Pri vyhodnocovaní dotazníka sme určili priemer stupnice odpovedí otázok pre skúmané tri škály motivácií. Chceli sme určiť početnosti priemerov žiakov pre jednotlivé škály. Tieto početnosti sme zobrazili graficky.



Obr.: 7 Početnosti priemerov stupnice motivácie k výkonu

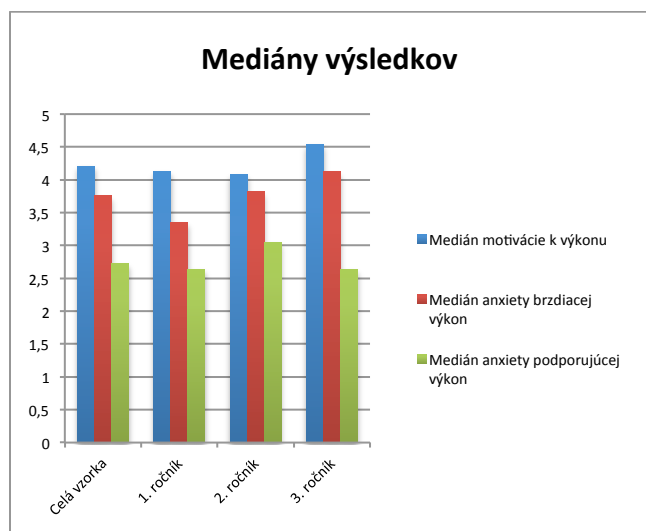
Na grafe sme dostali skoro normálne rozdelenie početnosti, pričom celá škála je posunutá smerom doprava. Znamená to, že naši študenti majú vysokú vnútornú motiváciu k dosahovaniu dobrých študijných výsledkov, s mediánom 4,21 pre všetky tri ročníky.



Tabuľka 1. Prehľad mediánov pre jednotlivé škály výkonu

	Medián motivácie k výkonu	Medián anxiety brzdiacej výkon	Medián anxiety podporujúcej výkon
Celá vzorka	4,21	3,76	2,73
1. ročník	4,13	3,35	2,64
2. ročník	4,08	3,82	3,05
3. ročník	4,54	4,12	2,64

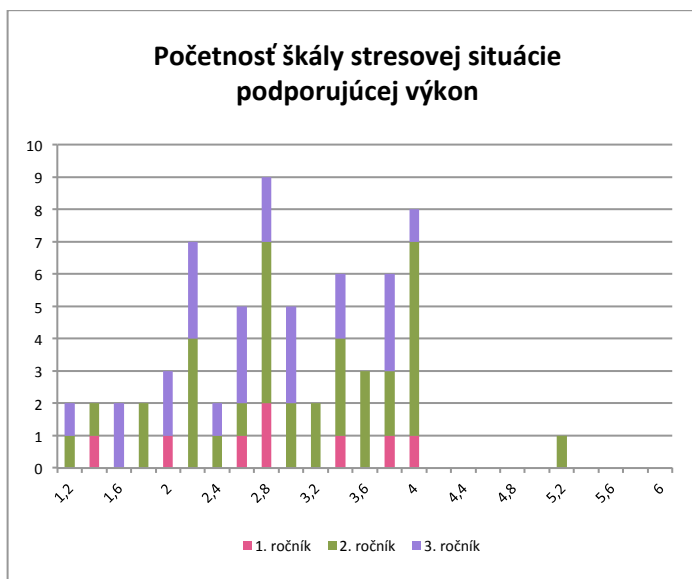
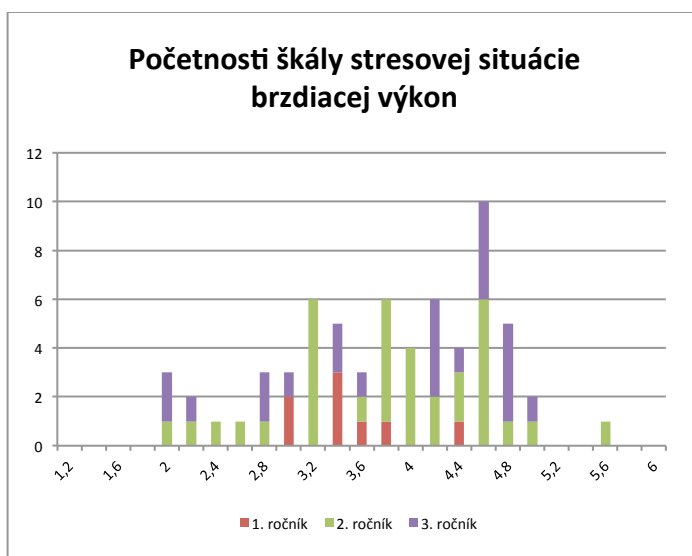
Grafické znázornenie mediánov v závislosti od ročníkov je názornejšie a zaujímavejšie.



Podľa mediánu pre anxiety brzdiacu výkon a súčasne pohľadom na graf môžeme vyvodit' záver, že žiaci školy sú citliví na stresory, ktoré u nich brzdia výkon, neradi pracujú pod napätím. Na grafe sa objavil skoro lineárny nárast tejto anxiety od prvého ročníka po tretí. Zdá sa nám, že žiaci s narastajúcimi skúsenosťami horšie znášajú stresové faktory, ktoré brzdia ich výkon.

Ak porovnáme medián škály anxiety podporujúcej výkon, tak ten je posunutý k hodnotám pod priemerným. Medián za celú vzorku je 2,73. Interpretujeme to ako nesúhlas s tvrdením, že mierny stres zlepšuje výkon žiakov.

Pritom žiaci druhého ročníka vnímajú anxiety podporujúcu výkon najlepšie zo všetkých žiakov. Pozrime si ešte grafy početností jednotlivých anxiet.



Vyhodnotenie dotazníka motivácie k výkonu sme považovali za dobrý štart pre začatie nášho výskumu.

Celý výskum chceme ale zamerať na sledovanie výkonov po našej intervencii. Ako sa budú meniť postoje žiakov ku programovaniu počas troch rokov štúdia na našej škole?

Momentálne pripravujeme dotazník, ktorý by mal súmať motiváciu výkonov žiakov v programovaní.

## 4. ZÁVER

Zatiaľ sme len na začiatku našej cesty, ktorá by mala viesť ku zlepšeniu výkonov v programovaní, aby sme dosiahli to, že študenti budú autormi nových myšlienok, nových algoritmov, nových prístrojov, automatov a robotov. Zdá sa nám, že budúcnosť je aj v tom, aby sa myšlienka schopnosti tvorby algoritmov a programov stala všeobecne rozšírenou, aby sa rozšírilo povedomie, že schopnosť písať efektívne programy,

DIDINFO 2015, Univerzita Mateja Bela, Banská Bystrica, 2015.

navrhovať šikovné riešenia nie je daná iba vyvolenému úzkemu okruhu ľudí ale je to schopnosť, ktorá sa dá naučiť.

Zatiaľ ešte nevieme presne odhadnúť ako sa namerané hodnoty škál výkonovej motivácie sa budú meniť, ale chceme sledovať, túto zmenu na rovnakej vzorke žiakov v priebehu ich štúdia na našej škole pôsobením motivačných stimulov, ktoré budú postupne do vyučovania zaradzované.

## **5. BIBLIOGRAFIA**

- [1] G. Petty, Moderní vyučování, Praha: Portál, 2008.
- [2] G. Petty, „Moderní vyučování,“ Praha, Portál, 2008, p. 54.
- [3] L. Maršálová, A. Hrabovská a T. Pardel, „Dotazník výkonovej motivácie,“ 1984.

# Informatické kompetencie žiakov ZŠ so zrakovým postihnutím a ich využitie na vyučovaní matematiky a fyziky používaním matematického editora LAMBDA

Mgr. Mária Kaliaková  
SŠI ZŠ pre žiakov so zrakovým  
postihnutím  
Svrčia 6  
842 11 Bratislava  
Slovenská republika  
maria.kaliakova@bee.sk

Mgr. Mária Stankovičová  
Univerzita Komenského v Bratislave  
Centrum podpory študentov so  
špecifickými potrebami  
Mlynská dolina  
842 48 Bratislava  
Slovenská republika  
stankovicova@cezap.sk

Mgr. Zdenka Vitálišová  
SŠI ZŠ pre žiakov so zrakovým  
postihnutím  
Svrčia 6  
842 11 Bratislava  
Slovenská republika  
zdenka.vitalisova@bee.sk

## ABSTRAKT

V príspevku uvádzame prehľad o nadobudnutých informatických kompetenciách žiakov 5. až 9. ročníka základnej školy. Zameriame sa na tie kompetencie, ktoré sú potrebné pre používanie matematického editora LAMBDA. V krátkosti predstavíme editor LAMBDA a oboznámime s metodikou osvojovania si zručností žiakov. Uvedieme praktické skúsenosti používania editora LAMBDA na hodinách matematiky a fyziky.

## ABSTRACT

In this paper we present an overview of the acquired informatics competencies of pupils of 5th to 9th grade of primary school. We will focus on those competencies that are necessary for the use of LAMBDA mathematical editor. We introduce the editor LAMBDA and familiarize the methodology of acquisition skills of pupils. We show practical experience using LAMBDA editor for mathematics and physics.

## Kľúčové slová

žiaci so zrakovým postihnutím, informatické kompetencie, asistenčné technológie, matematický editor, elektronický zošit, medzipredmetové vzťahy

## Keywords

pupils with visual impairment, informatics competencies, assistive technologies, equation editor, electronic notebook

## 1. ÚVOD

Naše skúsenosti s používaním editora LAMBDA sú výsledkom dlhoročnej spolupráce Centra podpory študentov so špecifickými potrebami na UK a Základnej školy pre žiakov so zrakovým postihnutím. Formou vzdelávacích kurzov alebo krúžkovej činnosti sa s editorom LAMBDA oboznamovali nielen žiaci ale aj učitelia. Cieľovou skupinou viacerých vzdelávacích aktivít boli aj študenti stredných škôl. V príspevku sa však zameriavame hlavne na spoluprácu v prebiehajúcom školskom roku s učiteľmi a žiakmi základnej školy.

Úspešná spolupráca by sa však nemohla uskutočniť, ak by zúčastneným žiakom chýbali potrebné informatické kompetencie. Tie sú predpokladom k rýchlemu napredovaniu a intenzívnemu sústredeniu sa na samotný matematický editor.

Čo je však najdôležitejšie, žiaci sú schopní využiť nadobudnuté zručnosti počas vyučovania matematiky a fyziky. Táto skúsenosť im napomáha vnímať jednotlivé predmety komplexne v súvislostiach medzipredmetových vzťahov.

## 2. VZDELÁVANIE ŽIAKOV SO ZRAKOVÝM POSTIHNUTÍM

V roku 2008 bola na Slovensku prijatá nová školská reforma [1], podľa ktorej sa informatika stala povinným predmetom vo všetkých ročníkoch a na všetkých stupňoch základných a stredných škôl, s výnimkou prvého ročníka. Pre špeciálne školy začal zákon platiť od septembra 2009.

Súčasťou ŠVP<sup>1</sup> je Vzdelávací program pre žiakov so zdravotným znevýhodnením, konkrétne so zrakovým postihnutím [2]. Cieľom vzdelávania žiakov so zrakovým postihnutím v primárnom a nižšom strednom stupni je pripraviť žiakov na vzdelávanie na akejkoľvek strednej škole. Vo výchovno-vzdelávacom procese má mimoriadny význam používanie kompenzačnej techniky, najmä IKT<sup>2</sup>.

Mnohí žiaci so zrakovým postihnutím sú schopní použitím kompenzačných pomôcok plnohodnotne využívať počítač a IKT, a tak dosiahnuť obsahový aj výkonový štandard porovnateľný so žiakmi intaktnej populácie. Preto postupujú v predmete informatická výchova a informatika podľa štátneho vzdelávacieho programu pre bežnú populáciu [3].

Pre žiakov so zrakovým postihnutím má predmet informatika a informatická výchova nezastupiteľnú úlohu. Umožní im prístup k informáciám a vedomostiam, rozvinúť svoje osobné talenty. V neposlednom rade podstatne uľahčuje integráciu, ako aj ich plnohodnotné začlenenie sa do spoločenského života, osamostatnenie sa a tak aj zvyšovanie vlastného sebavedomia.

### 2.1 Špecifiká predmetu Informatická výchova a Informatika pre žiakov so zrakovým postihnutím

V ŠVP je obsahový a výkonový štandard v predmete informatická výchova a informatika modifikovaný [4]. Program je potrebné doplniť o prácu so špeciálnym softvérom – zväčšovačom obrazovky, prípadne čítačom obrazovky. Nevidiacim žiakom a žiakom so zvyškami zraku je potrebné prispôbiť vzdelávací program ich možnostiam a potrebám vylúčením práce s grafikou a zameraním sa na čo najdôkladnejšie ovládanie prostriedkov IKT a niektorých dôležitých aplikácií. Jednou z takýchto aplikácií je aj program (elektronický matematický zošit) LAMBDA.

<sup>1</sup> ŠVP – Štátny vzdelávací program

<sup>2</sup> IKT – Informačno-komunikačné technológie

Vzhľadom k tomu, že žiaci so zrakovým postihnutím ovládajú počítač hlavne prostredníctvom klávesnice a s pomocou asistenčných technológií ako braillovský riadok, čítač obrazovky, alebo zväčšovač obrazovky prijímajú výstupné informácie z počítača, je dôležité, aby sa naučili:

- zapnúť a vypnúť počítač (odhlásiť sa),
- vyhľadať, spustiť a ukončiť program,
- ovládať počítač klávesnicou, čo si vyžaduje zvládnutie techniky písania desiatimi prstami,
- ovládať aplikácie pomocou klávesových skratiek,
- orientovať sa v súboroch a priečiňkoch (usporiadanie, vytváranie a spravovanie),
- uložiť, otvoriť, vymazať, kopírovať, presúvať, premenovať vlastný súbor (priečiňok),
- pracovať s viacerými aplikáciami súčasne (prepínanie medzi oknami, minimalizovať a maximalizovať pracovné okno),
- nastaviť hlasitosť systému a zapojiť slúchadlá,
- ovládať zväčšovací softvér alebo čítač obrazovky – nastaviť potrebné nastavenia vhodné pre prácu na počítači,
- ovládať braillovský riadok.

Rôznorodosť zrakového postihnutia si vyžaduje odlišné postupy a obsah vzdelávacieho programu v predmete informatická výchova a informatika, preto sa v ŠVP odporúča deliť triedu pri vyučovaní na dve skupiny – slabozrakí a nevidiaci a žiaci so zvyškami zraku, do ktorých sú žiaci zadelení podľa druhu a stupňa zrakového postihnutia.

Žiaci, ktorí sú v tomto školskom roku na druhom stupni v 7. až 9. ročníku, majú predmet informatika od 5. ročníka. Žiaci 5. a 6. ročníka mali na prvom stupni predmet informatická výchova.

Časová dotácia pre jednotlivé ročníky umožňuje žiakom okrem spomínaných zručností naučiť sa ovládať také aplikácie, ktoré môžu využívať aj na ďalších predmetoch. Nakoľko každá aplikácia si vyžaduje rôzne zručnosti, treba sa jej zvládnutiu venovať podľa potreby. Významným faktorom je aj šikovnosť jednotlivých žiakov.

V tabuľke 1 je prehľad časovej dotácie podľa ŠVP, doplnenej o dotáciu podľa ŠKVP<sup>3</sup>.

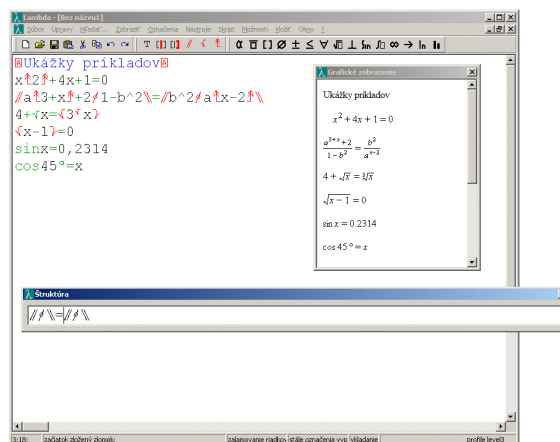
**Tabuľka 1. Časový rozsah výučby predmetu informatika v jednotlivých ročníkoch**

Ročník	Týždenne	Ročne
piaty	0,5+1	50
šiesty	0,5+1	50
siedmy	0,5+0,5	33
ôsmy	0,5+0,5	33
deviaty	0,5+0,5	33

<sup>3</sup> ŠKVP – Školský vzdelávací program

### 3. EDITOR LAMBDA – MATEMATICKÝ ELEKTRONICKÝ ZOŠIT PRE NEVIDIACICH ŽIAKOV

Žiaci druhého stupňa navštevujú krúžok Lambda, ktorý je zameraný na používanie matematického editora LAMBDA [5]. Editor LAMBDA (Linear Access to Mathematics for Braille Device and Audio-synthesis – Lineárny prístup k matematike pre braillovské zariadenia a hlasový výstup) je výsledkom medzinárodného projektu Lambda Project, ktorého cieľom bolo poskytnúť nevidiacim študentom nástroj, pomocou ktorého nielen matematický príklad zapišu, ale môžu aj prehľadne riešiť [6]. Editor LAMBDA možno teda považovať za elektronický zošit, do ktorého žiak zapisuje a z ktorého číta pomocou výstupu na braillovskom riadku, čítača obrazovky a zároveň je možné zápis zobrazíť v zabudovanom grafickom prehliadači.



**Obrázok 1. Ukážka editora s príkladmi matematického zápisu, grafického zobrazenia, kompaktnej štruktúry**

Výhodou editora je možnosť zobraziť matematický zápis v podobe, na akú je bežne učiteľ matematiky zvyknutý. Na úrovni učiva základnej školy je to dôležité hlavne pri zápise zlomkov, mocnín, odmocnín. Editor ponúka možnosti pre zápis symboliky aj vyššej matematiky (goniometria, limity, logaritmy...), ktorá je vhodná pre stredné a vysoké školy. Z dlhodobej perspektívy má teda význam začínať s editorom už na základnej škole. Osvojené zručnosti možno tiež považovať za prípravu na štúdium na strednej škole a neskôr aj na vysokej škole s matematickým a/alebo technickým zameraním.

#### 3.1 Pre koho je editor LAMBDA určený

Pre nevidiacich žiakov na hodinách matematiky a fyziky je editor výbornou pomôckou. Tak ako vidiaci žiaci používajú pri riešení úloh zošit a pero, tak nevidiaci žiaci zapisujú a riešia úlohy v editore LAMBDA.

Z praktických organizačných dôvodov je vhodné, aby boli žiaci oboznámení s editorom skôr ako ho začnú používať na vyučovacích hodinách. Žiaci sa s editorom oboznamujú formou krúžkovej činnosti.

#### 3.2 Krúžok Lambda

V súčasnosti krúžok navštevujú žiaci 5. – 9. ročníka, pričom sú rozdelení na dve skupiny (5. – 7. ročník a 8. – 9. ročník) vzhľadom na rôznu úroveň už nadobudnutých vedomostí. V predchádzajúcich školských rokoch prebiehali krúžky so zameraním na editor LAMBDA v spolupráci s Centrom podpory študentov so špecifickými potrebami UK a Katedrou základov

vyučovania informatiky FMFI UK, zväčša boli určené pre žiakov 8. a 9. ročníka [7]. Cieľom krúžkov bolo naučiť žiakov používať editor v rozsahu primeranom ich vedomostiam z matematiky [8].

Tematická náplň krúžku vychádza z časovo-tematických plánov z matematiky pre konkrétne ročníky. Tematický plán sa zostavuje v spolupráci s vyučujúcimi matematiky a fyziky na základe odporúčaní ŠVP Matematika [9] a Fyzika [10].

V tabuľke 2 je uvedený stručný prehľad tém z matematiky, ktorým sa venujú žiaci počas krúžkovej činnosti. Na konkrétnych príkladoch si precvičujú zápis symboliky k danej téme, tiež používanie klávesových skratiek.

**Tabuľka 2. Prehľad vybraných tém z matematiky pre krúžok zameraný na používanie editora LAMBDA**

Ročník	Téma
7.	Desatinné čísla
	Obsah, obvod štvorca a obdĺžnika
	Zlomky, početové výkony so zlomkami
	Percentá
	Objem a povrch kocky a kvádra
	Pomer, priama a nepriama úmernosť
8.	Celé čísla. Početové výkony s celými číslami
	Premenná, výraz, rovnica
	Zápis konštrukcie trojuholníka
	Obvod a obsah rovnobežníka, trojuholníka, lichobežníka
	Hranoly, ich objem a povrch
	Slovné úlohy na výpočet obsahu kruhu a obvodu kružnice
9.	Mocniny a odmocniny, zápis veľkých čísel
	Riešenie lineárnych rovníc a nerovníc
	Riešenie slovných úloh o pohybe a spoločnej práci
	Objem a povrch valca, ihlana, kužeľa
	Gul'a a rez gul'ou, objem a povrch gule
	Pytagorova veta
	Úprava celistvých a algebrických výrazov
	Početnosť, relatívna početnosť
	Aritmetický priemer

### 3.3 Metodika osvojovania zručností v používaní editora LAMBDA

Časová dotácia pre krúžok Lambda je 1 vyučovacia hodina týždenne – 45 minút, t.j. 33 hodín ročne. K dispozícii je počítačová učebňa, každý žiak pracuje samostatne. Braillovské riadky nie sú k dispozícii, žiaci pracujú len s hlasovým výstupom čítača obrazovky.

Najprv je potrebné žiakov oboznámiť s prostredím editora. Predpokladom pre efektívne používanie editora LAMBDA, je dobrá orientácia na klávesnici a skúsenosti s prácou v textovom editore. Žiaci druhého stupňa už disponujú týmito kompetenciami. [11]

Dôležité pre žiakov je uviesť si, že editor rozlišuje dve prostredia – matematické a textové. V každom prostredí reaguje

editor na vstupy z klávesnice iným spôsobom. Aby si žiaci zvykli na orientáciu v dokumente, je vhodné začať jednoduchými príkladmi, pri ktorých nie je nutné pamätať si klávesové príkazy okrem príkazu na vkladanie matematického a textového prostredia.

Neskôr sa žiaci oboznámia aj s inými klávesovými príkazmi podľa témy učiva, ktorému sa venujú, napríklad zlomky, symbolika v geometrii, mocniny, odmocniny a s používaním pomocných nástrojov editora. V tejto súvislosti treba zdôrazniť, že matematický zápis, ako ho poznajú vidiaci žiaci, má vizuálnu stránku (zlomková čiara, umiestnenie exponentu, znak odmocniny a podobne) a zároveň aj významovú stránku (čitateľ v zlomku je delený menovateľom, základ mocniny násobíme toľkokrát samého sebou, koľkokrát určuje exponent a podobne), teda konkrétne výpočty. Pomocou klávesových príkazov sú vkladané jednotlivé časti významovej stránky. Pre žiakov je potrebné sa ich naučiť používať v správnom poradí, aby ich mohol správne prečítať čítač obrazovky a tiež aby sa zápis ukázal správne v grafickom zobrazení.

Orientácia vo vlastnom výpočte je dôležitá, preto je u žiakov potrebné vytvoriť určité návyky v používaní niekoľkých praktických nástrojov, ktoré ponúka samotný editor:

- Dvojnásobné kopírovanie riadka – zachovanie postupnosti krokov pri výpočte. Žiak nemusí prepisovať celý zápis príkladu, v každom kroku mení len tú časť, ktorú práve počíta. Dajú sa ľahko sledovať a kontrolovať kroky výpočtu.
- Zobrazenie rozvinutej a kompaktnej štruktúry – vhodné pre ľahšiu orientáciu v dlhších a členitých zápisoch. Používa sa pri zložených zlomkoch.
- Kalkulačka – výpočet označeného výrazu v matematickom prostredí.
- Vloženie výsledku vypočítaného výrazu na miesto kurzora – predchádzanie chybným opisom kvôli nesprávnemu zapamätaniu.

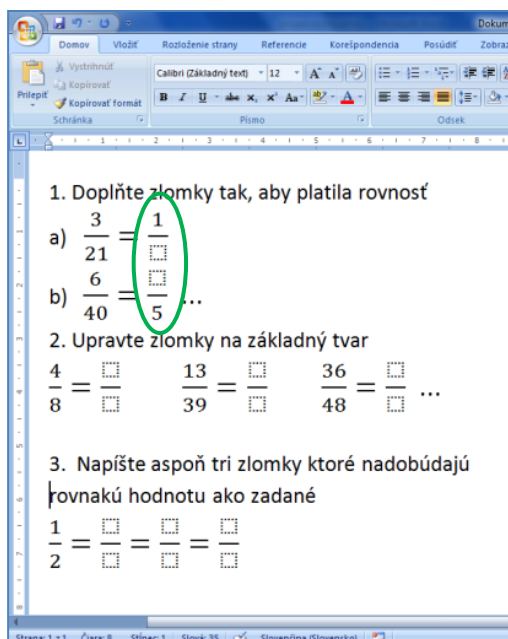
## 4. PRAKTICKÉ POUŽÍVANIE EDITORA LAMBDA NA HODINÁCH MATEMATIKY A FYZIKY

Obdobie rozmachu informačných technológií viedlo aj u žiakov so zrakovým postihnutím k radikálnym zmenám vo vzdelávaní predovšetkým čo sa týka prostriedkov. Notebooky vybavené špecifickým softvérom sa stali neodmysliteľnou pomôckou nevidiacich žiakov. Nahradili ťažké a hlučné pichtové stroje a niekoľko nie najľahších zakladačov naplnených braillovskými papiermi s poznámkami, ktoré žiaci potrebujú denne na vyučovaní.

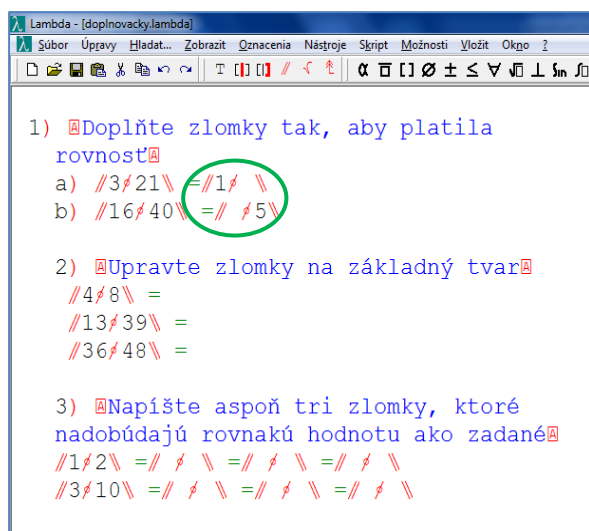
Používanie notebooku umožňuje zvýšiť pracovné tempo a vyrovnáť sa tempu ostatných spolužiakov. Takmer na všetkých hodinách si žiaci vystačia s textovým editorom, problém však predstavuje zložitejší matematický text. Doplnením softvérového vybavenia notebooku editorom LAMBDA získavajú žiaci skutočne univerzálny zošit i knihu.

Nevidiaci žiaci pracujú na hodinách matematiky v tomto prostredí počas celej vyučovacej hodiny. V úvode hodiny sa často opakuje prebrané učivo, aby sa mohlo naň nadviazať pri vysvetľovaní nového učiva. Aby tento proces prebiehal čo najefektívnejšie, sú pre žiakov pripravené pracovné listy, ktoré neraz obsahujú aj doplnňovacie cvičenia (Obrázok 2, Obrázok 3).





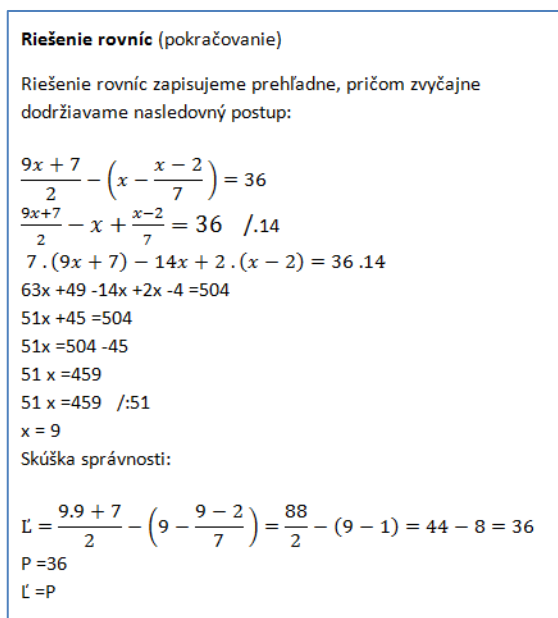
Obrázok 2. Ukážka pracovného listu z matematiky v programe Word



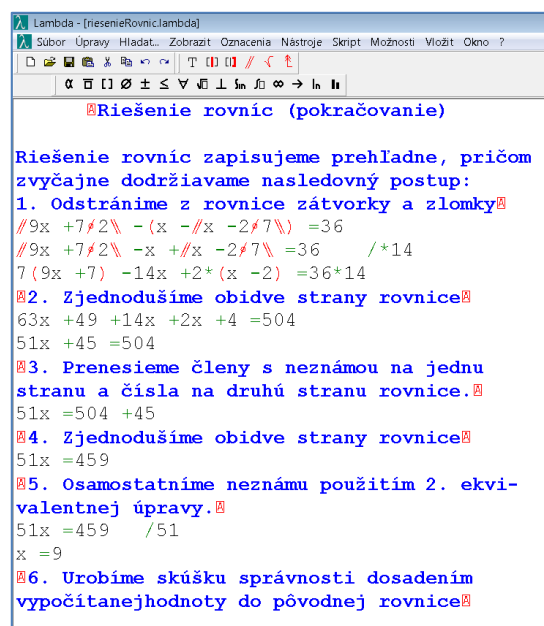
Obrázok 3. Ukážka pracovného listu z matematiky v programe Lambda

Pokiaľ by nevidiaci žiaci pracovali s pichtovým strojom, nedokázali by sa nastaviť na vhodné miesto. Museli by zadanie opísať a až potom začať riešiť, čo je časovo neefektívne. Ak dostanú zadanie v elektronickej podobe spracované v zošite LAMBDA, bez problémov sa nastavia na príslušný riadok a môžu svoje riešenie doplniť. Opisovaniu zadaní rôznych úloh sa vyhnú aj pri vypracovávaní rôznych testov či písomných prác.

Veľkou výhodou pri vysvetľovaní nového učiva je možnosť vopred pripraviť pre nevidiacich žiakov vzorové riešenie, ktoré ostatní žiaci sledujú na tabuli a nevidiaci žiaci ho sledujú krok za krokom súčasne s výkladom učiteľa v pripravenom súbore. Pracovný zošit tak nahrádza nevidiacemu žiakovi tabuľu (Obrázok 4, Obrázok 5).



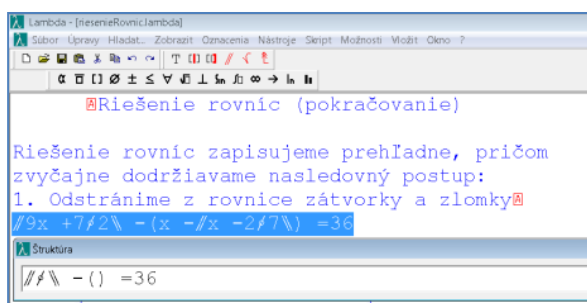
Obrázok 4. Ukážka riešeného príkladu na tabuli, učiteľ vysvetľuje jednotlivé kroky slovné



Obrázok 5. Ukážka riešeného príkladu pripraveného v programe Lambda s vysvetlením jednotlivých krokov

Využívanie textového módu programu LAMBDA umožňuje pripísať dôležitú poznámku k jednotlivým krokom, doplniť slovný komentár k riešeniu (Obrázok 5), čo uľahčí aj neskoršie utvrdzovanie vysvetleného učiva. Keďže poznámky sú uvedené v textovom režime, na čo čítač obrazovky upozorní, ľahko sa odlišia od samotného riešenia zadanej úlohy.

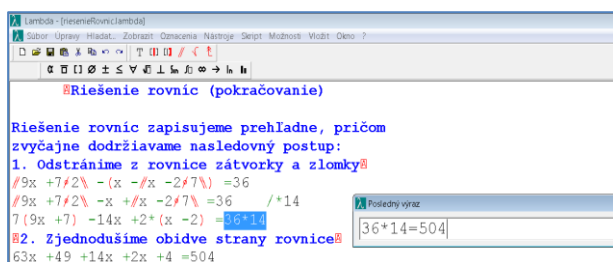
Pri úprave zložitejších zápisov, napríklad zložených zlomkov, ktoré majú v čitateli či menovateli viac matematických operácií, alebo zložitejších rovníc [12], je vhodné pre posúdenie jednotlivých krokov úpravy použiť náhľad na štruktúru – rozvinutú alebo kompaktnú. (Obrázok 6)



Obrázok 6. Ukážka rozvinutej a kompaktnej štruktúry

Využívanie náhľadu na štruktúru umožňuje ľahšie pochopiť zložitejšie štruktúry a rozhodnúť sa tak pre vhodný spôsob úpravy.

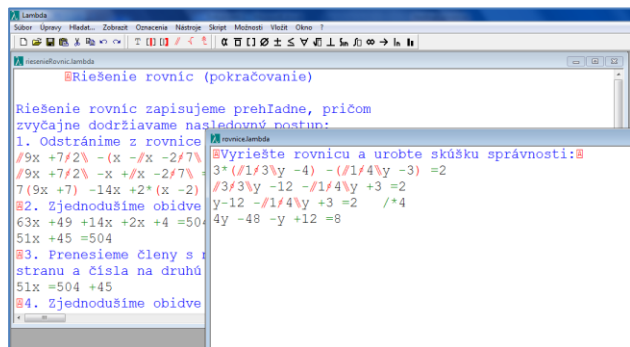
Zabudovaná kalkulačka nielenže šetrí čas, nie je potrebné prepínať medzi viacerými programami, ale umožňuje aj vypočítaný výsledok jednoducho vložiť na miesto, na ktorom je nastavený kurzor. Tento postup zefektívňuje prácu. Aby kalkulačka vypočítala výraz, je potrebné celý výraz dať do bloku, následne pomocou skratky vyvolať aplikáciu kalkulačka (Obrázok 7).



Obrázok 7. Kalkulačka

Výsledná hodnota sa uloží do pamäte, preto je možné túto hodnotu následne vložiť na miesto kurzora v ďalšom riadku.

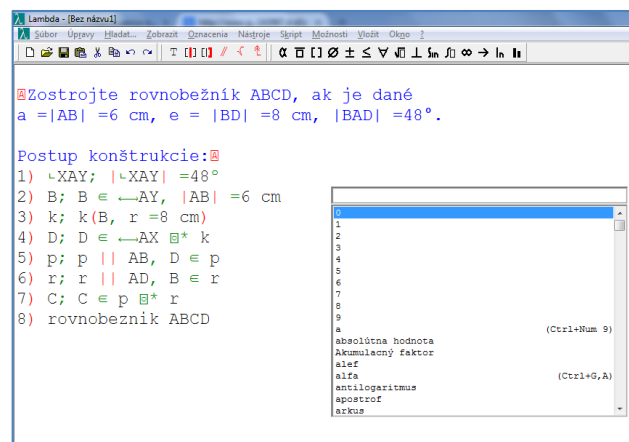
Editor LAMBDA umožňuje mať otvorených niekoľko samostatných okien – zošitov a prepínať sa medzi nimi. Táto funkcia je vhodná predovšetkým pri precvičovaní a utvrdzovaní vysvetleného učiva. Podľa vzorového príkladu môžu byť riešené nové podobné zadania a jednoduchým prepnutím medzi oknami kontrolovaná správnosť postupu. Tento proces zohráva významnú úlohu aj pri individuálnom osvojovaní si nového učiva. (Obrázok 8).



Obrázok 8. Možnosť pracovať v niekoľkých súboroch

Nemenej dôležitá pre nevidiacich žiakov je možnosť vrátiť sa v prípade nepochopenia určitého kroku o riadok i viac späť a opäť si informáciu prečítať.

Matematický zošit LAMBDA využívajú nevidiaci žiaci aj na hodinách geometrie. Zošit má svoje miesto nielen v planimetrii a stereometrii, ale aj v konštrukčnej geometrii [13]. Samozrejme, žiaci v tomto zošite nemôžu rysovať. Na rysovanie používajú gumenú podložku s fóliou, ale postup konštrukcie zaznamenávajú v LAMBDE. Pri týchto zápisoch využívajú v menu ponuku vložiť geometrický symbol alebo klávesom F5 otvoria okno umožňujúce dynamické vyhľadávanie a vkladanie prvkov (Obrázok 9). Táto funkcia umožňuje žiakom rýchlo vkladať rôzne matematické symboly bez toho, aby si žiaci museli pamätať ich skratky.

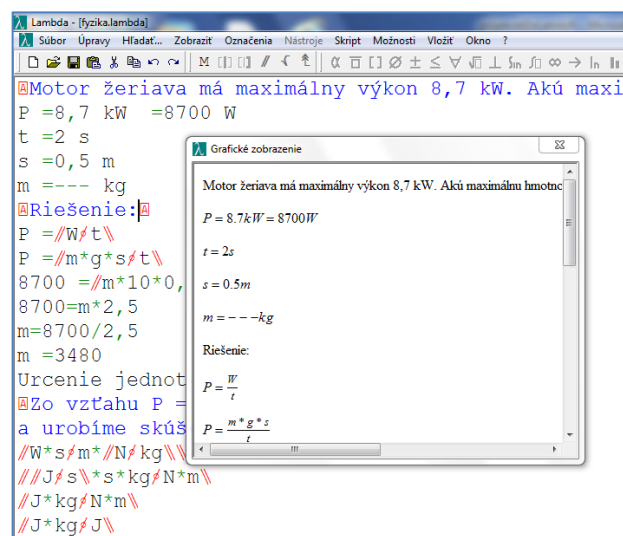


Obrázok 9. Dynamické vyhľadávanie geometrických symbolov

Napriek tomu, že program má množstvo výhod a poskytuje používateľom so zrakovým postihnutím celý matematický aparát od ZŠ po VŠ, niečo mu chýba – tvorba a vyplňovanie tabuliek nie je v tomto prostredí možná. V tomto prípade je nevyhnutné využívať vhodné textové editory.

Ako už bolo spomenuté v úvode, matematický zošit LAMBDA využívajú žiaci so zrakovým postihnutím aj na hodinách fyziky predovšetkým pri výpočtových úlohách [14].

Významným prvkom zošita je možnosť komunikovať s vidiacimi spolužiakmi i učiteľmi pomocou grafického zobrazenia zošita a exportu do webovej stránky (Obrázok 10).



Obrázok 10. Zobrazenie v zabudovanom grafickom prehliadači

## 4.1 Skúsenosti a názory žiakov spojené s používaním editora LAMBDA

Predpokladom k tomu, aby žiaci mohli plnohodnotne využívať editor LAMBDA na vyučovaní a tiež počas domácej prípravy, je nutnosť mať ho nainštalovaný na notebooku, ktorý môžu používať v škole aj doma. Kým vlastné notebooky nemajú, tak počas krúžku používajú počítač v školskej učebni, no počas vyučovania alebo pri domácej príprave sú znovu odkázaní na pichťov stroj.

V súčasnosti je notebook už takmer neodmysliteľnou pomôckou pre každého žiaka so zrakovým postihnutím. Žiaci nosievajú notebooky aj na krúžok. Konzultujú problémy, ktoré sa im vyskytli počas vyučovania. Vo všeobecnosti prijímajú editor LAMBDA veľmi pozitívne. Uvedomujú si, že čím lepšie sa ho naučia ovládať, tým viac im pomôže pri vzdelávaní. Stáva sa neodmysliteľnou pomôckou nielen pri osvojovaní si nového učiva a domácej príprave, ale aj počas previerok a samostatného riešenia úloh na vyučovaní.

Opýtali sme sa žiakov, ktorí používajú editor LAMBDA už druhý školský rok na ich osobné skúsenosti a názory.

Paťo (žiak 9. ročníka, slabozraký):

*„Spočiatku som nestíhal s ostatnými spolužiakmi, ale keď som sa naučil niekoľko základných skratiek, pracoval som rovnako rýchlo ako ostatní spolužiaci. Kedysi som pracoval vo Word, ale v editore LAMBDA sa oveľa ľahšie nájdu rôzne špeciálne symboly, práca so zlomkami je oveľa jednoduchšia, nemusím sa prepínať medzi viacerými programami.*

*S programom pracujem iba na hodinách matematiky, vhodné by bolo použitie aj na chémii hlavne pri riešení rovníc, super je aj že s týmto programom môžeme pracovať na strednej i vysokej škole. Vcelku mi urýchljuje prácu. Výhodou je, že keď sa mi náhodou vybiže notebook na hodine, nestratím informácie, pretože program automaticky všetko uloží.“*

Marek (žiak 9. ročníka, nevidiaci):

*„Ja som od prvého ročníka pracoval iba v braillovom písme. Nosil som ťažký pichťov stroj a množstvo papierov, ktoré som si musel počas hodiny aj niekoľkokrát vymieňať. Na jeden riadok sa totiž vmestí asi 40 znakov. Práca v PC sa mi zdá jednoduchšia a dokážem pracovať rýchlejšie. Už som si zvykol na elektronický zošit LAMBDA a vyhľadávanie skratiek v ňom. Výhodou je používanie PC spolu s braillovským riadkom, čo umožňuje získavať informácie nielen sluchom, ale aj hmatom zároveň. Tento spôsob práce mi vyhovuje najviac. Ľahšie si zapamätám nové informácie a poznatky. Privítal by som možnosť v editore LAMBDA vkladať tabuľky, ako sme zvyknutí v Excelli, a pracovať s nimi.“*

## 5. ZÁVER

Z našej osobnej skúsenosti vyplýva, že rozvoj informatických kompetencií má význam pre nevidiacich žiakov z hľadiska ich prípravy pre ďalšie štúdium. V predmetoch informatická výchova a informatika nadobúdajú žiaci so zrakovým postihnutím dôležité vedomosti a zručnosti. Vďaka tomu sú schopní naučiť sa samostatne ovládať programy potrebné pre ich vzdelávanie. Editor LAMBDA sa ukázal ako vhodná pomôcka na vyučovaní matematiky a fyziky. Používanie matematického editora, ktorý spolupracuje s čítačom obrazovky, znamená pre nevidiacich žiakov možnosť efektívnej integrácie na základnej aj strednej škole.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] Zákon č. 245/2008 Z. z. o výchove a vzdelávaní (školský zákon z 22. mája 2008) [online] [cit. 2015-01-30]. Dostupné na: <[http://www.uips.sk/sub/uips.sk/images/OddMladezASport/Mladez/legislativa/245\\_2008.pdf](http://www.uips.sk/sub/uips.sk/images/OddMladezASport/Mladez/legislativa/245_2008.pdf)>
- [2] Štátny vzdelávací program [online] [cit. 2015-01-30] Dostupné na: <<http://www.statpedu.sk/sk/Statny-vzdelavaci-program.alej>>
- [3] Štátny vzdelávací program. Informatika [online]. [cit. 2015-01-30] Dostupné na: <[http://www.statpedu.sk/files/documents/svp/2stzs/iscsed2/vzdelavacie\\_oblasti/informatika\\_iscsed2.pdf](http://www.statpedu.sk/files/documents/svp/2stzs/iscsed2/vzdelavacie_oblasti/informatika_iscsed2.pdf)>
- [4] Vzdelávací program pre deti a žiakov so zrakovým postihnutím [online]. [cit. 2015-01-30] Dostupné na: <[http://www.statpedu.sk/files/documents/svp/vp\\_pre\\_zz/vp\\_zrakovo/vp\\_zp\\_%20iscsed\\_1\\_2\\_3\\_5.pdf](http://www.statpedu.sk/files/documents/svp/vp_pre_zz/vp_zrakovo/vp_zp_%20iscsed_1_2_3_5.pdf)>
- [5] Veia progetti for accessibility. Lambda [online]. [cit. 2015-01-30] Dostupné na: <[http://www.veia.it/en/lambda\\_product](http://www.veia.it/en/lambda_product)>
- [6] Lambda Project [online]. [cit. 2015-01-30] Dostupné na: <<http://lambdaproject.org/>>
- [7] HORŇANSKÝ, Martin. Sprístupňovanie matematiky nevidiacim na Slovensku prostredníctvom editora Lambda, Študentská vedecká konferencia. ŠVK, FMFI UK Bratislava 2009, p.42-60
- [8] HORŇANSKÝ, Martin., LECKÝ, Peter., KOHANOVÁ, Iveta. Editor Lambda – Nástroj na sprístupnenie matematiky nevidiacim a jeho rozširovanie na Slovensku, Konferencia DidInfo, Banská Bystrica, 2009
- [9] Štátny vzdelávací program Matematika. [online]. [cit. 2015-01-30] Dostupné na: <[http://www.statpedu.sk/files/documents/svp/2stzs/iscsed2/vzdelavacie\\_oblasti/matematika\\_iscsed2.pdf](http://www.statpedu.sk/files/documents/svp/2stzs/iscsed2/vzdelavacie_oblasti/matematika_iscsed2.pdf)>
- [10] Štátny vzdelávací program Fyzika. [online]. [cit. 2015-01-30] Dostupné na: <[http://www.statpedu.sk/files/documents/svp/2stzs/iscsed2/vzdelavacie\\_oblasti/fyzika\\_iscsed2.pdf](http://www.statpedu.sk/files/documents/svp/2stzs/iscsed2/vzdelavacie_oblasti/fyzika_iscsed2.pdf)>
- [11] JAŠKOVÁ, Ľ., STANKOVIČOVÁ, M. Rozvoj informatických kompetencií u nevidiacich žiakov druhého stupňa ZŠ a ich využitie v rôznych predmetoch a osobnom živote. In: Lopúchová, J. Výdobytky techniky pomáhajú pri modernizácii vzdelávania zrakovo postihnutých – Odborná tyflopédická konferencia, IRIS, Bratislava, 2014, ISBN 978-80-89726-26-4
- [12] ŽABKA, J. - ČERNEK, P. : Matematika pre 8. ročník ZŠ a 3. ročník gymnázií s osemročným štúdiom, 2. časť. Bratislava: Orbis Pictus Istropolitana, 2012. 129 s. ISBN 978-80-8120-125-7
- [13] ŠEDIVÝ, O.- ČERETKOVÁ, S. –MALPEROVÁ, M. – BÁLINT, Ľ.: Matematika pre 8. ročník základných škôl, 2. časť. Bratislava: SPN, 2001. 8s. ISBN 80-08-03032-1
- [14] ŠMATINOVÁ, Anna. Zbierka úloh z fyziky pre 6. – 8. ročník ZŠ. Bratislava: SPN, 1995. 84 s. ISBN80-08-00026-0

# Sémantický diferenciál v pedagogickom výskume

Krisztina Czakoová  
Univerzita J. Selyeho  
Bratislavská cesta 3322  
94501 Komárno  
Slovenská republika  
czakok@selyeuni.sk

Veronika Stoffová  
Trnavská Univerzita  
Hornopotočná 23  
918 43 Trnava  
Slovenská republika  
nika.stoffova@seznam.cz

## ABSTRAKT

Predmetom pedagogického výskumu na rôznych úrovniach vzdelávacieho systému a na rôznych typoch škôl je často postoj žiakov ku škole, školskému prostrediu, k spolužiakom, rodičom, učiteľom, počítačom, digitálnym technológiám a pod. Cieľom článku je formulovať určité metodické odporúčania pre meranie postojov žiakov vrátane ich hodnotovej orientácie, pre učiteľov základných a stredných škôl, ale aj pre diplomantov a doktorandov, ktorí sa zaoberajú kvalitatívnym pedagogickým výskumom, aby mohli výsledky svojej výskumnej práce a pedagogickej praxe bez väčších problémov uplatňovať v pedagogickej diagnostike žiaka a triedy. Ako hlavná výskumná metóda je opísaný sémantický diferenciál.

## ABSTRACT

Educational research at different levels of the education system and the different types of schools is often focused on pupil attitudes to school, to the school environment, to classmates, parents, teachers, computers, digital technology and so on. The aim of the article is to formulate certain methodological recommendations for primary and secondary schools teachers for measuring the attitudes of students, including their value orientation. The recommendations are useful also for diploma students and PhD candidates engaged in qualitative educational research that could results from their research and teaching practice without any problems applied in educational diagnosis of a pupil and class. As the main research method is described semantic differential.

## Kľúčové slová

Sémantický diferenciál, pedagogický výskum, kvalitatívny výskum, pedagogická diagnostika žiaka

## Keywords

Semantic differential, educational research, qualitative research, educational diagnostics of pupil

## 1. ÚVOD

Výchovno-vzdelávacia činnosť môže byť len ťažko kvalitne a efektívne vykonávaná bez predchádzajúcej diagnostiky žiaka a triedy, do ktorej patrí, kolektívu, ktorého je súčasťou. Správna diagnostika prináša nielen dôležité informácie o aktuálnom stave, ale umožňuje jeho monitorovanie a sledovanie jeho časového vývoja. Tieto informácie sú veľmi dôležité pre správne riadenie procesu vzdelávania a umožňujú tiež vhodne vplyvať na

jednotlivé javy, ktoré zohrávajú dôležitú úlohu vo vytváraní príjemnej tvorivej pracovnej atmosféry v triede.

V oblasti kvalitatívneho výskumu zohráva dôležitú úlohu sémantický diferenciál. Táto metóda, ktorá je na Slovensku zatiaľ málo používaná, umožňuje skúmať a diagnostikovať vzťahy a postoje jednotlivcov – edukantov ku škole, k školskému a sociálnemu prostrediu, rodičom, učiteľom, vychovávateľom a pod. Úlohou článku je opísať metodický postup ako túto metódu aplikovať v pedagogickom výskume a ukázať možné štatistické spracovania jeho výsledkov (Chráska 2003, Chrásková 2005, 2014a, 2014b)

Bežne používané didaktické testy sú skôr orientované na meranie výkonu žiakov, a odrážajú úspešnosť vzdelávania, hlavne vyučovacieho procesu. Hoci takto získané informácie sú nesporne dôležité pre ďalšie riadenie vyučovania, nesmieme podceňovať ani diagnostiku vzťahov, motívov a postojov žiakov spojených s výchovno-vzdelávacím prostredím. Znalosť týchto zložiek osobnosti pomôže učiteľom v práci so žiakmi a poskytne im informácie, ktoré bežne v školách absentujú (Chrásková 2011, Gavora 2000, Miovský 2006, Kerlinger 1972).

Určenie kvantitatívnych ukazovateľov procesu diagnostikovania postojov je bez používania výpočtovej techniky a moderných technológií v súčasnej informačnej spoločnosti nereálne a ťažko realizovateľné. Preto na spracovanie výsledkov pedagogického výskumu odporúčame použiť adekvátne štatistické metódy, ktoré sa dajú ľahko realizovať pomocou prostriedkov IKT. Počítačové spracovanie výsledkov výskumu urýchli proces hodnotenia a zvýši kvalitu výstupov. Ich grafické zobrazenie prispeje k zvýšeniu názornosti a tiež k lepšej interpretácii výsledkov.

## 2. ROZDIEL MEDZI NÁZOROM A POSTOJOM

Mnohí autori upozorňujú na rozdiel medzi názorom a postojom. Kým názor je neutrálny výrok, o ktorom sa domnievame, že je pravdivý. Proti tomu postoj je vždy výrazne emocionálne zafarbený, súvisí s hodnotovým systémom jednotlivca. Možno dokonca povedať, že postoje priamo vychádzajú z hodnotového rebríčka. Môžeme však tiež nájsť jednotlivcov, ktorí zastávajú postoj diametrálne odlišný od hodnôt. Mnohí odborníci sa domnievajú, že pri vyjadrovaní postojov hrá úlohu sociálna stratégia, teda medzi vysloveným názorom a skutočným postojom je síce súvislosť, ale nie sú totožné. Postoje sú hlavným nástrojom kontaktu s realitou (Chráska 2011, Průcha a kol. 2001).

Postoje sa väčšinou opisujú pomocou troch dimenzií:

1. **Kognitívna dimenzia** – týka sa názorov, myšlienok, ktoré má jednotliviec o predmete postoja.
2. **Emocionálna dimenzia** – týka sa toho, čo osoba cíti k predmetu postoja, emócií a emocionálnych reakcií.
3. **Konatívna či behaviorálna dimenzia** – týka sa sklonov k správaniu vo vzťahu k predmetu postoja.

Na základe predchádzajúceho možno tvrdiť, že medzi názorom a postojom je významový rozdiel a tieto dva pojmy nemôžeme stotožniť a ich pomenovania považovať za synonymá.

### 3. SÉMANTICKÝ DIFERENCIÁL

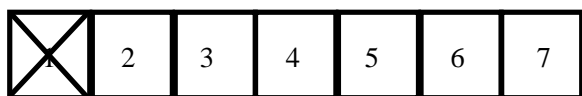
Sémantický diferenciál je výskumná metóda vyvinutá na meranie individuálnych, psychologických významov slov alebo postojov k niečomu. Zameriava sa na jednoduché hodnotiace názory a je teda vhodná predovšetkým na meranie emocionálnych a behaviorálnych aspektov postoja. Jeho veľkou prednosťou je ľahká administrácia a relatívne rýchle vyhodnotenie (Chrásková 2011, Chráska 1995, Gavora 2000).

Sémantický diferenciál bol vypracovaný americkým psychológom Ch. Osgoodom pôvodne mal slúžiť na meranie konotatívneho významu pojmov. Pri použití tejto metódy, každý pojem možno vyjadriť ako bod v tzv. sémantickom priestore (pozri Obr. 3). Pravdou je, že ak posudzuje jeden objekt alebo pojem viac jednotlivcov, každý z nich ho vníma trochu inak, ba niekedy dokonca veľmi odlišne. Vedľa spoločného kultúrneho významu (denotácia) má totiž každý pojem ešte ďalšie, vedľajšie významy (konotácie), ktoré jednotliví posudzovatelia charakterizujú. Základné dimenzie sémantického priestoru boli stanovené pomocou faktorovej analýzy a pomocou nej boli určené tri najvýznamnejšie faktory. Každý pojem sa posudzuje potom z hľadiska týchto troch faktorov:

faktor hodnotenia (h);  
faktor potencie / sily (p);  
faktor aktivity (a).

Vlastný sémantický diferenciál sa skladá z určitého počtu škál, ktoré bývajú väčšinou sedembodové. Každá škála je významne sýtená len jedným faktorom. Krajné body škály vždy označujú jeden pól daného faktora (napr. pre hodnotenie príjemný – nepríjemný, pre potenciu jednoduchý – zložitý a pre aktivitu rýchly – pomalý) a respondenti majú na škále vyznačiť, ako ho vnímajú oni. Napr. ak respondent vníma budúcnosť ako to najhoršie, neperspektívne, tak označí na škále 1 (Pozri Obr. 1).

Budúcnosť

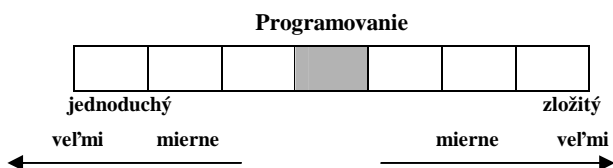


bez perspektívy

skvelá

Obr. 1 Príklad škály pre sémantický diferenciál

Na mapovanie významu pojmov metódou sémantického diferenciálu autori použili nasledujúcu grafickú úpravu formulára so škálovaním.



Obr. 2 Iný príklad škály pre sémantický diferenciál

Sémantický diferenciál meria poznávaciu a emocionálnu zložku postojov, kde ide hlavne o faktor hodnotenia. Reliabilita sémantického diferenciálu ako aj jeho validita sú veľmi vysoké.

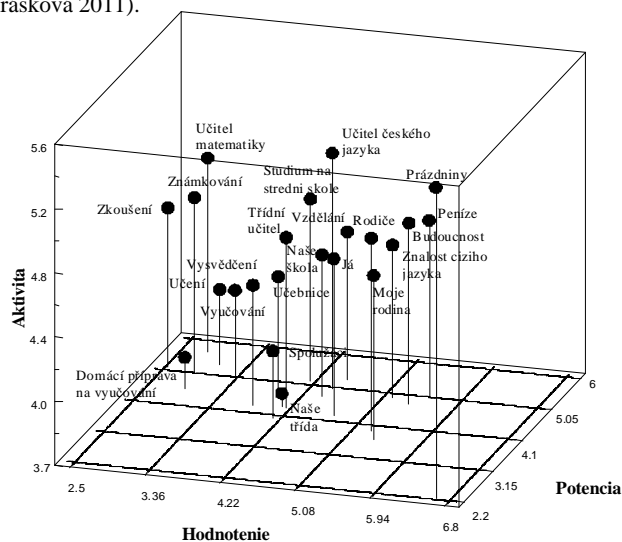
Na presnejší opis sémantického priestoru je niekedy výhodné určiť, ako ďaleko sú jednotlivé pojmy v sémantickom priestore od seba. Ako miera vzdialenosti sa používa tzv. lineárna vzdialenosť (dištancia)  $D$ , ktorá sa určí podľa nasledujúceho vzťahu:

$$D_{ij} = \sqrt{(x_i - x_j)^2} \quad (1)$$

Kde  $D_{ij}$  je lineárna vzdialenosť – dištancia medzi pojmom  $i$  a pojmom  $j$ ,  $x_i$  je škálovaná hodnota pojmu  $i$  a  $x_j$  je škálovaná hodnota pojmu  $j$ .

Pri výpočte hodnoty  $D_{ij}$  medzi dvoma pojmami môžeme vychádzať buď z jedného faktora (najčastejšie faktora hodnotenia) alebo z faktorov dvoch či troch. Vypočítané hodnoty lineárnej vzdialenosti sa zapisujú do tzv. D-matic. Pri analýze D-matic sa všímajú najmä malé a veľké hodnoty lineárnej vzdialenosti, ktoré vypovedajú o (veľkej alebo malej) podobnosti vnímania daných pojmov.

Aby meranie bolo validné, mala by byť každá škála sémantického diferenciálu významne sýtená len jedným faktorom. Pôvodných škál profesora Osgooda bolo celkom 50 (Osgood a kol. 1957, Chrásková 2011).



Obr. 3 Sémantický priestor pojmov školského a sociálneho prostredia u žiakov strednej školy (Chrásková 2011)

Osgood po vykonanej faktorovej analýze meraní na škálach uvedených v tabuľke 1 extrahoval 4 spoločné faktory, z ktorých posledný, štvrtý faktor bol už prakticky bezvýznamný. Najsilnejší bol faktor prvý – označený ako faktor hodnotenia (h), druhý ako faktor potencie (p) a tretí ako faktor aktivity (a). (Pozri tabuľku 1).

Pri navrhovaní škál pre sémantický diferenciál možno vychádzať z pôvodných Osgoodových škál, je však potrebné dodržať určitý pomer, teda treba pamätať na to, že škály hodnotiace sú obvykle najsilnejšie. Teda nemali by početne prevažovať nad škálami meracími potenciu či aktivitu. Obvyčajne sa používajú sedemstupňové škály (Osgood a kol. 1957). Aby sa znížilo nebezpečenstvo stereotypného posudzovania na škálach, je vhodné niektoré škály zaradiť ako reverzné. V ďalej uvedenom záznamovom liste na skúmanie vzťahu respondentov k "rodičom" sú tieto reverzné škály označené \*. (Pozri Obr. 4).



**Tabuľka 1: Pôvodné škály pre sémantický diferenciál podľa Ch. Osgooda spresnené jednou z autoriek**

1. dobrý - zlý	h	26. vlhký - suchý	IV.
2. veľký - malý	p	27. úctivý - neúctivý	h
3. krásny - škaredý	h	28. uvoľnený - napätý	h
4. jasný - nejasný	h	29. odvážny - zbabelý	h
5. tvrdý - mäkký	p	30. dlhý - krátky	p
6. sladký - horký	h	31. bohatý - chudobný	h
7. silný - slabý	p	32. jasný - zahmlený	h
8. čistý - špinavý	h	33. horúci - studený	a
9. vysoký - nízky	h	34. tučný - chudý	p
10. pokojný - vzrušený	h	35. pekný - škaredý	h
11. chutný - odporný	h	36. svetlý - tmavý	h
12. hodnotný - bezcenný	h	37. tlmný - prenikavý	p
13. biely - čierny	a	38. hranatý - guľatý	a
14. mladý - starý	a	39. voňavý - páchnuci	h
15. láskavý - krutý	h	40. štedrý - skúpy	h
16. hlučný - tichý	p	41. aktívny - pasívny	a
17. hlboký - plytký	p	42. drsný - hladký	h
18. príjemný - nepríjemný	h	43. zdravý - chorý	h
19. čierny - biely	h	44. rýchly - pomalý	a
20. chutný - nechutný	h	45. čestný - nečestný	h
21. veselý - smutný	h	46. hrboľatý - hladký	p
22. ostrý - tupý	a	47. blízky - vzdialený	h
23. prázdny - plný	h	48. dráždivý - nedráždivý	h
24. divoký - krotký	h	49. zdravý - chorý	h
25. ťažký - ľahký	p	50. široký - úzky	h

Keď však bola vykonaná kontrolná faktorová analýza v českých sociálnokultúrnych podmienkach (ktoré možno považovať za rovnaké so slovenskými) bolo zistené, že na rozptyle hodnôt sa významne podieľajú iba dva faktory (Chráska 2007). Autor, profesor Chráska sa presvedčil o tom, že extrakcia troch faktorov vedie k pomerne nespoľahlivému meraniu, lebo jedna škála meria v prípade rôznych pojmov rôzne faktory. Z toho dôvodu metódu sémantického diferenciálu modifikoval tak, aby sa dal jednoducho aplikovať v bežných podmienkach. Prvý faktor, označený ako faktor hodnotenia, zvolil v zhode s výsledkami Osgooda. Druhý faktor definoval ako kombináciu pôvodných faktorov potencie a aktivity a nazval ho faktorom energie. Škály, ktoré sú sytené faktorom energie, vyjadrujú ako ďaleko sú vybrané pojmy pociťované respondentmi ako "čosi", čo je spojené s námahou, ťažkosťami, premenami či aktivitou. Na základe vykonaných analýz bol následne vytvorený merací nástroj ATER (Attitudes Toward Education Reality), obsahujúci 10 škál, z ktorých 5 meria faktor hodnotenia (h) a 5 faktor energie (e). Na obrázku 4 „\*“ opäť označuje reverzné škály.

#### 4. METÓDY NA ŠTATISTICKÚ ANALÝZU EMPIRICKÝCH ÚDAJOV A NÁSTROJE IKT NA ICH SPRACOVANIE

Ako hlavnú výskumnú metódu na zistenie názorov a postojov žiakov a študentov odporúčame použiť sémantický diferenciál. Na jednoduché spracovanie získaných údajov pomocou záznamových listov dvojfaktorového sémantického diferenciálu odporúčame tabuľkový procesor EXCEL, ktorý umožňuje aj grafické znázornenie výsledkov. Grafické zobrazenie výsledkov podporuje ich ľahšiu a presvedčivú interpretáciu. Na štatistickú analýzu navrhujeme použiť nasledujúce metódy: Študentov t-test, ANOVA, zhľukovú analýzu (cluster analysis), výpočet

Spearmanovho koeficientu poradovej korelácie a pod. Štatistická analýza sa jednoducho realizuje v prostredí programových balíkov na to určených, ako je napr. STATISTICA 9.Cz (Chráska 2007, Chrásková 2011, Meloun a kol. 2005).

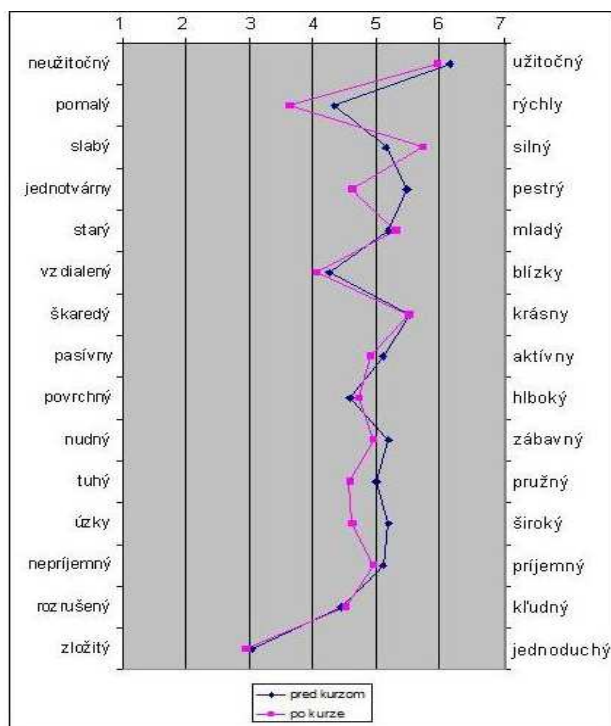
RODIČIA			
1	dobrý		zlý h
2	nenáročný		náročný e*
3	nepríjemný		príjemný h*
4	problémový		bezprobl. e
5	svetlý		tmavý h
6	ľahký		ťažký e*
7	oškľivý		krásni h
8	nenáročný		náročný e*
9	sladký		kyslí h
10	prísni		mierni e

**Obr. 4 Záznamový list dvojfaktorového sémantického diferenciálu ATER pre pojem RODIČIA“ (Chrásková 2011)**

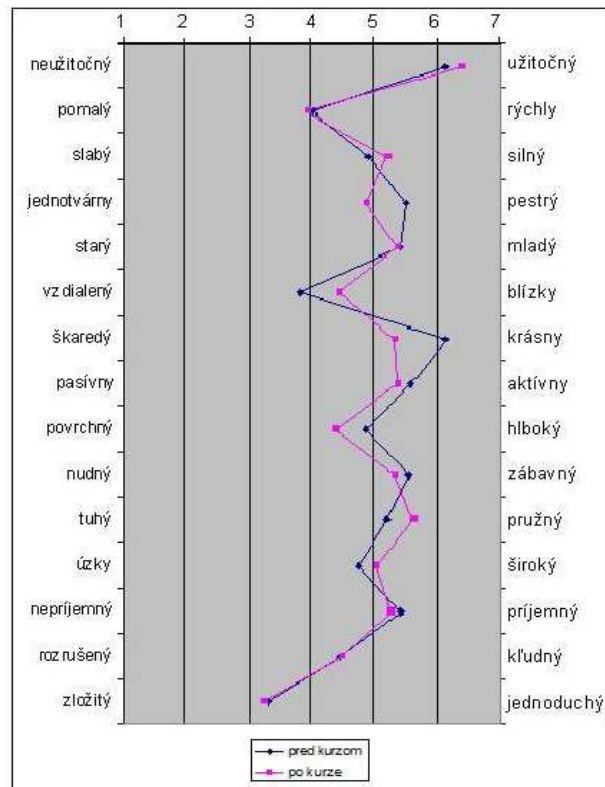
PROGRAMOVANIE									
užitočný									neužitočný
pomalý									rýchly
silný									slabý
jednotvárný									pestrý
mladý									starý
vzdialený									blízky
krásny									škaredý
pasívny									aktívny
hlboký									povrchný
nudný									zábavný
pružný									tuhý
úzky									široký
príjemný									nepríjemný
rozrušený									pokojný
jednoduchý									zložitý
<div> <div>veľmi</div> <div>mierne</div> <div>mierne</div> <div>veľmi</div> </div>									

**Obr. 5 Autentický záznamový list sémantického diferenciálu na skúmanie vzťahu respondentov k programovaniu**

Sémantický diferenciál sme použili na skúmanie vzťahov budúcich učiteľov informatiky a tiež učiteľov primárneho vzdelávania, ktorí absolvovali jednosemestrový predmet orientovaný na tvorbu didaktických aplikácií v prostredí Imagine. Ich postoje sme merali pred a po absolvovaní kurzu. Grafické hodnotenie vzťahu a postoja k programovaniu a k tvorbe vlastných aplikácií v tomto prostredí uvádzame v grafickom vyjadrení na obr. 6 a 7. Na zber údajov sme použili záznamový list dvojfaktorového sémantického diferenciálu na obr. 5.



Obr. 6 Grafické hodnotenie pojmu PROGRAMOVANIE



Obr. 7 Hodnotenie TVORBY VLASTNEJ APLIKÁCIE

## 5. ZÁVER

Metódou sémantického diferenciálu možno pomerne jednoducho a dostatočne spoľahlivo merať individuálne, psychologické chápanie pojmov u respondentov. Pri tom sa vychádza z predpokladu, že vzťahy možno spoľahlivo zachytiť/vyjadriť pomocou vhodne zvolených pojmov. Pomocou štatistických analýz bolo overené, že zjednodušená metóda dvojfaktorového sémantického diferenciálu je dostatočne spoľahlivá a presná na meranie postojov a vzťahov.

Pre jednoduchú aplikáciu sémantického diferenciálu a jeho vyhodnotenie odporúčame použiť navrhnuté škály (príp. podľa opísaných pravidiel upravené a novonavrhnuté škály). Predpokladáme, že dvojfaktorový sémantický diferenciál dostatočne spoľahlivo meria oba faktory, tak hodnotenie, ako aj energiu skúmaných pojmov, preto ho odporúčame využívať v pedagogickom výskume tak pre učiteľov, ako aj pre doktorandov.

*Príspevok vznikol v rámci projektu KEGA 010UJS-4/2014 „Modelovanie, simulácia a animácia vo vzdelávaní“.*

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] GAVORA, P. *Úvod do pedagogického výskumu*. Brno : Paido, 2000. ISBN 80-85931-79-6
- [2] GAVORA, P. a kol. 2010. Elektronická učebnica pedagogického výskumu. [online]. Bratislava : Univerzita Komenského, 2010. Dostupné na: <http://www.emetodologia.fedu.uniba.sk/> ISBN 978-80-223-2951-4.
- [3] CHRÁSKA, M. Postoje k učiteľské profesi v pregraduální učitelské přípravě. In. Sborník příspěvků z XI. Konference ČAPV. Brno : Paido, 2003. ISBN 80-7315-046-8.
- [4] CHRÁSKA, M. *Základy výskumu v pedagogice*. Olomouc : PdF UP, 2000. ISBN 80-7076-798-9.
- [5] CHRÁSKA, M. Změny v sémantickém prostoru studentů pedagogické fakulty. *Pedagogika*. 1995, roč. 45, č. 1, s. 71-76. ISSN 3330-3815.
- [6] CHRÁSKA, M.: *Hypotézy a jejich ověřování v klasických pedagogických výzkumech*. 1. vyd. Olomouc: Votobia, 2005. 61 s. ISBN 80-7220-253-7.
- [7] CHRÁSKA, M. *Metody pedagogického výskumu. Základy kvantitativního výskumu*. 1. vyd. Praha : Grada, 2007. 272 s. ISBN 978-80-247-1369-4.
- [8] CHRÁSKOVÁ, M.: Využívání nástrojů IKT při statistickém zpracování výstupů z pedagogického výskumu. (Rigorózná práca). Komárno : UJS v Komárne, 2011, 120 s.
- [9] CHRÁSKOVÁ, M.: Formative effect of study at faculty of education on students' attitudes to widereducational reality. In: *Psychology and psychiatry, sociology and healthcare, education – conference preceedings. Volume 1*, STEF92 Technology Ltd. : Albena, s. 175-186, 2014 ISBN 978-619-7105-22-3, ISSN 2367-5659
- [10] CHRÁSKOVÁ, M.: Role of health literacy in an information society. In: *EDICATION – TECHNOLOGY- COMPUTER SCIENCE: Main problem of technology and Professional education – Scientific annual No/5/2014/PART 1*. Katedra

DIDINFO 2015, Univerzita Mateja Bela, Banská Bystrica, 2015.

Nowoczesnych Technologii Edukacyjnych Uniwersytetu  
Rzeszowskiego : Rzeszow, 2014, s. 401-406

- [11] KERLINGER, F. N. *Základy výzkumu chování*. Praha : Academia, 1972.
- [12] MIOVSKÝ, M. *Kvalitativní přístup a metody v psychologickém výzkumu*. 1. vyd. Praha : Grada, 2006. 332 s. ISBN 80-247-1362-4
- [13] OSGOOD, Ch. E. – SUCI, G. J. – TANNENBAUM, P. H. *The measurement of meaning*. Urbana : University of Illinois Press, 1957. ISBN 0-252-74539-6.
- [14] PRŮCHA, J. – WALTEROVÁ, E. – MAREŠ, J. *Pedagogický slovník*. 3. vyd. Praha : PORTÁL, 2001. 322 s. ISBN 80-7178-579-2.
- [15] MELOUN, M. – MILITKÝ, J. – HILL, M. *Počítačová analýza vícerozměrných dat v příkladech*. Praha : Academia, 2005. 450 s. ISBN 80-200-1335-0

# Programovací jazyk Python vo výučbe predmetu Algoritmy a dátové štruktúry

Andrej Blaho

Katedra základov a vyučovania informatiky, FMFI UK  
v Bratislave  
Mlynská dolina  
848 48 Bratislava  
Slovensko  
blaho@fmph.uniba.sk

Peter Tomcsányi

Katedra základov a vyučovania informatiky, FMFI UK  
v Bratislave  
Mlynská dolina  
848 48 Bratislava  
Slovensko  
tomcsanyi@fmph.uniba.sk

## ABSTRAKT

Na FMFI UK už dva roky učíme predmet Algoritmy a dátové štruktúry pre študentov druhého ročníka odboru Aplikovaná informatika s použitím programovacieho jazyka Python. Voľba programovacieho jazyka sa na jednej strane zdá byť prirodzená lebo je to v súčasnosti populárny programovací jazyk a študenti v ňom absolvovali základný kurz programovania v prvom ročníku štúdií. Na druhej strane tu však sú aj otázky nakoľko je interpretovaný jazyk vhodný na takéto použitie, či v ňom môžu študenti získať dobrú predstavu o časovej zložitosti algoritmov, teda či niektoré vlastnosti jazyka nesťažujú názorné vysvetlenie niektorých pojmov z teórie zložitosti algoritmov.

Pred začiatkom výučby sme mali predstavu o odpovediach na zmienené otázky. Praktická výučba predmetu nám okrem iného poskytla aj ďalšie argumenty v prospech použitia Pythonu, ďalšie príklady kedy je potrebné v detailoch upraviť všeobecné prístupy k výučbe zložitosti algoritmov s prihliadnutím na konkrétne vlastnosti Pythonu ako aj príklady keď Python pomáha pri jednoduchšej a názornej implementácii abstraktných dátových typov.

V článku popisujeme náš prístup k výučbe predmetu a na konkrétnych príkladoch ukážeme ako Python pomáha predmetu ADŠ aj ako si vyučujúci musí dať pozor na niektoré jeho špecifické vlastnosti.

Článok považujeme za príspevok do diskusie o spôsoboch výučby zložitosti algoritmov vo vysokoškolskej príprave odborníkov - programátorov. Môže poslúžiť ako motivácia pri zmenách študijných programov na iných fakultách na Slovensku aj v zahraničí.

## ABSTRACT

We use the programming language Python for teaching the subject of Algorithms and Data Structures to the students of Applied Informatics at our faculty. We do it already for two years. The choice of the programming language seems to be natural because it is a popular programming language at present and also because the students use it in their introductory programming course. On the other hand some people raise questions whether an interpreted language is appropriate for this kind of use, whether the students can gain a good insight into the complexity of algorithms, i.e. whether some of its features do not hinder explanation of some important concepts of the algorithmic complexity theory.

Before starting teaching we already had some ideas about the answers to such questions. The actual teaching of the subject gave us some more arguments in favor of using Python, some more

examples of when some details of the general approach to teaching algorithmic complexity must be specifically adjusted because of the Python language as well as some examples when Python specifically helps us to implement abstract data types in easy and straightforward manner.

In the paper we describe our approach to teaching the subject. We show actual examples how Python helps to teach the subject and which specific features of the language the teacher must be aware of.

We consider our paper being a contribution to the discussion on the ways of teaching the subject of Algorithms and Data Structures for undergraduate students that will become programmers. It may serve as motivation for changing the study programs at other faculties both in Slovakia and abroad.

## Kľúčové slová

Python, Algoritmy a dátové štruktúry, Abstraktné dátové typy, Zložitost' algoritmov

## Keywords

Python, Algorithms and Data Structures, Abstract Data Types, Complexity of Algorithms

## 1. ÚVOD

V študijnom programe Aplikovaná informatika na Fakulte matematiky fyziky a informatiky Univerzity Komenského v Bratislave je predmet Algoritmy a dátové štruktúry (ADŠ) zaradený v treťom semestri ako priame pokračovanie úvodného dvojsemestrového kurzu programovania.

V akademickom roku 2013/2014 sa úvodný kurz programovania zmenil - namiesto jazyka Pascal sa začal používať jazyk Python. Preto sme sa rozhodli zmeniť aj predmet ADŠ tak, aby bol postavený na jazyku Python. So zmenou sme však nečakali kým príde do druhého ročníka takí prváci, ktorý už prešli novým kurzom programovania, ale predmet ADŠ sme zmenili v tom istom akademickom roku. Chceli sme tak získať skúsenosti s výučbou predmetu čo najskeer aj za cenu, že prvý rok jeho výučby bude iný ako tie ďalšie.

Cieľom predmetu ADŠ je:

- pochopiť zložitost' jednoduchých algoritmov, vedieť ju zistiť, porovnať, resp. preveriť z nameraných hodnôt behu programu pre rôzne vstupy,
- spoznať základné dátové typy (napr. polia, spájané zoznamy, stromy, prioritné fronty, asociatívne polia, grafy a pod.) z pohľadu abstraktných dátových typov (ADT),

- zoznámiť sa s rôznymi realizáciami týchto ADT, so zložitou týchto realizácií, využitím týchto realizácií pri riešení rôznych problémov, napr. pri triedení.

Keďže predmet je nasadený v 3. semestri štúdia a študenti mali zatiaľ veľmi malé skúsenosti s efektívnosťou algoritmov, je veľmi dôležité, aby sa študenti zoznámili so všetkými dátovými konštrukciami a algoritmami prakticky pri programovaní, čo znamená, že aby buď tieto štruktúry programovali od nuly, alebo aby dostali niektoré predpripravené konštrukcie (napr. базové triedy) a z nich odvádzali rôzne realizácie. Veľmi dôležitou súčasťou je aj kontrola efektívnosti realizácie, t.j. meranie časov pre rôzne veľké vstupy a tiež praktické porovnávanie rôznych realizácií. Takto mohli už od najjednoduchších tém vidieť, ako sa správajú algoritmy so zložitostami napr.  $O(n)$ ,  $O(n^2)$ ,  $O(n \cdot \log n)$ .

Pri príprave predmetu sme čerpali hlavne z učebnice [1], ale využili sme aj ďalšie zdroje hlavne z analogických kurzov na iných univerzitách vo svete.

Počas výučby sme používali štandardnú verziu Pythonu 3.3 resp. 3.4 známu pod názvom CPython z [2] a jej prostredie IDLE.

Stránka nášho predmetu je dostupná na internete [3]. Odporúčame im aj interaktívny kurz Pythonu [4].

## 2. PRIEBEH VÝUČBY

V čase písania príspevku máme za sebou dva roky výučby ADŠ v Pythone podľa nového plánu.

### 2.1 Pilotný akademický rok 2013/14

Prvá verzia predmetu bola dosť náročná nielen pre študentov ale aj pre nás vyučujúcich. Študentom sme chceli okrem samotného obsahu ADŠ predstaviť aj jazyk Python a to na takej úrovni, aby sa dali ukázať konštrukcie základných dátových štruktúr aj s príslušnými algoritmami.

Postrehy z prvého roku:

- prvé tri týždne semestra (prednášky, cvičenia, projekty) boli venované čisto jazyku Python - študenti dovtedy poznali jazyk Pascal, vďaka čomu nemali problém so smerníkmi, dynamickými premennými a štruktúrami, objektovým programovaním,
- väčšina študentov tento nový jazyk prijala veľmi pozitívne, hoci pre niektorých študentov bol takýto rýchlokurz na ich hranici zvládania - najväčšie problémy boli s tými vlastnosťami jazyka Python, ktoré ho výrazne odlišujú od Pascalu aj od iných programovacích jazykov, s ktorými sa študenti stretli:
  - chýbajúce deklarácie premenných - ak nie je typ premennej zdokumentovaný v komentári, tak ho je možné zistiť len čítaním kódu funkcie,
  - prístup k lokálnym a globálnym premenným sa výrazne líši od Pascalu,
  - práca s atribútmi triedy vyžaduje explicitné používanie prefixu *self*,
  - práca s asociatívnym poľom (štandardný typ dictionary v Pythone),
  - špeciálne (magické) metódy tried, ktoré zabezpečujú napr. preťažovanie operátorov,
  - iterátory a generátory,

- pomocou Pythonu sa veľmi dobre darilo predviesť zložitost rôznych realizácií algoritmov: dobre sa dá merať čas behu pre rôzne použité konštrukcie a rôzne veľké vstupy.

### 2.2 Akademický rok 2014/15

Na základe skúseností z prvého roku sme pripravili výučbu v akademickom roku 2014/15 keď už takmer všetci študenti druhého ročníka absolvovali v prvom ročníku dvojsemestrálny kurz programovania v Pythone. Vďaka tomu sme nemuseli venovať čas výučbe samotného Pythonu a mohli sme sa celý semester venovať témam samotnej prednášky ADŠ.

Aj pri týchto študentoch sa ukázali niektoré problémy s jazykom Python, najmä:

- pojem abstraktnej triedy,
- špeciálne (magické) metódy tried,
- polia, resp. iné meniteľné typy ako parametre funkcií,
- iterátory, generátory,
- pochopenie odkazov (references) keďže na rozdiel od minuloročných študentov títo sa neučili smerníky v Pascale a jazyk C++ sa učia až paralelne s touto prednáškou.

Pre nás vyučujúcich to znamená, že musíme dať väčší dôraz hlavne na precvičovanie polí ako parametrov a tiež magických metód už v prvom ročníku.

Študenti na odovzdávanie projektov (cvičenia, domáce úlohy, projekty, skúšky) využívajú súborový server L.I.S.T., ktorý je k dispozícii na katedre aplikovanej informatiky. Tento server poskytuje aj testovacie nástroje (napr. UnitTest) pre jazyk Python. Keďže bol tento server v tomto semestri v testovacej prevádzke (dokončuje sa ako diplomová práca nášho študenta), samotné testovanie sa využilo len počas praktickej skúšky. V predchádzajúcom semestri sa intenzívne využíval aj na každotýždenné testovanie domácich úloh.

Praktická skúška pomocou testovacieho nástroja prebiehala nasledovne:

- študent si na začiatku skúšky zo servera stiahol pripravené súbory (napr. s definíciou базových tried),
- počas skúšky, ktorá trvá 2,5 hodiny môže ľubovoľne veľakrát uložiť svoje riešenie na server, kde sa automaticky spustí testovací program - ten spustí sériu 10 testov a študent sa okamžite dozvie, koľko z nich prešlo úspešne a aké chyby sa objavili pri neúspešných testoch,
- takto môže postupne odladiť všetky chyby, až kým nie je program 100% správny,
- do výslednej známky za praktickú skúšku sa počíta to odovzdané riešenie, ktoré bolo najúspešnejšie,
- aj študenti, ktorým automatické testy neprešli, mohli dosiahnuť hodnotenie lebo v takomto prípade vyučujúci hodnotil ich program tradičným spôsobom (čítaním zdrojového kódu programu) a prihliadal na to, aká bola príčina zlyhania v automatickom testovaní.

Pre nás vyučujúcich je takýto spôsob testovania dosť náročný, lebo pre každé zadanie skúšky treba vytvoriť špecifický testovací program v Pythone, ktorý preverí čo najviac požadovaných prípadov riešenia (napr. spúšťa riešenie s rôznymi súbormi vstupných údajov a kontroluje správnosť výsledku, čo môže byť halda, hešovací tabuľka, vyhľadávací strom a pod.). Na druhej strane automatické testovanie odbremení vyučujúceho od práce



pri hodnotení tých študentov, ktorým prešli všetky testy v poriadku a umožňuje mu sa zamerať na študentov, ktorým automatické testy neprešli.

### 3. ŠPECIFIKÁ PYTHONU A ADŠ

Po dvoch rokoch výučby predmetu ADŠ v Pythone vidíme tieto výhody a nevýhody jazyka:

- zápis algoritmov je často veľmi dobre čitateľný aj pre študentov, ktorí nepoznajú Python, ale len pokiaľ sa nepoužijú veľmi špecifické vlastnosti jazyka (napríklad iterátory),
- to, že nie je potrebné deklarovať premenné a ich typy nielen skracuje zápis programu, ale často zjednodušuje funkčnosť algoritmu pre rôzne typy (napr. dobre zapísaný algoritmus triedenia je funkčný pre ľubovoľný typ prvkov triedeného poľa), na druhej strane to ale môže sťažiť štúdium kódu lebo informácia o type premenných je často podstatná pre pochopenie algoritmov,
- pri definovaní dátových typov môže byť zaujímavé to, že trieda môže byť odvodená z viacerých tried (napr. novú triedu vytvoríme z bázeovej abstraktnej triedy a zároveň z nejakej realizácie stromu),
- preťažovanie operácií a niektorých systémových funkcií: pri vytváraní novej triedy môžeme veľmi prirodzene zabudovať do nej špeciálne metódy vďaka ktorým sa budú k nemu správať operátory a niektoré neobjektové funkcie rovnako ako k štandardným typom, vďaka tomuto je testovanie správnosti a rýchlosti takejto realizácie veľmi prirodzené, na druhej strane to vyžaduje špecifické znalosti Pythonu,
- elegantný zápis generátorov, t.j. funkcií, ktoré nevytvoria naraz celý výsledok, napr. postupnosť nejakých hodnôt, ale hodnoty sa vytvárajú postupne tak, ako sa budú vo volanom programe používať,
- celočíselná aritmetika nie je obmedzená nejakým konkrétnym počtom bitov (tak ako vo väčšine iných programovacích jazykov), ale len veľkosťou voľnej pamäte počítača,
- Python je interpreter, a preto sa s ním veľmi príjemne experimentuje, resp. sa ladia jednoduché zápisy programov, na druhej strane však používané prostredie IDLE poskytuje len obmedzené nástroje na ladenie (krokovanie a sledovanie premenných),
- keďže hĺbka rekurzie je nastavená na 1000, študenti sú v niektorých prípadoch nútení vytvárať nerekurzívne verzie algoritmov, čo je z pohľadu predmetu niekedy veľmi dôležitá zručnosť (resp. musia uvažovať o situáciách, kedy obmedzenie rekurzie môže spôsobovať problémy),
- znaky v znakových reťazcoch sú uchovávané vo formáte Unicode, čo je obrovskou výhodou najmä pri spracovávaní slovenských textov (napr. ako testovacie súbory v rôznych úlohách),
- Python je interpreter a teda v niektorých prípadoch sú jeho programy výrazne pomalšie ako napr. zapísané (a preložené) v jazyku C.

V nasledujúcich podkapitolách rozoberieme niektoré výhody a nevýhody do väčších podrobností.

#### 3.1 Vnútorne zobrazenie reťazcov

Jednoznačne kladne hodnotíme prístup Pythonu ku kódovaniu znakov v reťazcoch v pamäti. Od verzie 3.3 sa pre ich vnútorné

zobrazenie jednotlivých používajú výlučne skutočné kódové hodnoty (code points) znakov v Unicode.

Na rozdiel od Javy, C++, Lazarusu či Delphi ale aj starších verzií Pythonu, ktoré používajú na vnútorné zobrazenie reťazcov kódy UTF-16, UTF-8 prípadne aj osembitové kódy, tak pri práci s reťazcami odpadajú problémy s určením dĺžky reťazca (dĺžka je vždy počet znakov), prístupom k jednotlivým znakom reťazca, ako aj s vysvetľovaním ďalších špeciálnych kódov UTF-16 či UTF-8.

Jediné miesto, kde je potrebné prihliadať na kódovanie je vstup a výstup textových súborov, ale aj tu je to veľmi jednoduché (ak nie je súbor uložený vo veľmi neštandardnom kódovaní), stačí vo funkcii `open` uviesť názov kódovania:

```
f = open('súbor.txt', encoding='utf-8')
```

#### 3.2 Metóda `append` zoznamov

Zoznam je jedna zo základných dátových štruktúr Pythonu. Na rozdiel od zoznamu v Lispe a jemu príbuzných jazykoch nie je implementovaný ako spájaný zoznam ale ako dynamické pole referencií (smerníkov).

Metóda `append` pridáva prvok na koniec zoznamu:

```
x = []
n = 10
for i in range(n):
    x.append(i)
print(x)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Programátor, ktorý nepozná Python by sa mohol domnievať, že `append` je implementovaný jednoduchým spôsobom, pri ktorom sa realokuje pamäť pri pridaní každého jedného prvku. To by znamenalo, že operácia `append` na  $n$ -prvkovom zozname má zložitosť  $O(n)$ .

Teda, že vyššie uvedený program na postupné vytvorenie zoznamu by mal mať časovú zložitosť  $O(n^2)$ . Praktický experiment s meraním času by nás však presvedčí, že to nie je pravda:

```
import time
for n in [100000, 1000000, 10000000]:
    x = []
    ti = time.time()
    for i in range(n):
        x.append(i)
    print(time.time() - ti)
100000 0.05600619316101074
1000000 0.5910589694976807
10000000 6.049605131149292
```

Vidíme, že pre desaťkrát viac prvkov je čas približne desaťnásobne väčší, nie stonásobne väčší. Je to preto, lebo `append` je implementovaný tak, že si vytvára "rezervu" pre ďalšie predĺžovanie poľa.

To sa dá dokázať jednoduchým pokusom za pomoci funkcie `getsizeof` z modulu `sys` (výsledky sú pre 32-bitovú verziu Pythonu, 64-bitová dá trochu iné výsledky):

```
import sys
x = []
for i in range(18):
    print(len(x), sys.getsizeof(x))
    x.append(i)
0 36
1 52
2 52
3 52
```

```

4 52
5 68
6 68
7 68
8 68
9 100
10 100
11 100
12 100
13 100
14 100
15 100
16 100
17 136

```

Vidíme, že zoznam sa v pamäti zväčšil keď bol pridaný prvý prvok, piaty prvok, deviaty a sedemnásť prvok.

Tento a ďalšie experimenty teda treba so študentmi spraviť, aby neskôr vedeli správne odhadovať zložitosť algoritmov, ktoré používajú metódu *append*. Experimenty vedú k zadefinovaniu pojmu *amortizovanej zložitosti* a k poznatku, že amortizovaná zložitosť metódy *append* je  $O(1)$ .

Teda existencia zložitého základného typu zoznam v Pythone na jednej strane v sebe skrýva zložitejšie funkcie, ktoré vyžadujú dôkladnejšie vysvetlenie, na druhej strane to dáva možnosť na jednoduchom príklade ukázať dôležitý pojem *amortizovanej zložitosti*.

### 3.3 Obmedzenia kladené na rekurziu

Rekurzívne algoritmy sú dôležitou súčasťou výučby predmetu ADŠ.

Avšak štandardná verzia Pythonu neimplementuje elimináciu chvostovej rekursie, preto v nej nie je možné uprednostňovať rekurziu pred iteráciou tak, ako je to bežné v napríklad Lispe. Navyše aj hĺbka zásobníka volaní je obmedzená na 1000 (limit sa dá sa zväčšiť, ale neodporúča sa to).

Preto v Pythone treba riešiť iteratívne všetko, čo sa dá. V mnohých prípadoch tomu jazyk pomáha jednoduchými a kompaktnými zápismi pre iteratívne vytvorenie zoznamov (list comprehension):

```

x = [i**2 for i in range(10)]
y = [i for i in x if i % 2 == 0]

```

Preto sme pri výučbe venovali väčšiu (než obvyklú) pozornosť jednak nerekurzívnym algoritmom (ak také existujú pre daný problém), jednak metódam simulovania rekursie pomocou zásobníka.

Pri praktických cvičeniach sme však pozorovali, že študenti často dostatočne nechápali prečo komplikujeme niektoré rekurzívne algoritmy ich prepisovaním pomocou zásobníka pričom iné nechávame nezmenené.

Preto by sme mali možno (ešte) viac času venovať vysvetleniu na konkrétnych príkladoch, ktoré algoritmy môžu naraziť na limit hĺbky zásobníka aj pri relatívne malých dátach (napríklad nebalansované binárne vyhľadávacie stromy alebo Quicksort) a ktoré na limit určite nenarazia (napríklad Heapsort, ktorý by teoreticky narazil na limit až pri  $n = 2^{1000} \approx 10^{301}$  pričom horný odhad pre počet atómov v celom pozorovateľnom vesmíre je podľa [1, str. 129] "len"  $10^{100}$ ).

### 3.4 Vstavané funkcie sú vždy rýchlejšie

Pretože Python je interpretovaný jazyk, programy v ňom napísané nikdy nedosiahnu rýchlosť kompilovaného jazyka.

Keďže nám v predmete ADŠ ide o porovnanie rôznych algoritmov a ich asymptotickej zložitosti, mohlo by sa zdať, že absolútna rýchlosť behu programov nehrá v ňom dôležitú úlohu.

Je to pravda pokiaľ porovnávame algoritmus naprogramovaný v Pythone s iným algoritmom taktiež naprogramovaným v Pythone. Avšak napríklad pri testovaní triediacich algoritmov často používame tento návod na testovanie správnosti implementácie konkrétneho algoritmu:

```

from random import *
pole = [randrange(1000) for i in range(n)]
print(moje_triedenie(pole) == pole.sorted())

```

Druhý riadok vygeneruje náhodné pole dĺžky  $n$ , v treťom riadku sa na pole vyvolá testovaný triediaci algoritmus *moje\_triedenie* a jeho výsledok sa porovná s výsledkom štandardnej metódy *sorted*, ktorá triedi zoznam a vypíše sa výsledok porovnania (logická hodnota).

Keď sa v takomto príklade zároveň meria čas, tak študenti zistia, že nimi naprogramované triedenie, hoci teória o ňom tvrdí, že je jedno z najlepších možných, je stále 10 až 20-krát pomalšie ako vstavaná funkcia *sorted*. Takéto zistenie často na prvý pohľad protirečí všetkému, čo sa na predmete ADŠ učia a počas prvého cvičenia k téme triedenie sa na tento rozpor obvykle pýtajú.

Vstavaná funkcia *sorted* totiž nie je naprogramovaná v Pythone, ale v jazyku C rovnako ako iné časovo náročné časti štandardných Pythonovských knižníc. Aj keď v rámci tejto prednášky do ďalších podrobností nezachádzame, je veľmi dôležité vysvetliť tento zdanlivý problém. Pochopenie vysvetlenia je dôležité nielen pre zachovanie hodnovernosti tvrdení prezentovaných na predmete ADŠ ale aj ako ukážka limitov interpretovaných jazykov a akýsi návod do praxe, čo sa dá robiť keď rýchlosť programu v interpretovanom jazyku nie je dostatočná hoci používame teoreticky dobrý algoritmus.

### 3.5 Python skrýva niektoré detaily

Python ako vyšší programovací jazyk ukrýva pred programátorom veľa implementačných detailov. Ak by sme chceli študentov naozaj naučiť aj tie najväčšie implementačné detaily na úrovni správy pamäte, študenti by to mali programovať v jazyku C. Nám sa ale zdá dôležitejšie, aby radšej chápali problematiku zložitosti konkrétnej realizácie hoci v Pythone. Implementačné detaily môžu riešiť v ďalších semestroch štúdia, keď sa naučia ďalšie programovacie jazyky (napr. C++ v 3. semestri).

## 4. ZÁVER

V našom príspevku sme chceli oboznámiť kolegov s niektorými detailami nášho prístupu k výučbe predmetu ADŠ na našej fakulte. Poukazujeme na niektoré vlastnosti jazyka Python, pre ktoré ho odporúčame použiť v tomto predmete ako aj na niektoré také vlastnosti, ktorým treba pri výučbe venovať viac pozornosti aby neskomplikovali pochopenie vysvetľovaných pojmov ale aby naopak priniesli študentom nový pohľad na danú problematiku.

Veríme, že môže poslúžiť kolegom jednak na porovnanie našich a ich prístupov k výučbe ADŠ ako aj pri prípadných úvahách o zmenách či vylepšeniach v podobných predmetoch, ktoré učia na svojich školách.

## 5. BIBLIOGRAFICKÉ ODKAZY

- [1] Goodrich, M. T., Tamassia, R., Goldwasser, M. H.: *Data Structures and Algorithms in Python* (1st ed.). Wiley Publishing 2013, ISBN:1118290275 9781118290279.

DIDINFO 2015, Univerzita Mateja Bela, Banská Bystrica, 2015.

- [2] Domovská stránka Python.org. [online] [cit. 2015-02-09]. Dostupné na : <<https://www.python.org/>>
- [3] Stránka predmetu ADŠ. [online] [cit. 2015-02-09] Dostupné na: <<http://struct.input.sk>>
- [4] Interaktívny kurz Pythonu. [online] [cit. 2015-02-09] Dostupné na: <<http://interactivepython.org/runestone/static/pythonds/index.html>>

# Vyučovanie programovania vývojom androidových aplikácií v App Inventore

Monika Tomcsányiová

Katedra základov a vyučovania informatiky, FMFI UK  
v Bratislave  
Mlynská dolina  
848 48 Bratislava  
Slovensko  
tomcsanyiova@fmph.uniba.sk

Peter Tomcsányi

Katedra základov a vyučovania informatiky, FMFI UK  
v Bratislave  
Mlynská dolina  
848 48 Bratislava  
Slovensko  
tomcsanyi@fmph.uniba.sk

## ABSTRAKT

Zariadenia s operačným systémom Android sú pomerne rozšírené. Žiaci strednej, ale aj základnej školy, majú svoj vlastný mobil alebo tablet s Androidom a z Google Play si do neho sťahujú množstvo aplikácií, ktoré používajú vo svojom každodennom živote. Vývoj androidových aplikácií však nie je jednoduchý. Väčšinou si vyžaduje skúseného programátora v Jave alebo v inom profesionálnom programovacom jazyku. Autori prostredia App Inventor však ponúkajú programovacie prostredie, v ktorom aj začínajúci programátor, napr. so skúsenosťami v prostredí Scratch, dokáže naprogramovať jednoduchú aplikáciu. V článku predstavíme prostredie App Inventor a možnosti jeho použitia vo vyučovaní informatiky. Poukážeme na také koncepty prostredia, ktoré sa dajú využiť vo vyučovaní programovania pre žiakov na 2. stupni ZŠ, resp. v prvých ročníkoch strednej školy, t.j. pre žiakov vo veku 14-17 rokov. V článku tiež predstavíme naše vlastné tutoriály na vývoj niekoľkých mobilných aplikácií v tomto prostredí.

## ABSTRACT

Devices with Android operating system are fairly widespread. Pupils attending lower and higher secondary schools usually have their own mobile phone or tablet with Android and they download many applications from Google Play into their devices, which they use in their everyday life. However, development of Android applications is not easy. In most cases it requires an experienced programmer in Java or in other professional programming language. Authors of the App Inventor environment offer a programming environment, in which even a beginner programmer, e.g. with experience in Scratch environment, could program a simple application. In this article we introduce the App Inventor environment and possibilities of its use in teaching informatics. We point out concepts of this environment, which can be used in teaching programming to pupils attending lower or higher secondary school, i.e. for pupils aged 14-17 years. In this article we also present our own tutorials for development of several mobile applications in this environment.

## Kľúčové slová

vyučovanie, programovanie, Android, aplikácia, App Inventor

## Keywords

teaching, programming, Android, application, App Inventor

## 1. ÚVOD

Aplikácie pre androidové zariadenia sú v súčasnosti zrejme najštáhovanejšími programami z internetu. Každý, kto vlastní

mobilné zariadenie, hľadá na Google Play aplikácie, ktoré by mu umožnili čo najlepšie využiť svoj mobil alebo tablet. Naprogramovať profesionálnu, graficky zaujímavú a komerčne úspešnú aplikáciu však nie je jednoduché a väčšinou je na to potrebný tím vývojárov s dobrými znalosťami programovania v niektorom profesionálnom programovacom jazyku, ktorý podporuje vývoj mobilných aplikácií pre operačný systém Android. Tím výskumníkov a vývojárov na MIT však navrhol jazyk a programovacie prostredie, ktoré umožňuje vývoj androidových aplikácií pre žiakov bez predchádzajúcich programátorských skúseností. Navrhli a implementovali programovacie prostredie App Inventor, pozri [1], v ktorom sa aplikácia programuje priamo na webe pomocou skladania a spájania blokov príkazov, podobne ako v prostredí Scratch. Okrem toho autori a kolektív ich spolupracovníkov, pripravili aj niekoľko tutoriálov rôznej programátorskej náročnosti, v ktorých vysvetľujú základné programátorské nástroje, ktoré App Inventor ponúka. Na vývoji pomerne jednoduchých, ale napriek tomu zaujímavých, aplikácií v nich ukazujú možnosti programovania aplikácií v tomto prostredí.

## 2. APP INVENTOR

App Inventor, pozri [1], je programovací nástroj na vývoj androidových aplikácií, ktorý bol pôvodne navrhnutý v Google Labs. Od roku 2012 jeho vývoj prevzalo MIT Media Lab, spoločnosť Google však naďalej zostala v tomto projekte. Autori App Inventora ponúkajú v súčasnosti webový programátorský nástroj a prostredie, ktoré umožňuje navrhovať dizajn mobilnej aplikácie a program pre jednotlivé prvky tohto dizajnu (tlačidlá, textové okná, obrázky...).



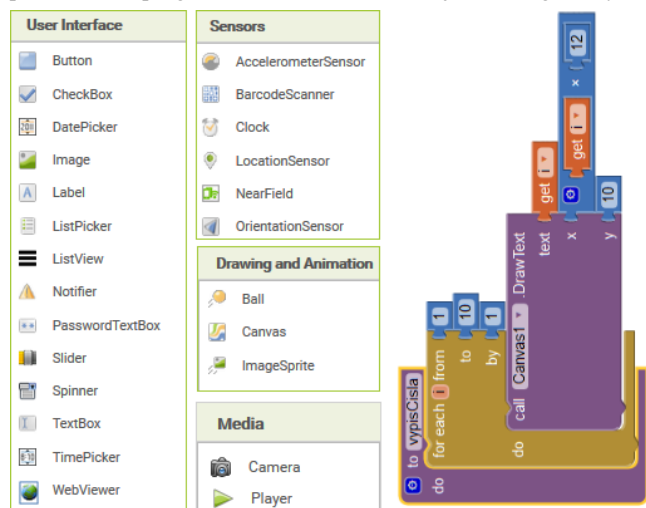
Obrázok 1. App Inventor, časť Designer.

Prostredie App Inventora navrhli autori tak, aby programovanie v ňom čo najviac zjednodušili a aby toto programovanie bolo pre žiakov prirodzeným pokračovaním programovania v prostredí Scratch. App Inventor obsahuje dve oddelené časti: Designer a Blocks. V časti Designer navrhujeme a vytvárame dizajn

aplikácie, t.j. kde a ako budú umiestnené vizuálne prvky aplikácie, ako sú napr. tlačidlá, grafická plocha, postavy. Po prepnutí do časti Blocks vytvárame program pre jednotlivé viditeľné aj neviditeľné prvky (komponenty) našej budúcej aplikácie. Programujeme pomocou ťahania predpripravených blokov príkazov pre jednotlivé komponenty. Každý komponent má v App Inventore pevne určené udalosti a vlastnosti, ktoré môžeme o komponente zistiť a tiež meniť, a tak vytvárať a programovať aplikáciu. Program v App Inventore vytvárame priamo na webe a laďme ho priamo na svojom pripojenom mobilnom zariadení. Spojenie webového prostredia s našim androidovým zariadením je veľmi jednoduché a stačí, ak je mobil alebo tablet v rovnakej bezdrôtovej sieti ako počítač, na ktorom programujeme. Výhodou takéhoto ladenia aplikácie je, že bežiacu aplikáciu vidíme a môžeme aj nedokončenú skúšať v reálnom čase priamo na tablete. Takže ak niečo vo svojom programe upravíme, môžeme si v bežiacей aplikácii ihneď vyskúšať funkčnosť zmeny, ktorú sme urobili. Po ukončení vývoja aplikácie môžeme aplikáciu exportovať a následne si ju nainštalovať na svoje mobilné zariadenie alebo ju poslať svojim priateľom. Takýto spôsob vyučovania programovania pomocou vývoja androidových aplikácií, interaktivita pri ich vývoji a následné zdieľanie hotovej aplikácie, môže byť zaujímavé a atraktívne pre žiakov vo veku 14 až 17 rokov, t.j. žiakov 8. a 9. ročníka ZŠ, resp. pre žiakov v prvých ročníkoch SŠ. Preto sme sa rozhodli pripraviť niekoľko metodických materiálov, pomocou ktorých môže učiteľ zaradiť programovanie v App Inventore do svojho vyučovania. V súčasnosti programovanie v App Inventore zaraďujú do svojich úvodných kurzov programovania niektoré stredné školy a aj niektoré VŠ, pozri [2], [3] alebo [4].

## 2.1 Dizajn a programovanie

App Inventor obsahuje a podporuje programovanie základných grafických prvkov, pomocou ktorých dokážu žiaci pripraviť nielen funkčnú, ale aj vizuálne zaujímavú aplikáciu, aké vyvíjajú profesionálni programátori a aké sa nachádzajú na Google Play.



Obrázok 2. Výber dizajnových komponentov pre aplikáciu a časť programu.

Do svojej aplikácie môžeme umiestniť napr. tlačidlo (Button), zaškrŕvacie políčko (CheckBox), popis (Label), vstupné textové okno (TextBox), grafickú plochu (Canvas), postavu (ImageSprite), časovač (Clock), posúvač (Slider), ale aj fotoaparát (Camera), prehrávač zvukov (Sound), či rozpoznávač hlasu

(SpeechRecognizer), pozri metodické materiály na zdieľanom google disku [5]. Pri programovaní aplikácií môžu žiaci používať jednoduché premenné (čísla, texty), pracovať so zoznamom, riadiť beh programu pomocou cyklu, vetvenia alebo programovaním udalostí. App Inventor podporuje aj vytváranie vlastných podprogramov bez parametrov aj s parametrami. Keďže pri práci s mobilnými zariadeniami je dôležitý aj dizajn aplikácie, pri programovaní v App Inventore je dôležitá vizuálna časť aplikácie a programovanie jednotlivých komponentov dizajnu musí byť pri programovaní aplikácie v súlade. Ak do našej aplikácie v časti Designer nevložíme žiadne komponenty, nemáme vlastne čo programovať.

Pri úvode do programovania je preto vhodné rozmyslieť si hneď pri návrhu aplikácie, aké prvky budú v aplikácii potrebné a aká bude ich funkčnosť a vložiť ich v časti Designer do novovytváranej aplikácie. V časti Blocks budeme potom navrhovať pre tieto komponenty program.

## 3. TUTORIÁLY

V nasledujúcej časti nášho článku ponúkame prehľad niekoľkých tutoriálov na vývoj aplikácií z oficiálnych stránok App Inventora, pozri [6]. Ide o námety na aplikácie, o ktorých si myslíme, že sú vhodné pre začínajúcich programátorov – žiakov vo veku 14-17 rokov. Pre niektoré námety sme navrhli vlastné rozšírenia a pripravili sme tiež niekoľko nových námetov na vyučovacie hodiny informatiky, ku ktorým sme pripravili vlastné tutoriály, pozri [5]. Veríme, že ak sa učiteľ rozhodne zaradiť tvorbu aplikácií v App Inventore do svojho vyučovania informatiky, pomôžu mu tieto materiály pri príprave jeho vyučovacích hodín.

### 3.1 Aplikácia Paint Pot

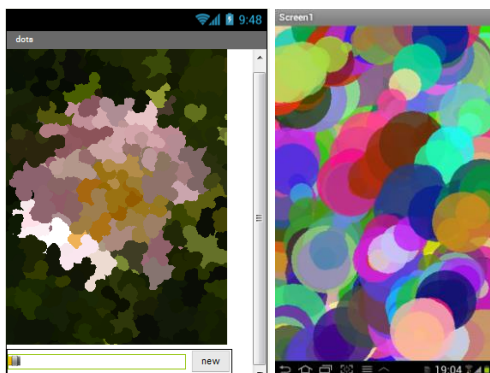
Aplikácia Paint Pot je jedna z prvých, pre ktorú autori App Inventora pripravili tutoriál. Je to program na kreslenie čiar do grafickej plochy (Canvas), pozri [7]. Z programátorských zručností sa v ňom žiaci naučia, ako vložiť do aplikácie grafickú plochu, ako nastaviť farbu pera a ako kresliť bodky (DrawCircle) a čiary (DrawLine). Ku koncu tutoriálu je uvedené, ako pridať do aplikácie tlačidlo, ktorým sa používateľ pomocou kamery vo svojom mobilnom zariadení odfotí a svoju fotku vloží ako pozadie aplikácie, pričom aplikácia mu umožní v tejto fotke pomocou bodiek a čiar dokresliť do fotky napr. fúzy, alebo ju inak upraviť. Uvedený tutoriál a koncepty, ktoré sú v ňom prezentované, sú vhodné na oboznámenie žiakov so základnými prvkami dizajnu a s metódami, ktoré App Inventor ponúka pre grafickú plochu a pre tlačidlá. Ako rozšírenie tejto aplikácie môžeme na ďalšej hodine, resp. pre šikovnejších žiakov, použiť aktivity naprogramovania našich aplikácií Veľa farieb, Vyobdskuj obrázok<sup>1</sup> a Kresli bodky<sup>2</sup>. Tie postupne nadväzujú na úvodnú aplikáciu a ukazujú žiakom ako zistiť farbu bodu grafickej plochy na mieste, kde sa jej dotkneme. Podľa toho program nastaví kresliacemu peru rovnakú farbu ako je farba bodu na obrazovke a nakreslí bodku konkrétnej alebo náhodnej veľkosti. Touto aplikáciou môžeme vytvoriť zaujímavý efekt rozostrenia obrázku, pozri Obrázok 3, vľavo. V aktivite Kresli bodky, sa žiaci naučia vygenerovať zoznam štyroch čísel a vytvoriť z neho farbu. To, že sa farba pera skladá z troch hodnôt 0 až 255, ktoré vyjadrujú zastúpenie troch základných farieb (červená, zelená, modrá),

<sup>1</sup> vyobdskuj.pdf

<sup>2</sup> kresli\_bodky.pdf



môžu žiaci poznať napr. z prostredia Imagine Logo, resp. z niektorého grafického editora. V App Inventore môže mať farba aj štvrtý komponent, ktorý vyjadruje jej priesvitnosť. Takto navrhnutá aplikácia umožňuje používateľovi pri ťahaní prsta kresliť body rôznej farby a priesvitnosti.



**Obrázok 3.** Dve rozšírenia aplikácie PaintPot: Vybodkuj obrázok (vľavo) a Kresli body (vpravo). Tutoriály ku nim nájdete v [5].

### 3.2 Aplikácia Android Mash

V tutoriáli aplikácie Android Mash, pozri [8], je vysvetlený koncept generovania náhodných čísel, fungovania a programovania časovača (Clock), vloženia postavy (ImageSprite) do grafickej plochy (Canvas). V návode je vysvetlenie programovania postavy pomocou udalostí a tiež práca s premennými. Táto aktivita je vhodná na oboznámenie žiakov s postavou a jej udalosťou po dotknutí sa postavy (Touch) a tiež s časovačom a jeho nastaveniami. Výskumy ukázali, že pri programovaní takýchto hier sa žiak začne zaujímať a chce vedieť vypísať, koľko krčkov sa mu podarilo chytiť. Takže následné vysvetlenie vytvorenia premennej a zmeny jej hodnoty bude žiak považovať za prirodzené a ľahšie pochopiteľné, ako keď sa vytváranie premenných ukazuje pri programovaní matematických úloh, napr. v aplikácii Kalkulačka, pozri napr. videotutoriál [9].

### 3.3 Aplikácia Digitálne hodiny

Ako pokračovanie vyššie uvedených tutoriálov sme navrhli metodický materiál na vytvorenie aplikácie Digitálne hodiny<sup>3</sup>.



**Obrázok 4.** Žiacke aplikácie Digitálne hodiny vytvorené pomocou nášho tutoriálu.

Pri jej programovaní si žiaci zopakujú prácu s časovačom a dozvedia sa, ako zistiť a zobrazit čas, ktorý je nastavený v ich mobilnom zariadení. Nami navrhnutú metodiku vyskúšala so svojimi žiakmi pani učiteľka zo ZŠ v Nitre a poslala nám fotku, pozri Obrázok 4, na ktorej vidíme hotové už bežiacie žiacke aplikácie digitálnych hodín. Rozšírením tejto aktivity môže byť návrh aplikácie Digitálne hodiny<sup>1</sup>, ktorá vypisuje čas priamo do grafickej plochy<sup>4</sup>.

### 3.4 Aplikácia Vynásob a povedz

Tutoriál Vynásob a povedz ukazuje využitie rozpoznávača hlasu, ktorý je súčasťou mobilných zariadení. V prostredí App Inventor ide o nevizuálny komponent SpeechRecognizer. V tutoriáli najprv pripravíme jednoduchú aplikáciu, v ktorej zistíme ako tento komponent funguje. Ďalej pokračujeme v návrhu a programovaní aplikácie, ktorá vygeneruje úlohu na násobenie, umožní žiakovi povedať výsledok násobenia a skontroluje, či je tento výsledok správny. Techniku rozpoznávania slov môžeme využiť aj neskôr pri programovaní iných aplikácií, napr. na vkladanie položiek do nákupného zoznamu, resp. vytvorenie hry, v ktorej má hráč podľa indícií uhádnuť slovo. V tutoriáli sme sa sústredili na vytvorenie jednoduchého dizajnu a na to, že už vytvorený dizajn aplikácie sa dá meniť aj počas jej behu a to zviditeľňovaním, resp. skrývaním niektorých objektov.

### 3.5 Aplikácia Poskladaj tvár klauna

Náš tutoriál programovania aplikácie Poskladaj tvár klauna<sup>5</sup> ukazuje, ako počas behu aplikácie meniť obrázky jednotlivých postáv (ImageSprite). Všetky postavy majú naprogramovanú udalosť TouchDown, pričom podľa toho, ktorej postavy sa dotkneme zmeníme klobúk, oči alebo ústa klauna na obrázok, ktorý je v súbore s príslušným názvom.

### 3.6 Aplikácia Zoznam priateľov

Pre pokročilejších programátorov sme pripravili tutoriál na vytvorenie aplikácie Zoznam priateľov<sup>6</sup>. Pri programovaní tejto aplikácie sa žiaci naučia vytvoriť a používať údajovú štruktúru – zoznam, vkladať a vypisovať zoznam pomocou komponentu ListPicker a používať komponent TinyDB na uchovávanie údajov aplikácie aj po jej ukončení.

## 4. ZHRNUTIE

Vo vzorových tutoriáloch z domovskej stránky App Inventora a našich vlastných tutoriálov, ktoré sme uviedli v tomto článku, sa žiaci oboznámia s možnosťami programovania v App Inventore, naučia sa používať a programovať niektoré vizuálne aj nevizuálne komponenty androidových aplikácií. Jednotlivé tutoriály, ktoré tu spomíname, sú zoradené do gradovanej postupnosti. Niektoré z nich prezentujú nové informatické koncepty, iné prehľbujú koncepty, ktoré žiaci nadobudli pri programovaní aplikácií podľa ďalších tutoriálov. Samozrejme v niekoľkých prezentovaných tutoriáloch nie je možné obsiahnuť veľké množstvo konceptov. Veríme, že do našej zbierky tutoriálov, pozri [5], budeme postupne pridávať ďalšie návody na programovanie mobilných aplikácií, ktoré budú rozvíjať napr. prácu s animáciami alebo

<sup>4</sup>digi\_hodiny1.pdf

<sup>5</sup>klaun.pdf

<sup>6</sup>zozn\_priatelov.pdf

<sup>3</sup>digi\_hodiny.pdf



ktoré prezentujú v App Inventore prácu so stringami. Pre tutoriály, ktoré sme navrhli uvádzame v Tabuľke 1 prvky dizajnu a programátorské koncepty, ktoré sú v nich uvedené, resp. ktoré sa viac rozvíjajú.

**Tabuľka 1. Zoznam aktivít, prvkov dizajnu a programátorských konštrukcií z tutoriálov.**

Paint Pot, Veľa farieb, Kresli bodky, Vybodkuj obrázok	Canvas, DrawCircle, DrawLine, Colors, <i>sekvencia príkazov, náhodné čísla</i>
Android Mash	Clock, ImageSprite, <i>premenné</i> , Label
Digitálne hodiny	Clock, Label, <i>náhodné čísla</i>
Digitálne hodiny 1	Clock, Canvas, <i>náhodné čísla</i>
Vynásob a povedz	SpeechRecognizer, zmena dizajnu počas behu aplikácie, <i>premenné</i>
Poskladaj tvár klauna	ImageSprite, zmena tvaru podľa obrázkového súboru
Zoznam priateľov	Lists, <i>vytváranie zoznamu, vkladanie a vyberanie zo zoznamu</i> , ListPicker, TinyDB

Tieto aktivity sú len úvodom do programovania v App Inventore. Ak majú žiaci väčší záujem o programovanie mobilných aplikácií, ďalšie tutoriály nájdu napr. v knihe [10].

Pri vytváraní tutoriálov a programovaní aplikácií v prostredí App Inventor sme našli aj niekoľko jeho obmedzení. V súčasnosti toto prostredie nepodporuje objektovo-orientované programovanie, t.j. neumožňuje navrhovať triedy a ani dynamicky vytvárať objekty. Ďalším nedostatkom, na ktorý sme narazili je pomerne zložitá naprogramovanie intuitívneho ťahania postavy (ImageSprite) po grafickej ploche (Canvas). Aj napriek týmto obmedzeniam však poskytuje App Inventor dostatočne veľa nástrojov na vývoj rôzne orientovaných mobilných aplikácií pre začínajúcich programátorov.

## 5. ZÁVER

Návrh a vytváranie vlastných androidových aplikácií je pre žiakov atraktívne a môže zaujať aj tých, ktorí nemajú záujem programovať v Pascale alebo v iných jazykoch a vytvárať

desktopové programy. Prezentovanými vzorovými a našimi tutoriálmi, sme sa snažili ukázať, že programovanie v prostredí App Inventor môže byť vhodné aj pre začínajúcich programátorov. Veríme preto, že vývoj mobilných aplikácií v App Inventore a ich prezentovanie v galérii alebo ich posielanie na mobilné zariadenie svojim spolužiakom, môže byť jeden zo spôsobov, ako priblížiť programovanie žiakom vo veku 14-17 rokov.

## 6. BLIOGRAFICKÉ ODKAZY

- [1] *App Inventor*. [online]. [cit. 2015-02-03]. Dostupné na: <<http://appinventor.mit.edu/>>
- [2] *Inventing Mobile Apps*. [online]. [cit. 2015-02-03]. Dostupné na: <<https://sites.google.com/site/wellesleycs117fall11/>>
- [3] *Computer Science and Business Technology at Humboldt CI*. [online]. [cit. 2015-02-13]. Dostupné na: <<http://hutchison-teach.ca/res-ai.html>>
- [4] Morelli, R.: Computing with Mobile Phones. Kurz AppInventora na Trinity College. [online]. [cit. 2015-02-06]. Dostupné na: <<http://turing.cs.trincoll.edu/~ram/cpsc110>>
- [5] Zdieľaný adresár s našimi tutoriálmi pre App Inventor [online]. [cit. 2015-02-06]. Dostupné na: <<https://drive.google.com/folderview?id=0BwZ8Q3yrkySRbEF6bVJkTEVIdW8>>
- [6] Wolber, D.: *App Inventor 2: Create your own Android Apps*. [online]. [cit. 2015-02-08]. Dostupné na: <<http://www.appinventor.org/book2>>
- [7] Paint Pot. *Tutoriál na vývoj mobilnej aplikácie*. [online]. [cit. 2015-02-03]. Dostupné na: <<http://www.appinventor.org/paintpot2>>
- [8] Android Mash. *Tutoriál na vývoj mobilnej aplikácie*. [online]. [cit. 2015-02-03]. Dostupné na: <<http://www.appinventor.org/AndroidMash>>
- [9] Kalkulačka. *Tutoriál na vývoj mobilnej aplikácie*. [online]. [cit. 2015-02-03]. Dostupné na: <<https://www.youtube.com/watch?v=m3t6YpzYap4>>
- [10] Wolber, D.: App Inventor Tutorials and Book Chapters. [online]. [cit. 2015-02-03]. Dostupné na: <<https://sites.google.com/site/appinventor/projects>>

# Elektronická učebnica pre iPad

Peter Trhan

Univerzita Mateja Bela, Fakulta  
prírodných vied, Katedra informatiky  
Tajovského 40  
974 01 Banská Bystrica  
Slovenská republika  
peter.trhan@umb.sk

Svetlana Gáperová

Univerzita Mateja Bela, Fakulta  
prírodných vied, Katedra biológie  
a ekológie  
Tajovského 40  
974 01 Banská Bystrica  
Slovenská republika  
svetlana.gaperova@umb.sk

Ján Gáper

Technická univerzita vo Zvolene,  
Fakulta ekológie a environmentalistiky,  
Katedra biológie a všeobecnej  
ekológie  
T. G. Masaryka 2117/24  
960 53 Zvolen  
Slovenská republika  
jan.gaper@tuzvo.sk

Ostravská univerzita v Ostravě,  
Přírodovědecká fakulta, Katedra  
biologie a ekologie  
Chittussiho 10  
710 00 Slezská Ostrava  
Česká republika  
jan.gaper@osu.cz

## ABSTRAKT

Aplikácia iBooks Author je nástroj pre vytváranie elektronických učebníc s interaktívnymi a multimediálnymi prvkami pre mobilné zariadenia a počítače od firmy Apple. Táto aplikácia generuje elektronické učebnice vo formáte ibooks, ktoré firma Apple nazýva iBooks textbooks. Pre takto vytvorenú učebnicu vo formáte ibooks môžeme podať žiadosť v Slovenskej národnej knižnici o pridelenie jej medzinárodného štandardného čísla knihy (ISBN), a preto aj diela vytvorené v aplikácii iBooks Author sú plnohodnotné knihy a zároveň môžu veľmi dobre poslúžiť ako učebné texty alebo učebnice v školách pre tablety iPad. Nedostatkom takto vytvorenej učebnice vo formáte ibooks je, že za poplatok sa pre licenčné podmienky môže táto učebnica šíriť len cez obchod iBooks Store v sekcii Textbooks.

## ABSTRACT

The iBooks Author application is a tool for creating electronic textbooks with interactive and multimedia features designed to be used with Apple mobile devices and computers. This application enables its user to generate electronic textbooks in ibooks format, commonly referred to by Apple as iBooks textbooks. The author of an ibooks-based textbook can apply for an International Standard Book Number (ISBN) with the Slovak National Library. The texts created by means of iBooks Author may thus be regarded as full-fledged books. At the same time, they can serve as useful learning texts or coursebooks at schools that use Apple iPads. A disadvantage of an iBooks textbook is that it can only be distributed for a fee via the Apple-run iBooks Store which offers these products in its Textbook section.

## Kľúčové slová

Učebnica, Textbook, iBooks Author, iPad.

## Keywords

Textbook, iBooks Author, iPad.

## 1. ÚVOD

Používanie mobilných zariadení s vhodným multimediálnym a interaktívnym obsahom počas vyučovania alebo samoštúdia

je pre žiakov, študentov mnohokrát zaujímavejšie a pútavejšie, ako používanie tlačenej učebnice. Na tento účel, t. j. čítanie elektronických kníh (elektronická kniha, e-kniha alebo eKniha – termín pre knihu v elektronickom formáte) sú lepšie mobilné zariadenia s väčším displejom a dostatočnou výdržou batérie, minimálne 8 až 10 hodín, aby ich bolo možné používať nepretržite celý študijný čas jedného dňa. Tieto požiadavky, ale s obmedzenými možnosťami pre interaktívne a multimediálne prvky spĺňajú čítačky eKníh a aplikácie pre tablety s podporou najpoužívanejších formátov pre eKnihy, ako sú formáty ePub, MOBI, AZW, KF8, PDF.

Čítačky eKníh disponujú veľkou výdržou batérie, ale tablety okrem čítania eKníh ponúkajú aj široké spektrum ďalších aplikácií pre bežnú prácu žiaka, študenta a učiteľa. Ak používateľ potrebuje mobilné zariadenie aj na čítanie a odosielanie emailov, vytváranie textových dokumentov, prezentácií, fotografií, videa, prehrávanie hudby, prezeranie webových stránok, kreslenie, komunikáciu, hranie hier a iných činností je lepším výberom kvalitný tablet s displejom s vysokým rozlíšením, kvôli vernému zobrazeniu s minimálnou únavou očí.

Bežný používateľ tabletu vyžaduje jeho intuitívne ovládanie, spoľahlivý, rýchly, jednoduchý na údržbu a relatívne bezpečný operačný systém a čo je najdôležitejšie širokú škálu kvalitných aplikácií, tzv. ekosystém aplikácií.

Naším cieľom je vytvorenie interaktívnej a multimediálnej elektronickej učebnice a okrem toho môžeme využiť tablety aj na čítanie eKníh. Na tvorbu takejto učebnice je potrebná aplikácia, vhodný komplexný nástroj, ktorý by bol bezplatný, naplno využíval možnosti ePub formátu, štylizoval texty, jednoducho vkladal tvary, grafy, tabuľky, obrázky, videá, zvuky, 3D objekty, matematické výrazy, hypertextové odkazy, interaktívne a ovládacie HTML prvky (HTML Widgets), HTML kódy, podporoval import súborov z textových editorov a iné užitočné funkcie.

Aby sme sa vyhli obmedzeniam pre univerzálne používanie elektronickej učebnice na výkonovo rôznych zariadeniach a mohli použiť všetky vyššie uvedené interaktívne a multimediálne prvky,

potrebujeme vytvoriť učebnicu pre konkrétny tablet s veľkým priestorom možností ePub formátu.

Ktoré aplikácie pre počítače spĺňajú uvedené požiadavky na vytvorenie multimediálnej a interaktívnej učebnice? Najjednoduchší spôsob na vytvorenie učebnice v ePub formáte je použitie textového editora a konverzia vytvoreného dokumentu do tohto formátu. Napríklad bezplatný textový editor Pages od Apple má priamu podporu konverzie do ePub formátu. Do textového editora Microsoft Word existuje bezplatné rozšírenie ePubee Maker, ktoré tiež zabezpečí bezproblémovú konverziu do tohto formátu. Dalšími podobnými aplikáciami sú napríklad Tablo (webová platforma) a Vellum. Pri textových editoroch, ktoré nemajú podporu exportu do ePub formátu, môžeme použiť automatické konvertory, ako sú napríklad Calibre a Stanza Desktop. V tomto prípade už môžu vzniknúť chyby pri konverziách zložitejších textových dokumentoch. Takýmto postupom však nezískame plnohodnotnú interaktívnu a multimediálnu elektronickú učebnicu, pretože môžu vzniknúť problémy s podporou jazyka HTML5 a nemáme tu k dispozícii HTML ovládacie prvky (HTML Widgets), uvedené aplikácie sú však dobré pri konverzii beletrie.

Lepším spôsobom na vytvorenie učebnice je použitie editora pre ePub formát. Pravdepodobne najpopulárnejší ePub editor je bezplatná aplikácia Sigil [11], podobou aplikáciou je editor eCub. Práca v týchto editoroch je náročnejšia, ako písať dokument v textovom editore a na zložitejšie formátovanie vyžaduje používateľa so znalosťou jazyka HTML5 a kaskádových štýlov. Najväčším nedostatkom týchto aplikácií je to, že vytvárajú ePub formát, ktorý je univerzálny pre zariadenia rôznych výrobcov a vytvorená učebnica sa nemusí zobrazovať správne práve na našom tablete alebo v rôznych aplikáciách na čítanie eKníh. Používateľsky príjemnejšími aplikáciami, ktoré neponúkajú HTML ovládacie prvky sú ešte Blurb BookWright, Byeink (webová platforma) a Book Creator (iOS a Android platforma).

Najlepšie je nájsť aplikáciu, ktorú vyvíja priamo výrobca tabletov a tým naplno využiť potenciál konkrétneho tabletu. Momentálne existuje takáto aplikácia a vyvíja ju firma Apple pre svoje počítače. Aplikácia iBooks Author je bezplatná a spĺňa všetky naše požiadavky na tvorbu multimediálnej a interaktívnej učebnice. Má však svoje obmedzenia pre distribúciu, ktoré sú spôsobené licenčnými podmienkami tejto aplikácie. Opisom tejto aplikácie sa budeme zaoberať v samostatnej kapitole.

Existuje však aj komerčná alternatíva aplikácie iBooks Author, ktorá je určená pre počítače Mac od Apple a jej výstupy sú optimalizované pre viaceré zariadenia a formáty eKníh. Táto aplikácia sa nazýva PubCoder a tiež ponúka interaktívne HTML ovládacie prvky a má rovnako dobré používateľské prostredie [12]. Cena tejto aplikácie je vysoká, licencia na jeden mesiac pre jedného používateľa je 50 €.

Pozrime sa ešte na niektoré existujúce projekty na podporu elektronických učebníc pre slovenské školy a na kompatibilitu ich výstupov pre vybraný tablet, ktorý by spĺňal vyššie uvedené požiadavky.

Webový portál eAktovka obsahuje bezplatné digitalizované učebnice pre základné a stredné školy, audio nahrávky, eKnihy (beletrie niektorých našich slovenských autorov pre predmet Slovenský jazyk a literatúra) a aj odbornú literatúru. Tento portál vznikol v rámci Rozvoja informatizácie regionálneho školstva a bol realizovaný Ústavom informácií a prognóz školstva [1]. Na tomto portáli sú elektronické verzie výstupov vo formáte PDF,

ePub alebo MP3. Uvedené formáty sú univerzálne použiteľné na všetkých tabletoch s operačným systémom Android aj iOS.

Nasledujúcim vybraným projektom je národný projekt Elektronizácia vzdelávacieho systému regionálneho školstva, ktorý realizuje Ministerstvo školstva, vedy, výskumu a športu SR v spolupráci s Metodicko-pedagogickým centrom [2]. Do projektu sa môžu zapojiť materské, základné alebo stredné školy, okrem škôl v Bratislavskom samosprávnom kraji. Zapojené školy môžu v rámci výpožičky získať aj tablety jednej značky Samsung Galaxy. Momentálne je vytvorený výstup formou webovej stránky [3], ktorý obsahuje prvých 50 lekcii digitálneho vzdelávacieho obsahu, ktoré sú konvertované do formátu HTML5 z webového portálu Planéta vedomostí [4]. Formát HTML5 je podporovaný webovými prehliadačmi na všetkých tabletoch. K týmto tabletom je dostupná platená aplikácia pre manažment tabletov Samsung School. Táto aplikácia je vhodná do tabletovej učebnice na monitorovanie, ovládanie tabletov, vykonávanie kvízov a ankiet, čo zabezpečuje interaktívnu spoluprácu medzi učiteľom a žiakmi [5][6].

Dalším projektom je projekt Škola na dotyk 2013 [7]. Týmto projektom získalo 10 vybraných škôl na Slovensku 300 tabletov Samsung Galaxy a aplikáciu Samsung School, ktoré školám ostali. Výstupné projekty zúčastnených škôl sú umiestnené na webovom portáli tohto projektu s sekcií Materiály. Projekty sú prevažne vytvorené online nástrojom TUUL, ktorý zabezpečuje online prezentáciu materiálov (obrázky, audio a video súbory, zip archívy, dokumenty, prezentácie, url adresy), ktoré si používateľ do svojho projektu vloží cez interaktívne prostredie [8]. Rovnako aj s týmto nástrojom sa dá pracovať s ľubovoľným tabletom.

Posledným projektom, ktorý opíšeme, je projekt spoločnosti WESTech (iStores.sk). Cez tento projekt si mohli učitelia všetkých stupňov škôl požičať jeden tablet iPad od firmy Apple s príslušenstvom a vyskúšať si ho na svojich vyučovacích hodinách. Výstupom projektu každého učiteľa boli tri prezentácie používania tabletu iPad a tri spracované vyučovacie hodiny odučené týmto tabletom. Do projektu sa zapojilo 137 učiteľov z celého Slovenska [9].

Zaujímavý článok „Tablety už učia aj na školách“ bol uverejnený v magazíne PROFIT 5/2013 a opisuje výhody aj nevýhody používania tabletov v školách a skúsenosti štyroch slovenských škôl vlastniacich tablety iPad [10].

Z nášho malého prieskumu používania tabletov v slovenských školách sme zistili, že cez projekty Ministerstva školstva, vedy, výskumu a športu SR sú na našich školách prevažne používané tablety značky Samsung. Tieto tablety pracujú s operačným systémom Android od firmy Google a každý výrobca tabletov si tento operačný systém prispôsobuje pre svoje zariadenia. Podpora aktualizácií operačného systému na takýchto zariadeniach je závislá na samotnom výrobcovi, z uvedeného dôvodu je táto podpora časovo krátkodobá, približne dva roky a majiteľ takéhoto tabletu sa neskôr musí uspokojiť so staršou verziou jeho operačného systému. Z tohto hľadiska sú na tom lepšie tablety od firmy Apple, ktorá si sama vyvíja operačné systémy pre svoje zariadenia. Napríklad na tablete iPad 2, ktorý sa začal predávať v roku 2011 si môžete aj dnes jednoduchým spôsobom nainštalovať najnovší operačný systém iOS 8.

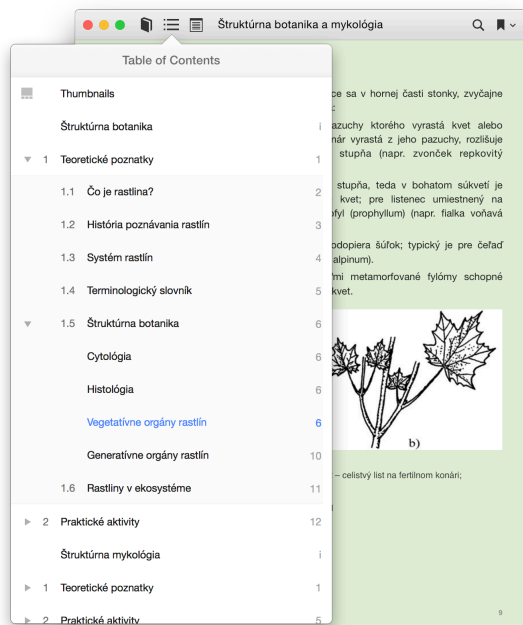
Aký konkrétny tablet si vybrať na tvorbu našej elektronickej učebnice? Z vyššie uvedených požiadaviek na formu učebnice, dostupnosti a kvality aplikácií na tvorbu takejto učebnice, dostupnosti terajších elektronických učebníc, elektronických

študijných materiálov, eKníh, edukačných aplikácií a obdobia použiteľnosti tabletu sme sa rozhodli pre tablet iPad od firmy Apple a jeho platformu iOS. Chýba nám len aplikácia Samsung School, ktorá je ale platená. Apple nám ponúka pre tieto tablety bezplatnú aplikáciu Apple Configurator, ktorá síce nemonitoruje tablety, ale umožňuje nám ich masovú konfiguráciu, inštaláciu, zálohovanie a obnovu. Do značnej miery výberu tabletu rozhodla bezplatná aplikácia aplikácia iBooks Author, kvôli jej cenovo nákladnejšej alternatíve PubCoder.

## 2. PROJEKT NA PODPORU TVORBY ELEKTRONICKEJ UČEBNICE

Na podporu tvorby multimediálnej a interaktívnej elektronickej učebnice Štruktúrna botanika a mykológia a jej využitie vo výučbe sme získali projekt KEGA č. 022-UMB/2013. Aj keď obsah učebnice nie je z oblasti informatiky, v tomto článku opisujeme hlavne jeho informatickú časť, ktorá môže pomôcť aj ďalším tvorcom elektronických učebníc, prípadne eKníh. Tento projekt nám zabezpečil finančné prostriedky pre nákup potrebnej techniky a odbornej literatúry, cestovné náhrady, konferencie na prezentovanie výsledkov projektu.

Pre riešenie projektu sme sa rozhodli v dôsledku absencie takýchto učebníc v tejto oblasti biológie v slovenskom jazyku. Pripravovaná učebnica bude mať široké využitie priamo vo vyučovacom procese, pri samoštúdiu, ale najmä pri realizácii praktických aktivít žiakov a študentov. Učebnica je určená študentom vysokých škôl prírodovedného zamerania, ale aj pre žiakov základných škôl a gymnázií. V rámci Štátneho vzdelávacieho programu sa štruktúrna botanika a mykológia na Slovensku vyučuje v nižšom sekundárnom vzdelávaní (ISCED 2) v 5. a 6. ročníku základných škôl, kde sa žiak prvýkrát oboznamuje so základnou terminológiou a získava prvé praktické skúsenosti z mikroskopovania. Vo vyššom sekundárnom vzdelávaní (ISCED 3A) sa štruktúrna botanika (cytológia, histológia, anatómia a morfológia rastlín) aj mykológia vyučuje v 1. a 2. ročníku gymnázia, na vysokých školách prírodovedného zamerania prevažne v prvom ročníku bakalárskeho štúdia.



Obrázok 1. Ukážka obsahu pripravovanej učebnice

Obsahovú náplň elektronickej učebnice tvoria dve časti – teoretická a praktická. Na obrázku 1 je ukážka učebnice s otvoreným obsahom v aplikácii iBooks. Súčasťou praktickej časti je databáza praktických úloh vytvorená študentmi a učiteľmi z výsledkov ich vlastných pozorovaní. Pri tvorbe tejto učebnice nadväzujeme na naše predchádzajúce publikácie [13][14][15].

Nakoľko štruktúrna botanika a mykológia obsahuje množstvo odborných termínov, je potrebné, aby ich študenti vedeli správne používať a aby získali čo najväčšie skúsenosti z praktických pozorovaní študovaných javov a práve interaktívna a multimediálna forma teoretickej časti učebnice pomôže k lepšiemu, rýchlejšiemu a názornému pochopeniu preberaného učiva. Taktiež veľmi užitočná bude aj praktická časť učebnice, v ktorej získajú študenti možnosť porovnať svoje reálne výsledky s inými výsledkami študentov, resp. učiteľa získanými za tých istých laboratórnych podmienok. Za týchto okolností môže laboratórium s tabletmi nahradiť učebňu vybavenú audiovizuálnou technikou a interaktívnou tabuľou.

V projekte sa bude realizovať aj nákup niekoľkých tabletov pre našich študentov, aby mohli pracovať s pripravovanou učebnicou priamo pri vyučovaní a laboratórnych cvičeniach pri praktických pokusoch a pozorovaniach.

Multimediálna a interaktívna verzia učebnice sa bude distribuovať bezplatne vo formáte ibooks cez obchod iBooks Store v sekcii Učebnice (Textbook) a na webových stránkach katedry. Pre tých, ktorí nemajú počítač alebo tablet od Apple, bude k dispozícii aj neinteraktívna verzia učebnice vo formáte PDF vyexportovaná z programu iBooks Author na webových stránkach katedry.

## 3. APLIKÁCIA IBOOKS AUTHOR

Aplikácia iBooks Author od Apple je autorský nástroj pre tvorbu multimediálnych a interaktívnych elektronických učebníc vo formáte ibooks. Formát ibooks vychádza z formátu ePub, ale na rozdiel od ePub formátu nie je otvoreným štandardom, t. j. nemá zdokumentované jeho dátové elementy a atribúty v značkovacom jazyku XML (jeho menný priestor), rovnako nie sú zdokumentované kaskádové štýly, ktoré zahŕňajú vlastnosti fontov, vykresľovanie matematických rovníc a hlavne interaktivitu dokumentu vo formáte ibooks [16]. Takto firma Apple rozšírila svoje dokumenty v ibooks formáte oproti bežným dokumentom v ePub formáte hlavne o interaktívne prvky formou HTML ovládacích prvkov (Widgets), ktoré môžeme používať len v ich aplikácii iBooks Author a dokumenty v nej vytvorené je možné čítať len v aplikácii iBooks, ktorá je dostupná pre operačné systémy OS X a iOS. Aplikácia podporuje export len do formátov PDF a TXT, tieto formáty môžeme čítať na ľubovoľných platformách, ale stratíme multimediálne a interaktívne prvky. Momentálne neexistuje konvertor z formátu ibooks do formátu ePub. Je to daň za to, že obe aplikácie sú zadarmo pre všetkých používateľov ich výrobkov. Ak by sme chceli podobnú aplikáciu pre ePub formát, napríklad aplikáciu PubCoder, musíme si za ňu zaplatiť.

Aplikácia iBooks Author ponúka jednoduché používanie nasledujúcich už vytvorených HTML ovládacích prvkov: galéria, médiá, test, prezentácia, interaktívny obrázok, 3D obrázok, rolujúci postranný panel, vyskakovacie okno a HTML. Posledný ovládací prvok umožňuje vytvoriť vlastný ovládací prvok pomocou jazykov HTML5 a JavaScript, kaskádových štýlov a obrázkov vo formáte PNG. Návod je uvedený na tejto webovej stránke [17].

Ovládanie aplikácie iBooks Author je prehľadné a intuitívne, podobné ovládaniu textového editora Pages. V službe YouTube sa nachádza množstvo návodov na ovládanie tejto aplikácie. Prvá väčšia príručka Jak na iBooks Author bola uverejnená v SuperApple magazíne 2012/3 [18]. Samozrejme aj firma Apple má návod k aplikácii na svojich webových stránkach [19] a dokumentáciu nájdeme aj v samotnej aplikácii v menu Pomocník.

Na slovenských a českých webových stránkach, kde sa opisuje aplikácia iBooks Author, sa vo väčšine prípadov uvádza, že pre učebnicu vytvorenú v tejto aplikácii nie je možné získať jej ISBN, kvôli licenčným podmienkam tejto aplikácie. V nasledujúcom texte bližšie rozoberieme tieto licenčné podmienky, ktoré sa týkajú distribúcie diel a možnosť získania ISBN pre našu učebnicu.

Medzinárodné štandardné číslo knihy (ISBN) môžeme v dnešnej dobe získať aj pre učebnicu vo formáte ibooks, ale o pridelenie ISBN musíme požiadať v Slovenskej národnej knižnici ešte pred poslaním učebnice na jej schválenie a následnú distribúciu cez službu iTunes Connect (služba iTunes Connect v tomto prípade manažuje predaj učebnice v obchode iBooks Store) prostredníctvom aplikácie iTunes Producer priamo cez sprievodcu z aplikácie iBooks Author. Podobným spôsobom by sme postupovali, keby sme chceli distribuovať eKnihu vo formáte ePub v obchode iBooks Store, lebo cez službu iTunes Connect a aplikáciu iTunes Producer nezískame jej ISBN.

V licenčných podmienkach aplikácie iBooks Author sa v časti distribúcie diel vygenerovaných touto aplikáciou uvádza:

- Ak je dielo poskytované bezplatne, môžeme ho distribuovať akýmkoľvek spôsobom.
- Ak je dielo poskytované za poplatok a obsahuje súbory vo formáte ibooks vygenerovanom aplikáciou iBooks Author, môžeme dielo distribuovať len prostredníctvom spoločnosti Apple, ale za predpokladu, že toto obmedzenie sa nevzťahuje na obsah diela, ak sa distribuuje vo formáte, ktorý neobsahuje súbory v ibooks formáte vygenerovanom aplikáciou iBooks Author. Ponechávate si všetky práva na obsah svojich diel a tento obsah môžete distribuovať akýmkoľvek spôsobom, ak neobsahuje súbory v ibooks formáte vygenerovanom aplikáciou iBooks Author.

Z uvedených licenčných podmienok vyplýva aj to, že ak učebnicu poskytujeme bezplatne, môžeme ju distribuovať aj sami vo formáte ibooks, ktorý nám vygeneruje aplikácia iBooks Author. V našom prípade nás licenčné podmienky nijakým spôsobom neobmedzujú, keďže učebnicu budeme distribuovať bezplatne a postačuje, keď bude mať čitateľ učebnice na svojom zariadení nainštalovaný program iBooks pre otvorenie dokumentov v formáte ibooks.

V opačnom prípade, ak by sme učebnicu vo formáte ibooks chceli predávať za poplatok, tak to môžeme urobiť len cez obchod iBooks Store. Ale ak zoberieme obsah našej učebnice a upravíme ho v niektorom textovom editore (t.j. nepoužijeme formát ibooks), tak takto upravenú učebnicu (napríklad aj ako eKnihu vo formáte ePub) môžeme predávať aj v iných obchodoch, nielen v obchode iBooks Store. Týmto spôsobom ale stratíme interaktívne a multimediálne prvky našej učebnice, ktoré sú v aplikácii iBooks Author už vytvorené a môžeme ich jednoduchým spôsobom používať.

## 4. ZÁVER

V tomto článku sme sa detailnejšie venovali požiadavkám na výber platformy a nástroja pre tvorbu našej pripravovanej učebnice. Pričom sme opisovali zaujímavosti, ktoré môžu byť užitočné aj pre iných autorov podobných učebníc alebo eKníh.

Osobne si myslíme, že eKnihy sú momentálne veľmi atraktívne, čomu veľmi pomohli mobilné čítacie zariadenia s kvalitnými displejmi a ich veľkou výhodou je hlavne to, že oproti tlačným knihám nezaberajú takmer žiadne miesto a ich počet na jednom zariadení obmedzuje len kapacita pamäťového média.

## 5. POĎAKOVANIE

Táto učebnica vzniká vďaka podpore Kultúrnej a grantovej agentúry Ministerstva školstva SR (grant KEGA022-UMB-4/2013). Autori ďakujú agentúre za finančnú podporu.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] MŠVVaŠ. eAktovka, zdigitalizované učebnice pre školy aj pre rodiny [online]. [cit. 2015-02-10]. Dostupné na: <<http://www.eaktovka.sk>>.
- [2] DIGIŠkola, Národný projekt „Elektronizácia vzdelávacieho systému regionálneho školstva“ [online]. [cit. 2015-02-10]. Dostupné na: <<http://www.digiskola.sk>>.
- [3] DIGIŠkola, digitálny vzdelávací obsah [online]. [cit. 2015-02-10]. Dostupné na: <<http://dvo.digiskola.sk>>.
- [4] MŠVVaŠ. Planéta vedomostí [online]. [cit. 2015-02-10]. Dostupné na: <<http://planetavedomosti.iedu.sk/index.php>>.
- [5] MŠVVaŠ. 2014. Samsung School prinesie školám skvalitnenie výučby s využitím tabletov [online]. [cit. 2015-02-10]. Dostupné na: <<http://www.minedu.sk/samsung-school-prinesie-skolam-skvalitnenie-vyucby-s-vyuzitim-tabletov/>>.
- [6] SAMSUNG. Samsung School v.1.5 [online]. [cit. 2015-02-10]. Dostupné na: <<http://www.samsung.com/global/business/mobile/solution/education/samsungschool>>.
- [7] Edulab. Škola na dotyk [online]. [cit. 2015-02-10]. Dostupné na: <<http://www.skolanadotyk.sk/o-projekte.html>>.
- [8] AGEMSOFT. TUUL, návod a FAQ [online]. [cit. 2015-02-10]. Dostupné na: <<http://tuul.sk/faq/>>.
- [9] WESTech. Projekt iPads pre učiteľov [online]. [cit. 2015-02-10]. Dostupné na: <<https://www.istores.sk/pages/skolenie/skolenie.html>>.
- [10] MRAVČÁK, A. 2013. Tablety už učia aj na školách. In *PROFIT 5/2013* [online]. [cit. 2015-02-10]. Dostupné na: <<http://profit.etrend.sk/dennik/tablety-uz-ucia-aj-na-skolach.html>>.
- [11] Sigil Ebook, Sigil is a multi-platform EPUB ebook Editor [online]. [cit. 2015-02-10]. Dostupné na: <<http://sigil-ebook.com>>.
- [12] PubCoder, features [online]. [cit. 2015-02-10]. Dostupné na: <<http://www.pubcoder.com/en/features/>>.
- [13] HINDÁK, F. – ŠOLTÉS, R. – GÁPER, J. – GÁPEROVÁ, S. – KYSELOVÁ, Z. – HINDÁKOVÁ, A., 2009. Botany – Lower plants/Botanika – Nižšie rastliny. Žilina : IHMB, 2009. 202 s.

DIDINFO 2015, Univerzita Mateja Bela, Banská Bystrica, 2015.

- [14] GÁPEROVÁ, S. – ROTH, P., 2007. *Anatómia a morfológia rastlín*. Banská Bystrica : Univerzita Mateja Bela, 2007. 146 s.
- [15] GÁPER, J. – GÁPEROVÁ, S., 2000. *Systematická botanika. Úvod do štúdia cyanobaktérií, rias a húb*. Banská Bystrica : Fakulta prírodných vied Univerzity Mateja Bela, 2000. 87 s.
- [16] Stack exchange. *iBooks vs epub: What's the difference?* [online]. [cit. 2015-02-10]. Dostupné na: <<http://apple.stackexchange.com/questions/130265/ibooks-vs-epub-whats-the-difference>>.
- [17] Apple. 2014. *iBooks Author: About HTML widget creation* [online]. [cit. 2015-02-10]. Dostupné na: <<http://support.apple.com/en-us/HT5068>>.
- [18] KREJČA, R., 2012. Jak na iBooks Author. In: *SuperApple magazín*. ISSN 1805-0212, 2012, č. 5–6, s. 82–91.
- [19] Apple. Mac Apps Support, iBooks Author [online]. [cit. 2015-02-10]. Dostupné na: <<https://www.apple.com/support/mac-apps/ibooksauthor/>>.



# PROGRAMOVÁNÍ VE SCRATCH BADATELSKY ORIENTOVANÝM PŘÍSTUPEM

Jiří Vaníček  
Jihočeská univerzita  
Jeronymova 10  
37115 České Budějovice  
Česká republika  
vanicek@pf.jcu.cz

## ABSTRAKT

Badatelsky orientovaná výuka je v poslední době frekventovaným termínem mezi moderními přístupy k výuce především přírodovědných předmětů. Vystala otázka, zdali tento přístup lze využít při výuce informatiky, konkrétně u výuky programování v didaktickém programovacím prostředí. V projektu výuky základů programování na 2. stupni ZŠ při praxi u budoucích učitelů bylo akčním výzkumem zkoumáno, zdali některý z přístupů a typů předkládaných aktivit odpovídá badatelsky orientované výuce. Přitom jsme získali zkušenosti s tím, jak jsou budoucí učitelé schopni vést takovou výuku, jakým způsobem dnešní žáci k učení se programování přistupují, nakolik je téma či konkrétní aktivity oslovovaly a jak vnímali tuto výuku jejich učitelé – studenti učitelství informatiky.

## ABSTRACT

Inquiry-based learning is recently one of often used approaches to natural sciences education. A question whether this approach is usable by computer science teaching has arisen especially by programming in educational programming environment. In the project of teaching basic of programming at lower secondary school as a part of pedagogical praxis of future teachers, several types of learning activities including elements of inquiry-based learning were tested in action research. In this article we describe findings how teachers without teaching experience are able to teach this method, how are pupils approaching programming and how much this topic and specific activities are perceived by their teachers – pre-service teacher students..

## Klíčová slova

Výuka programování, badatelsky orientovaná výuka, 2. stupeň ZŠ, Scratch.

## Keywords

Teaching programming, inquiry-based learning, lower secondary school, Scratch.

## 1. ÚVOD

Celosvětový trend zavádění informatického obsahu do výuky na nižších stupních škol v minulém roce pozitivně zasáhl i Českou republiku v podobě přijetí *Strategie digitálního vzdělávání do r. 2020* vládou ČR. V tomto strategickém dokumentu je jako jeden ze tří prioritních cílů uveden rozvoj informatického myšlení žáků [1, s.14]. Lze očekávat, že realizace tohoto vládního usnesení bude mít dopad na reformu Rámcových vzdělávacích programů pro 1. i 2. stupeň základní školy a tudíž pro zavádění informatického kurikula do výuky v daleko větším měřítku než dosud. S tím souvisí nutné vzdělávání učitelů informatiky a poskytnutí metodiky výuky informatiky těmto učitelům, především těm bez informatického vzdělání.

Výuka matematiky a přírodovědných předmětů na základních a středních školách se dlouhodobě potýká s nevhodnými metodami výuky, založenými na reprodukci poznatků, na instrukcích a tutoriálech, které nevedou k porozumění učivu na hlubší úrovni a v důsledku nerozvíjejí přemýšlení. Reakcí na tento stav je tzv. badatelsky orientovaná výuka (BOV, inquiry based learning, IBL), výuková metoda, při jejímž užití jsou žákovy znalosti vytvářeny systémem kladených otázek a řešením problémů [2]. Student pozoruje reálný svět a na základě zjištěných výsledků navrhuje možná vysvětlení zkoumaného problému [3]. Podle Papáčka se učitel stává zasvěceným průvodcem při řešení problému a vede žáka způsobem obdobným jako při skutečném výzkumu [2]. V souladu s konstruktivistickými teoriemi učení je student aktivní, řeší problémy, získává zkušenosti a je veden k vytváření kognitivních modelů jevů, které pozoruje. Aktivní a kreativní pozice studenta ve výuce je pro úspěšné učení se nutnou podmínkou [4].

Přístupy zahrnované pod BOV lze členit podle toho, které části badatelského procesu přebírá za žáky či studenty učitel. Eastwell [5] rozlišuje čtyři podoby bádání (inquiry): potvrzující, strukturované, nasměrované a otevřené. Přitom platí, že

- při potvrzujícím bádání je úkolem žáka ověřit známé výsledky, přičemž otázky i postup jsou mu známy;
- při strukturovaném bádání učitel sděluje otázky a žák má na základě implementace poskytnutého možného postupu formulovat vysvětlení;
- při nasměrovaném bádání učitel dává výzkumnou otázku, žák vytváří metodický postup a realizuje jej;
- při otevřeném bádání sám žák klade výzkumnou otázku, na jejímž základě promýšlí postup, realizuje jej a formuluje vysvětlení.

Je otázkou, nakolik lze badatelským přístupem vyučovat informatiku, konkrétně programování (nebo obráceně, nakolik lze prostředí světa informatiky využít k vzdělávání člověka badatelským způsobem). Nakolik lze jednotlivé fáze badatelsky orientované výuky realizovat při výuce programování bez toho, abychom zásadně měnili výukové schéma a příliš neměnili prioritu výukových cílů? Jaké prvky výuky je třeba akcentovat, aby žák prošel fázemi podle Bella et al. [3], jako jsou orientace v problému a kladení otázek, generování hypotéz, plánování designu experimentu a jeho realizace, analýza a interpretace získaných poznatků, vytvoření závěru a evaluace?

Jistě lze předpokládat, že při vhodně vedené výuce podle metodik, založeným na konstrukcionistickém přístupu, budou žáci objevovat, mohou experimentovat, být vedeni k tvorbě a ověřování hypotéz, budou evaluovat a odhalovat příčiny pomocí analýzy chování systému. Je ovšem možné tyto dílčí aktivity vůbec nějak sklobit do jednoho organického celku, který by

"vytvářel ze žáka mladého badatele"? Nabízí informatika stejně plodné pole pro BOV jako ostatní přírodní vědy? Umožňují didaktická programovací prostředí realizovat "reálné" experimenty stejně jako biologické, fyzikální a chemické pokusy se skutečnými artefakty a přírodními jevy? Poskytuje téma programování dostatečně vhodné prostředí, tak aby orientace na BOV neodváděla pozornost od hlavního cíle, rozvíjet člověka algoritmicky myslícího, schopného uchopit a řešit problém, modelovat situaci a tento model implementovat do počítače?

Toto jsou otázky, na které není snadné položit odpověď. My jsme se u jejich vědomí proto nezaměřili na změnu paradigmatu výuky, zůstali jsme u snah modifikovat standardní výuku programování zařazením některých prvků BOV (přičemž některé z nich se s konstrukcionistickým přístupem kryly).

## 2. PROJEKT VÝUKY

Semestrový projekt výuky programování na základní škole s prvky badatelsky orientované výuky v prostředí Scratch [6] proběhl na podzim 2014. Výuka programování byla realizována na jedné učební skupině 13 žáků 9. třídy běžné městské školy, která měla na české poměry nadprůměrnou dotaci informatiky (po 1 hodině v 5., 6. a 8. ročníku). Obsahem předchozí výuky nebyla žádná informatická témata, pouze uživatelský přístup (např. kancelářský software, Internet a prezentace, úpravy fotografií, vektorová grafika). Výuku v projektu vedli studenti učitelství posledního ročníku magisterského studia v rámci své pedagogické praxe a při výuce se střídali, obsah výuky také pod odhledem metodika praxe připravovali. Bylo tak možno pozorovat jak přístup žáků, tak i učitelů bez zkušeností s tímto tématem.

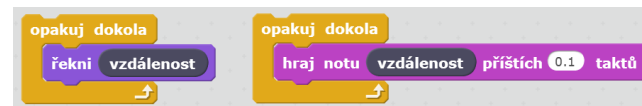
Výuka probíhala zhruba podle osnovy vytvořené Krejsou [7], který porovnal programovací prostředí Scratch a Imagine, provedl analýzu metodických postupů výuky od různých autorů především z pohledu zařazování nových programových struktur, prvků a nástrojů a na základě této analýzy sestavil vlastní metodický postup. Náměty úloh byly brány z práce Krejsy [7], z programovacích aktivit podle Mikolajové [8], z internetového úložiště Scratch a byly přetvářeny ponejvíce po stránce příběhu, který daná aktivita ztvárňovala. Po úvodních hodinách, věnovaných seznámení s prostředím, následovaly vlastnosti objektů (např. kostýmy), cykly a rozhodování, spouštění akcí, komunikace mezi objekty, synchronizace mezi vlákny, proměnné. Rozsah výuky byl 15 hodin, 1 hodina týdně.

Abychom uplatnili zásady badatelsky orientované výuky, učitelé se snažili připravovat a vést výuku tak, že s žáky vytvářeli komplexnější úlohy, např. jednoduché počítačové hry nebo příběhy postav. Učitelé rozdělili tyto úlohy na dílčí, které předkládali žákům k řešení, vedli tak žáky sérii kratších úloh s motivací vytvořit rozsáhlejší dílo. Žáci přitom měli možnost experimentovat, argumentovat a vysvětlovat své postupy a později si kontrolovat nebo opsat řešení tak, aby v navazující úloze mohli pokračovat. Tento styl výuky jsme připravovali s vědomím, že učitelé jsou úplnými začátečníky; na druhou stranu bylo výhodou, že nepřistupovali k výuce s předsudky.

Do výuky (nikoliv na její závěr) byl zařazen kratší programovací projekt interaktivního vánočního přání, jehož obsah byl zcela na žákovi a jehož realizace probíhala většinou doma, ovšem nejdůležitější části projektu, úvod a závěr, a také konzultace k žakovským řešením dílčích prací probíhaly ve škole.

Na závěr výukového celku byla realizována dvouhodinová aktivita v duchu badatelsky orientované výuky, v níž měli žáci naprogramovat hru, v níž bude hráč hrát proti počítači. Jejím

popisu a výsledkům jsme věnovali kapitolu 3.3 článku. Celé téma bylo zakončeno exkurzí na Pedagogické fakultě, při níž žáci pracovali s robotickými stavebnicemi LEGO WeDo, ovládanými z prostředí Scratch. Dostali k dispozici sestavené roboty vybavené motory, světly a čidly vzdálenosti a programovali jednoduché úkoly, např. větrák se střídavým chodem, bezpečný fén (který se zastaví po přiblížení ruky), digitální měřič vzdálenosti, stroboskop, bezdotykový hudební nástroj.



**Obrázek 1. Jádru programu digitálního měřiče vzdálenosti (vlevo) a bezdotykového hudebního nástroje (vpravo) ve Scratch pro robota, sestaveného ve stavebnici LEGO WeDo.**

Na každou odučenou hodinu připadaly 3 hodiny přípravy, která zahrnovala jednak skupinový rozbor a shrnutí pozorování, jednak konzultace přípravy kurikula následující vyučovací jednotky s metodikem a s kmenovým učitelem. Mimo tento režim probíhaly analýzy prací žáků.

## 3. VÝSLEDKY

Po zkušenostech s výukou výše uvedeného projektu můžeme konstatovat, že moderní výuka programování v didaktických programovacích prostředích s metodikami vyvinutými v posledních desetiletích má v sobě řadu prvků badatelsky orientované výuky, zvláště pokud se dá žákům prostor k vyjadřování, k řešení (dílkových) problémů. Základní rozdíl vidíme v cíli, kdy při BOV je cílem onen badatelský přístup, vedoucí k poznatkům, ke správnému uvažování, zatímco u programování je cílem využít takového uvažování či poznatků k vytvoření produktu, tedy programu pro počítač. V projektu byly použity především postupy potvrzujícího a nasměrovaného badatelského přístupu.

Kvalita výuky a splnění výukových cílů poměrně souvisely s tím, jakým způsobem učitel výuku připravil. Učitelé připravovali kuriulum, složené ze dvou druhů aktivit:

- krátké programovací úlohy (etudy), jejichž cílem bylo procvičit konkrétní dovednost, konkrétní poznatek
- větší programovací celky (projekty), jejichž cílem bylo vytvořit rozsáhlejší dílo, např. hru

Výuka pomocí série programovacích etud o délce několika minut směřovala vždy k naučení konkrétní dovednosti, byla orientována na konkrétní poznatek. Pro učitele bylo snadnější takovou aktivitu připravit a porozumět tomu, jak z takovýto aktivit hodinu seskládat. Často tyto aktivity neměla tak silnou motivaci jako u větších projektů, běžný žák byl při těchto úlohách více samostatný, počítač poskytoval dostatečnou zpětnou vazbu, ovšem nebyla rozvíjena žákova tvořivost.

Uvědomujeme si, že označení větších programovacích celků jako projektů je nepřesné, protože šlo spíše o sérii navazujících úloh, vedoucích k vytvoření produktu, jenž např. v Imagine bývá projektem nazýván. Šlo nicméně o změněný přístup v tom, že žákům nebyl tento postup tvorby programu předkládán, ale aktivně jej vytvářeli, a to na základě otázek od učitele. Výuka pomocí programování větších děl jako série navazujících úloh byla především na začátku každé aktivity silně motivační, ovšem učitel ani nepředpokládal, že by žák mohl daný komplexní problém rozdělit na dílčí části, a tuto práci dělal za žáka.

### 3.1 Přístup a chyby učitelů

Učitelé častěji volili výuku tak, aby během hodiny vznikl větší projekt, viděli v něm větší motivaci pro žáky a patrně i pro sebe. To jim ovšem přinášelo problémy především v tom, že zvolené projekty, jak se při jejich přípravě ukazovalo, pro svůj chod velmi často potřebovaly dílčí znalosti, které žáci ještě neprobírali a jejichž naučení by z časových i koncepčních důvodů nebylo realizovatelné (takovým příkladem bylo zařazení klonování pro potřeby hry). Učitelé pak danou novou pasáž prostě žákům předložili buď jako již vytvořený kód, nebo k opsání bez hlubšího vysvětlení. Problém byl i v organizaci času; tyto aktivity často přesahovaly do další vyučovací hodiny a tak bouraly její koncepci jednak proto, že konec aktivity odučil jiný učitel, jednak proto, že jemu ubírala prostor pro realizaci jeho vlastního kurikula.

Z chování našich studentů učitelství můžeme vyvozovat, že podobně by pracoval učitel, který nemá potřebné základy a chtěl by s žáky především vytvořit nějaký naprogramovaný produkt. Vlastní ambice učitele „něco velkého s dětmi udělat“ se pak může podepsat na splnění cílů výuky, na nekoncepčním vedení výuky, na nedostatečném procvičení základů s důsledkem pozdější neschopnosti naprogramovat i jednodušší úkoly.

Druhá chyba, již se učitelé při projektování výuky dopouštěli, spočívala v tom, že příliš brzy přecházeli k vytváření dlouhých víceřádkových kódů, v nichž se žáci hůře orientovali, a nenapadala je řešení, při nichž by se tomuto mohli vyhnout např. použitím více vláken, komunikace mezi objekty apod.

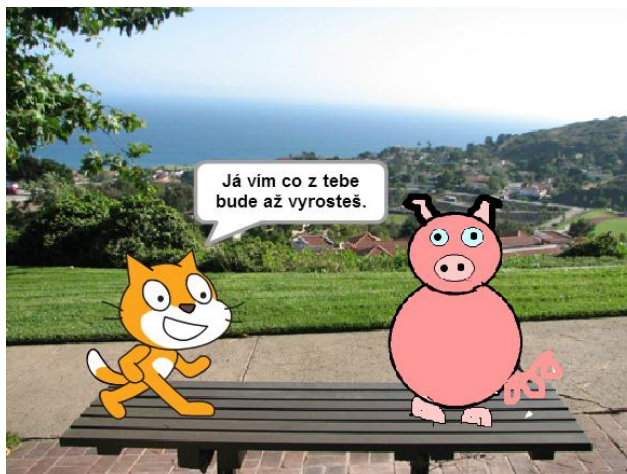
Typickou chybou, která má příčinu v samotném prostředí Scratch, je používání nekonečného cyklu, které Scratch nabízí jako elegantní řešení při počáteční výuce. Aktivity za použití této struktury jsou motivační a v úvodu do tematického celku dokážou žáky k programování přitáhnout. Žáci i učitelé však měli později potíže se tohoto zlovyku zbavit a místo jednoduchého cyklu s podmínkou vytvářeli nekonečné cykly s vnořenými rozhodováními, ukončováním běhu kódu natvrdo.

Jiným problémem, který umožňuje jednoduché objektové programování, je špatná synchronizace, která je často uživateli hlouběji skrytá a přidělá mu (i jeho nezkušenému učiteli) potíže, jejichž příčinu nedokáže odhalit a pak chybu svádí na podivné chování aplikace. Uvedeme příklad, který jsme pozorovali. Při tvorbě hry „kočka chytá myš“ žáci stanovili, že pokud myš narazí do kočky, odečte se bod ve skóre, kočka řekne „Mňam“ a myš skočí opět na začátek stránky (obr. 3 nahoře). Skóre bylo na začátku hry nastaveno na počet myší, které je třeba ulovit. Nevhodnou synchronizací příkazů k odeslání a přijetí zprávy s příkazem k přeskočení myši na začátek stránky docházelo k tomu, že než kočka dořekla svoji řeč, program mnohokrát zkontroloval, zda se myš dotýká kočky a mnohokrát tak odečetl skóre. Tím se skóre stihlo vynulovat již při chycení první myši. Tato chyba v kódu je pro učitele – začátečníka obtížně detekovatelná a naši studenti učitelství ji nedokázali v dané situaci odhalit.

Učitelé měli též hodnotit práce žáků. Bylo zajímavé sledovat, jak si přitom počínají. Vesměs platilo, že učitelé kontrolovali, zda program řádně funguje, případně u projektu sledovali, které z vyučovaných programovacích prvků a technik žáci ve svých pracích použili (snažili se kvantitativně vyhodnotit, zda žáci použili cyklus, zprávy, rozhodování, kostýmy apod.) Při tomto hodnocení nebrali v potaz, zda žák použil originální techniku nebo přístup či vtipné řešení netradičním nasazením jiných programovacích prvků (např. odebrání zápalek lze vizuálně řešit skryváním jednotlivých objektů zápalky interakcí nebo změnou

kostýmu sady zápalek jako jednoho objektu, postupným překrýváním zápalek na pozadí jiným popojíždějícím objektem atd.).

Námět práce či nápad učitelé často do hodnocení nezahrnovali a pokud podle jejich představy námět nesplňoval pravidla etikety, nezařadili jej mezi prezentované nejlepší, i když ostatními kritérii splňoval (např. projekt na obr. 2 dole). Zde byli učitelé na žáky příliš pruderní oproti tomu, jak se sami chovali a vyjadřovali – budilo to dojem uplatňované dvojí morálky nebo nejistoty, zda škola takto svobodné vyjadřování může tolerovat a kde je hranice takové tolerance.



Obrázek 2. Kreativita žáků u domácích úkolů. Komiks – vtip. [10]

### 3.2 Jak reagovali žáci

Po první hodině byli žáci spíše rozpačití a rozhodně nevypadali z programování nadšení patrně z důvodu, že se nedokázali zorientovat v prostředí a v typu mentální činnosti, který bude při výuce vyžadován. Po druhé hodině se díky motivaci k vytváření příběhů asi třetina žáků pro programování nadchla – pravidelně vytvářeli dobrovolné domácí úkoly, což se následně projevilo v tom, že tito žáci byli jistější při výuce, častěji se hlásili a pracovali samostatně.

Po prvních 7 hodinách výuky během 2 měsíců nebyl ve znalostech žáků vidět významný pokrok. Ti až na výjimky nebyli schopni úlohy řešit samostatně, nepamatovali si příkazy a techniky práce, nebyli schopni objevit, kterou techniku nebo který příkaz v dané



situaci použit. Modifikace prací při domácích úkolech spočívaly převážně v námětu nebo úpravě pozadí a kostýmů, jen výjimečně se objevovaly změny v psaných scénářích. Přesto žáci na vložení Hodině kódu, při níž programovali sadu úloh želví grafiky s motivací z filmu Frozen [10], dokázali jednoduché programovací úlohy řešit. Mohli jsme ovšem u některých slabších žáků pozorovat bezradnost ve chvíli, kdy nevhodně poskládali sekvenci příkazů do těla cyklu a nedokázali přijít na žádný způsob, jak chybný program odstranit, takže se uchýlovali k metodě pokus – omyl bez jakékoliv systematickosti (např. že by zkoušeli všechny možnosti).

Při pozdější analýze jsme usoudili, že žáci dostávali málo pestré programovací úlohy, založené pouze na vytváření kódu, nikoliv např. na opravách chybného kódu, při nich by museli odhalovat příčinu chyby. Zdálo se nám totiž, že aktivity Hodiny kódu spíše připomínají opravy programu než jeho tvorbu. Také pouhých několik hodin výuky zřejmě nestačí k dostatečnému zafixování byť jen elementárních základů programování a k zažití přístupu k práci v programovacím prostředí.

Největší pokrok ve znalostech žáků jsme překvapivě zaznamenali po absolvování krátkého projektu vytvoření elektronického vánočního přání na vlastní téma. Žáci museli přijmout vlastní zodpovědnost, a to byl možná hlavní faktor, který rozhodl o tom, že sice samotné projekty až na výjimky nepřinášely originální nápady jak tematické, tak programovací, ale po skončení projektu a další třítydenní pauze bylo pozorováno, že žáci lépe rozumí vysvětlování a jsou schopni samostatněji pracovat a zdůvodňovat.

Téma programování dokázalo hned zpočátku oslovit 4 žáky, kteří se mu nadstandardně věnovali, např. řešili dobrovolné domácí úkoly. Tito žáci, kteří se věnovali programování i o svém volnu, měli také výjimečné nápady a dokázali je uplatnit i v hodinách variováním témat programovaných projektů (např. hra kočka s myši na chytání fotbalového míče – obr. 3, rozhovor mezi postavami na scénku s potravinami v žaludku – obr. 2 dole). Tito žáci si brzy uvědomili, že povaha různých činností ze života má z hlediska programování stejnou strukturu a lze být modelována stejným způsobem.

Jako úspěšné se jevíly aktivity, kde žáci pokud možno neprogramovali, ale vymýšleli scénáře pro postavy, dialogy, kreslili kostýmy, tedy zabývali se činnostmi, v nichž uživatelsky měnili námět, ale nemodifikovali program. Pokud k změnám kódu docházelo, jednalo se většinou o jednoduché příkazy, spouštěné událostmi, o lineární příběhy; nevyskytoval se cyklus nebo rozhodování. Z hlediska naučení konkrétních kompetencí se nám jevíly přínosnější programovací etudy, které šly v pochopení látky více do hloubky.

Sledovali jsme, u kterých námětů úloh žáci nejlépe přijímali nové poznatky:

- kreslení vlastních kostýmů postav: při vlastní modifikaci námětu aktivity s minimálně měněným programem
- proměnné: tehdy, když chtěli sledovat skóre hry
- podmínky: tehdy, když chtěli ukončit hru
- zprávy: když chtěli řídit rozhovor dvou a více postav

Žáci raději vytvářeli programy, v nichž se nevyskytoval cyklus (dávali přednost jednoduchým spouštěným akcím ve více vláknech nebo nekonečné smyčce - preferovali tedy jednoduchý kód). Zda je to dáno nepřípravou Scratche používat zabudovaný počítač cyklů (jako je *repcount* v Imagine), nevíme, protože k situaci, kdy by byl potřeba pravý cyklus FOR, jsme se nedostali.



Obrázek 3. Tatáž hra ve dvojím hávu: kočka chytá myš a fotbalové penalty. Nahoře vyučovaná verze, dole spontánní žákovská modifikace.

### 3.3 Programování herní strategie

Na závěr výukového celku byla realizována dvouhodinová aktivita v duchu badatelsky orientované výuky, jejímž cílem bylo naprogramovat hru hráče proti počítači. Žáci měli nejprve sami odhalit výherní strategii u jednoduché deskové hry odebrání zápalek NIM [11] a teprve potom implementovat objevená pravidla do programu pro počítač. Podobná aktivita byla realizována v předvýzkumu v r. 2013 u stejně starých žáků. Ti nejprve naprogramovali tuto hru pro dva hráče, následně ji hráli sami proti sobě s cílem porazit druhého, objevit vítěznou strategii a tu posléze formulovat tak, aby ji bylo možno převést na instrukce programu, v němž bude počítač hrát proti živému hráči.

Cílem této aktivity byl i světonázorový aspekt, při němž si žáci měli uvědomit, že „inteligentní počítač“ je třeba také naprogramovat, tedy že je to člověk, kdo nejprve musí chování počítače jako hráče vymyslet a poté naprogramovat. Počítač v této aktivitě měl být demýtizován (podle Schubertové a Schwilla je jedním z významných cílů výuky ukázat žákovi, že počítač je pouhý stroj, který vykonává příkazy [12]).

Pravidla hry jsou jednoduchá: dva hráči střídavě odebírají 1, 2 nebo 3 zápalky z hromádky 13 zápalek. Kdo odebere poslední

zápalku, prohrál. Výherní strategie v těchto hrách obecně spočívá v dosažení tzv. výherní pozice, z níž pokud hráč neudělá chybu (bude se držet optimální strategie), vždy zvítězí. První výherní pozice je 1 zápalka: pokud ji hráč dosáhne, ten druhý nemá jinou možnost, než zbývající zápalku sebrat, tudíž prohrál. Hledání posloupnosti výherních pozic vede k odhalení, že (u tohoto konkrétního zadání) mezi výherními pozicemi je vzdálenost 4 zápalek. Hra tedy nabízí jednoduchou vítěznou strategii a dané konkrétní zadání se 13 zápalkami, které jsou na začátku hry ve výherní pozici, znamená, že hráč, který nezačíná hru, vždy vyhraje. Pokud jako druhého hráče obsadíme počítač, můžeme zařídit, aby se držel vítězné strategie a stal se neporazitelný.

Základním přínosem této aktivity pro žáky bylo hledání optimální výherní strategie. Žáci byli motivováni snahou v roli hráče porazit vždy svého soupeře, a to v počítačové hře, kterou sami naprogramovali. Během jejího odhalování a pokusu o zápis do programu se objevovaly největší vášnivé diskuse za celou dobu výuky. Všichni žáci celkem rychle odhalili výherní situaci 5 zápalek, k výherní situaci 9 se propracovala asi čtvrtina žáků a asi další čtvrtina ji byla schopna akceptovat. Vítěznou strategii během 15 minut odhalili 3 žáci. Přesto jim chvíli trvalo pochopit, že existuje-li optimální strategie pro druhého hráče, nemůže první hráč nikdy vyhrát, a znovu a znovu to zkoušeli.

Teprve později, když byla strategie odhalena a více než polovinou žáků akceptována (jednak vysvětlováním mezi žáky navzájem, jednak „důkazem“ – výhrami ve vzájemných zápasech), učitel navrhl, že by žáci mohli tuto vítěznou strategii naučit počítač, aby protihráče porážel. Poté žáci zkoušeli formulovat, jak by měl počítač hrát (např. „musí odebrat vždy do čtyř“) a objevili, jak v programu zapsat tuto hodnotu výrazem. Následující implementace do programu ukázala, jak jsou si tah hráče a počítače podobné (obr. 4). Žáci dokázali objevit, že tah počítače musí být automatický, spouštěný pomocí zprávy, navíc se zpožděním simulujícím „přemýšlení počítače“. Po skončení aktivity měli nejlepší žáci zájem o to, jak modifikovat hru pro jiný počet zápalek.



Obrázek 4. Analogie mezi tahem hráče a tahem počítače s použitím výrazu z hry Zápalky

#### 4. ZÁVĚR

Projekt výuky základů programování s prvky badatelsky orientované výuky ukázal, že konstruktivistický přístup je blízký badatelsky orientované výuce, ovšem při výuce tohoto tématu má své hranice. Příčinou poměrně dlouhé doby „aklimatizace“ žáků na způsob práce při programování, než jsou schopni zodpovídat na otázky a sami bádát, nevidíme v problémech s orientací v neznámém prostředí, protože takovými situacemi žáci v jiných tématech procházeli. Příčinou byla právě jejich nová autorská role ve vztahu k počítači, které se museli nejprve naučit (např. pokud program nefungoval, žáci se zpočátku často ptali na tlačítko *Zpět*).

Ukázalo se, že na začátku žáci potřebují dostatek prostoru na úvodní zažití programátorského přístupu s velmi jednoduchými

vlastními programy ze základních příkazů, které budou sami schopni variovat neprogramátorským způsobem (tvorba grafiky, hledání námětů a témat, nikoliv vlastní tvorba programu). Dokud mají žáci tendenci pozměňovat hotové programy tímto způsobem, patrně je příliš brzy na pokračování do náročnějších partií programování. Teprve po relativně dlouhé době, kdy žáci prošli řadou situací, byli schopni o vytvořeném programu hovořit, nacházet hraniční situace, v nichž program nebude fungovat správně, navrhnout úpravy a argumentovat.

Druhá současně zkoumaná otázka, jak k takové výuce přistupují učitelé – začátečníci, ukázala, že učitelé z této výuky mají daleko větší respekt než z výuky např. kancelářských aplikací (kterou paralelně s touto výukou vedli v jiné třídě). Jejich nejistota nebyla patrná při vedení připravené výuky, ale především v situacích, kdy se na ně obrátil žák s nefungujícím programem. Detekce chyby je zdá se velice pokročilá kompetence a zařazení takovýchto úloh do vyučovaného tématu je potřebné, stejně jako zařazení aktivit v hledání žakovských chyb do didaktiky programování.

Projekt ukázal jistě možnosti, jak vést přípravu učitele na výuku programování na základní škole s tím, aby dokázal implementovat prvky badatelsky orientované výuky.

#### 5. PODĚKOVÁNÍ

Výzkum byl podporován projektem GAJU 017/2013/S.

#### 6. BIBLIOGRAFICKÉ ODKAZY

- [1] MŠMT ČR. *Strategie digitálního vzdělávání*. Verze pro připomínkování odbornou veřejností. [online] Praha: MŠMT, 2014 [cit. 2015-01-20]. Dostupné z <<http://diskuze.rvp.cz/download/file.php?id=4155>>
- [2] PAPÁČEK, M. Limity a šance zavádění badatelsky orientovaného vyučování přírodopisu a biologie v České republice. [online] In Papáček, M. (ed.) *Didaktika biologie v České republice a badatelsky orientované vyučování (DiBi 2010)*, pp. 145-162. České Budějovice: Jihočeská univerzita v Českých Budějovicích, 2010 [cit. 2014-01-30]. Dostupné z <<https://www.pf.jcu.cz/stru/katedry/bi/DiBi2010.pdf>>
- [3] BELL, T., URHAHNE, D., SCHANZE, S., & PLOETZNER, R. (2010). Collaborative inquiry learning: Models, tools, and challenges [online]. *International Journal of Science Education*, Vol. 32 No. 3, pp. 349-377 [cit. 2014-03-02]. Dostupné z <[http://peer.ccsd.cnrs.fr/docs/00/55/96/00/PDF/PEER\\_stage2\\_10.1080%252F09500690802582241.pdf](http://peer.ccsd.cnrs.fr/docs/00/55/96/00/PDF/PEER_stage2_10.1080%252F09500690802582241.pdf)>
- [4] HAJDUKOVIĆ JANDRIĆ, G., OBADOVIĆ, D. Ž., STOJANOVIĆ, M., & RANČIĆ, I. (2011). Impacts of the Implementation of the Problem-based Learning in Teaching Physics in Primary Schools. [online] *The New Educational Review*, Vol. 25 No. 3, pp. 194-204. [cit. 2014-03-03] Dostupné z <[http://www.educationalrev.us.edu.pl/vol/tner\\_3\\_2011.pdf](http://www.educationalrev.us.edu.pl/vol/tner_3_2011.pdf)>
- [5] EASTWELL, P. (2009). Letters: Inquiry Learning: Elements of Confusion and Frustration. [online]. *The American biology teacher*, Vol. 71 No. 5, pp. 263-264. [cit. 2014-03-02]. Dostupné z <<http://connection.ebscohost.com/c/letters/39755452/inquiry-learning-elements-confusion-frustration>>
- [6] RESNICK, M. et al (2009). Scratch: Programming for all [online]. *Communications of the ACM*, Vol. 52 No. 11. [cit. 2015-01-15] Dostupné z: <<http://web.media.mit.edu/~mres/papers/Scratch-CACM-final.pdf>>

- [7] KREJSA, J. *Výuka základů programování v prostředí Scratch*. Diplomová práce [online]. Jihočeská univerzita v Českých Budějovicích, Pedagogická fakulta, 2014. [cit. 2015-01-15] Vedoucí práce Jiří Vaniček. Dostupné z: <<http://theses.cz/id/b5f11x/>>.
- [8] MIKOLAJOVÁ, K. Vytváranie hier v prostredí Scratch – cesta k programovaniu na II. stupni ZŠ. In Andrejková, G. (ed.) *DidInfo 2011*. Banská Bystrica: Univerzita Mateja Bela, 2011, s. 31, 2011. ISBN 978-80-557-0142-4
- [9] CODE.ORG. *Hodina kódu. Frozen, kurs programování* [online]. Code.org, 2014 [cit. 15-01-23]. Dostupné z <<http://studio.code.org/s/frozen>>
- [10] Práce žáků ve Scratch, domácí úkol Komiks – vybrané práce. [online]. [cit. 15-01-20]. Dostupné z <<http://scratch.mit.edu/projects/32746550>>, <<http://scratch.mit.edu/projects/32376114>>
- [11] BURJÁN, V. & BURJÁNOVÁ, L. *Matematické hry*. Bratislava: Pytagoras, 1991.
- [12] SCHUBERT, S., & SCHWILL, A. *Didaktik der Informatik*. Heidelberg: Spektrum Akademischer Verlag, 2011.



# Aktivity s LEGO WeDo na 2. stupni ZŠ

Michaela Veselovská

KZVI FMFI UK Bratislava

Mlynská dolina 1

842 48 Bratislava

Slovensko

veselovska@fmph.uniba.sk

Karolína Mayerová

KZVI FMFI UK Bratislava

Mlynská dolina 1

842 48 Bratislava

Slovensko

mayerova@fmph.uniba.sk

## ABSTRAKT

V tomto článku opisujeme rôzne typy aktivít, ktoré sme realizovali s robotickou stavebnicou LEGO WeDo v piatom ročníku druhého stupňa vybranej základnej školy v Stupave. Tieto aktivity sú súčasťou návrhu koncepcie vyučovacích hodín s edukačnou robotikou, ktorú iteratívne vyvíjame v rámci nášho dizertačného výskumu. V článku popisujeme tieto aktivity z hľadiska stavby, programovania a prezentovania robotického modelu. Spomíname aj rozličné typy hodnotenia, ktoré sme pri týchto aktivitách použili. V závere článku popisujeme prínos realizovaných aktivít pri naplňaní vzdelávacích cieľov informatiky na druhom stupni ZŠ.

## ABSTRACT

In this paper, we describe the different types of activities that we conducted with robotic kit LEGO WeDo in the fifth year at lower secondary school in Stupava. These activities are part of conceptual design of lessons with educational robotics, which we have been iteratively developing within our dissertation research. In this article we describe these activities in terms of engineering, programming and presentation of robotic model. We mention even different types of assessments that we used for these activities. At the end of this paper we describe the benefits of implemented activities in meeting educational aims of informatics at lower secondary school.

## Kľúčové slová

edukačná robotika, robotické stavebnice LEGO WeDo, druhý stupeň ZŠ, rôzne typy aktivít

## Keywords

educational robotics, robotic kits LEGO WeDo, lower secondary school, different types of activities

## 1. ÚVOD

Za ostatných desať rokov sa edukačná robotika dostala do povedomia nie len vysokoškolských pedagógov, ale aj učiteľov stredných, základných a dokonca i materských škôl [1], [2], [3] a [4]. Edukačnú robotiku by sme mohli vnímať ako nástroj na rozvíjanie základných zručností pre život (zručnosti, ktoré umožňujú kognitívny a osobnostný rozvoj, zručnosti potrebné na timovú spoluprácu) prostredníctvom, ktorého si môžeme zlepšovať vyjadrovanie svojich názorov a konanie dôležitých rozhodnutí v živote [5]. S edukačnou robotikou sa môžeme stretnúť najmä v rámci rôznych robotických súťaží, na ktorých si žiaci rozvíjajú mnoho dôležitých zručností a schopností (zručnosti potrebné na riešenie problémov, programátorské zručnosti, schopnosti kolaborácie a komunikácie a pod). Podľa autorov [6] a [7] robotické súťaže predstavujú pre žiakov mimoriadny prínos nielen v oblasti akademického, ale aj osobnostného rozvoja. Do

robotických súťaží sa zapája často len niekoľko nadšencov. Ako však poskytnúť čo najväčšiemu počtu žiakov príležitosť pracovať s edukačnou robotikou a zároveň pre nich maximálne využívať jej potenciál na podporu ich poznávacieho procesu? Najprínosnejšie a zrejme aj najprirodzenejšie by bolo začleniť ju do vyučovania informatiky, prípadne s presahom do medzipredmetových projektov. Robotika by mala byť prístupná pre všetkých žiakov a nie len pre nadaných žiakov orientujúcich sa na prírodovedné či technicky zamerané vyučovacie predmety [5]. V našom výskume sme sa preto zamerali na sprístupnenie edukačnej robotiky pre všetkých žiakov 2. stupňa ZŠ, a teda na jej integráciu do predmetu informatika.

## 2. METÓDY VÝSKUMU

Opisované aktivity sú súčasťou nášho dizertačného výskumu, v ktorom sme zvolili stratégiu výskumu vývojom. V tomto procese navrhujeme a vyvíjame aktivity s edukačnou robotikou spolu s metodickými materiálmi pre učiteľov tak, aby naplňali vybrané ciele informatiky a zároveň umožňovali žiakom rozvíjať si znalosti a zručnosti pre 21. storočie. V súčasnosti sa nachádzame v závere druhej etapy výskumu vývojom, a teda väčšinu navrhnutých aktivít sme testovali druhýkrát (taktiež sme druhýkrát upravovali aj metodické materiály). V priebehu výskumu sme zvolili nasledovné metódy zberu a analýzy dát [8]: pološtruktúrovaný skupinový rozhovor, zúčastnené, neštruktúrované pozorovanie a metóda terénnych zápiskov a analýza videozáznamov. Výskum sme realizovali na Základnej škole kapitána Nálepku v Stupave. Navrhnuté aktivity s robotickou stavebnicou LEGO WeDo vyučovala vybraná učiteľka na základe vytvorených metodických materiálov počas svojej bežnej výučby informatiky. Učiteľka vyučovala dve triedy piateho ročníka, ktoré sa v priebehu vyučovania informatiky delia na viac skupín. Vo výskume participovala vždy len jedna skupina z každej triedy a v skupine bolo väčšinou 8 až 12 žiakov (chlapci aj dievčatá v rôznom pomere). V priebehu vyučovania aktivít boli na hodinách prítomní ešte dvaja výskumníci, ktorí zbierali dáta.

## 3. AKTIVITY S LEGO WeDo

Pri navrhovaní aktivít pre žiakov druhého stupňa základných škôl sme sa pridržovali princípov návrhu učebných činností podľa [9]:

- princíp sústredenia sa na kľúčové pojmy (v rámci aktivít sme sa snažili zaistiť, aby sa výučba sústredila na niekoľko kľúčových pojmov a generalizácií, nie na memorovanie množstva izolovaných faktov),
- princíp predchádzajúcich znalostí (najskôr sme zisťovali, aké majú žiaci predstavy o robotoch, či sa už stretli s robotickými stavebnicami, taktiež sme zisťovali, či má každý žiak skúsenosti so stavbou modelu z lego súčiastok, a potom sme im pripravili také činnosti, ktoré sa o tieto znalosti opierali alebo ich rekonštruovali),

- princíp kognitívnej náročnosti (niektoré učebné činnosti sme zamerali na tvorbu autentického projektu – vytváranie vlastného robotického modelu v rámci danej témy, ktorého realizácia si vyžadovala aktívne zapojenie tvorivého myslenia),
- princíp aktívneho spracovania (snažili sme sa pomôcť žiakom k hlbšiemu pochopeniu konštruovania a programovania robotického modelu prostredníctvom priamych skúseností),
- princíp rozmanitosti (výučbu sme sa snažili prispôbiť rôznym učebným štýlom, potrebám a preferenciám žiakov).

Aktivity sme ďalej vytvárali so zreteľom na revidovanú taxonómiu vzdelávacích cieľov pre edukačnú robotiku [10]. Vytvorená taxonómia vo forme tabuľky prezentovala vzdelávacie ciele a slúžila ako východisko pre naše nasledujúce iterácie vo vývoji vyučovacích hodín pre edukačnú robotiku. Taktiež nám pomohla vytvoriť rôznorodé typy úloh k objasneniu a hlbšiemu pochopeniu rôznych konštrukčných a programátorských konceptov [10].

Do väčšiny aktivít sme sa snažili zakomponovať konštrukcionistické prvky výučby [11] a [12], kde vyučujúca mala skôr úlohu podnecovať a povzbudzovať žiakov v objavovaní fungovania rôznych príkazov v softvéri. Taktiež žiakov napádala ku konfrontácii údajného fungovania príkazov (žiackeho vysvetlenia správania príkazov) a reálneho správania konkrétného robotického modelu. V niektorých aktivitách sme využívali aj inštruktívne prvky výučby.

V aktivitách, v ktorých žiaci stavali vlastné robotické modely sme sledovali princípy „tvorivej informatiky pre každého“ podľa [11]: zameranie sa na tému, kombinovanie umenia a konštrukcie, podnecovanie rozprávania príbehov a organizovanie výstav.

### 3.1 Konceptia navrhnutých hodín

V priebehu vyučovania všetkých navrhnutých hodín žiaci pracovali v dvojčlenných skupinách. Na základe našich doterajších skúseností, ale aj skúseností iných výskumníkov [13] zastávame názor, že s jednou stavebnicou by mali pracovať práve dvaja žiaci (a preto sme nedovoľovali vytváranie trojíc, pri nepárnom počte žiakov jeden žiak pracoval samostatne).

Počas druhej iterácie nášho výskumu sme vytvorili koncepciu aktivít pre edukačnú robotiku, ktorá má rozsah 12 vyučovacích hodín (jedna hodina = 45 minút). Jej rozsah sa pravdepodobne bude ešte upravovať.

1. Zoznamovanie sa s pojmom robot a objasňovanie si integrity tohto pojmu v reálnom živote
2. Stavba netriviálneho robotického modelu (Autíčko) podľa návodu a zoznamovanie sa so softvérom k robotickej stavebnici LEGO WeDo (zoznámenie sa s ikonami príkazov na ovládanie motora)
3. Stavba robotického modelu (Chalúpka na robotickej nôžke) podľa návodu a jeho programovanie podľa pracovného listu (zoznamovanie sa s ikonou príkazu cyklus)
4. Stavba (musí obsahovať motor) a programovanie (musí obsahovať ikonu príkazu cyklus) vlastného robotického modelu v rámci konkrétnej témy (Najlepší pomocník)

5. Stavba jednoduchého robotického modelu s pohybovým senzorom (Vznášadlo z budúcnosti) podľa návodu a jeho programovanie podľa pracovného listu (využitie senzoru pri programovaní)
6. Stavba (musí obsahovať motor a pohybový senzor) vlastného robotického modelu (Inteligentný sluha) a programovanie s požiadavkami (musí obsahovať ikony príkazov cyklus, zmena rýchlosti a času točenia motora, zvuky a senzor)
7. Stavba časti robotického modelu podľa obrázka (obsahuje pohybový senzor), dostavanie podľa vlastných predstáv (musí obsahovať motor) a jeho programovanie podľa pracovného listu
8. Programovanie hotového robotického modelu (Moderná korytnačka) s polohovým senzorom a vytváranie programov podľa pracovného listu
9. Stavba časti robotického modelu podľa návodu (obsahuje motor), dostavanie podľa vlastných predstáv (musí obsahovať polohový senzor) a jeho programovanie podľa pracovného listu
10. Stavba (musí obsahovať motor a aspoň jeden senzor) a programovanie vlastného robotického modelu v rámci konkrétnej témy (Mimozemský prepravný prostriedok)
11. Návrh vlastnej konštrukcie a definovanie správania robotického modelu v rámci danej témy (Rozprávková krajina)
12. Stavba a programovanie robotického modelu podľa návrhu z predošlej hodiny a jeho prezentovanie

## 3.2 Typy úloh s robotickou stavebnicou

V rámci nami navrhnutých vyučovacích hodín s edukačnou robotikou sa žiaci stretli s rôznymi typmi aktivít, ktoré sme rozdelili do nasledujúcich kategórií:

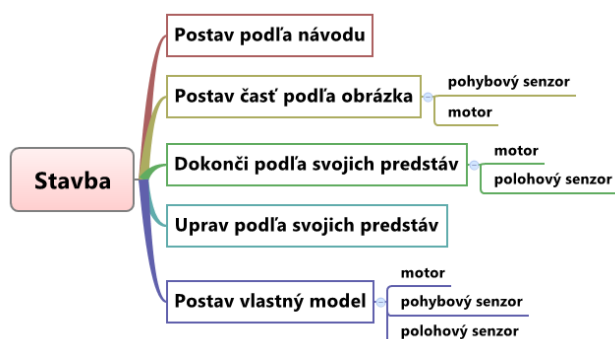
- stavba robotického modelu,
- programovanie robotického modelu,
- prezentovanie robotického modelu.

Na základe tohto rozdelenia sme spolu s vyučujúcou hodnotili celkovú prácu žiakov. Nasledujú tri podkapitoly, v ktorých popisujeme rozličné typy úloh, ako príklady na lepšie porozumenie troch spomenutých kategórií.

### 3.2.1 Stavba robotického modelu

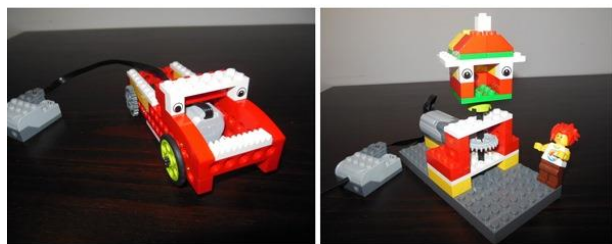
Pri stavbe robotického modelu sa žiaci stretli s piatimi typmi úloh (pozri Obrázok 1), v ktorých:

- stavali robotické modely presne podľa návodu,
- stavali časť modelu exaktne podľa obrázka,
- druhú časť modelu dostavovali podľa svojich predstáv (s určitými požiadavkami),
- postavený model (či už podľa návodu alebo obrázka) upravovali podľa svojich predstáv,
- stavali vlastný robotický model.



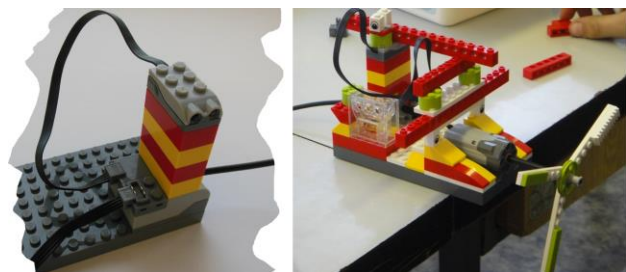
Obrázok 1. Typy úloh pri stavbe robotického modelu s povinnými komponentmi pre daný typ úlohy

V prvom type úlohy, žiaci stavali robotický model podľa návodu, v ktorom sa zoznamovali napríklad s netriviálnou stavbou modelu Autíčka a Chalúpky na robotickej nôžke (pozri Obrázok 2). V takomto type úlohy sa žiaci zoznamovali s určitými spôsobmi konkrétneho zabudovania motoru do robotického modelu. Pri stavbe Autíčka sa žiaci oboznamovali so zmenou osi otáčania motora o 90 stupňov v horizontálnom stave smerom vľavo a pri stavbe Chalúpky na robotickej nôžke sa stretli so zmenou osi otáčania motora o 90 stupňov z horizontálneho stavu na vertikálny.



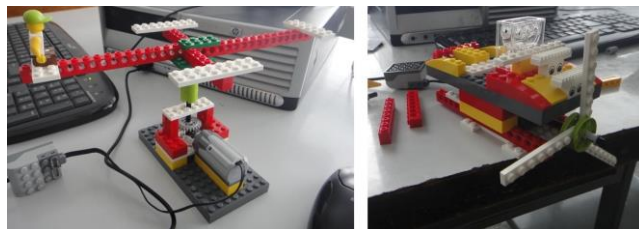
Obrázok 2. Príklady robotických modelov, ktoré žiaci stavali podľa návodov

Pri stavbe robotického modelu podľa obrázka (pozri Obrázok 3 vľavo – využitie pohybového senzora) žiaci nadobúdali skúsenosti s možným zakomponovaním pohybového senzoru alebo motora do výsledného modelu. Pri dotváraní modelu podľa vlastných predstáv zas museli do existujúceho modelu pristiaviť motor alebo polohový senzor (pozri Obrázok 3 vpravo – pristenie motora). Pri úprave robotického modelu postaveného podľa návodu alebo obrázka sme žiakom nekládli žiadne obmedzenia.



Obrázok 3. Obrázok, podľa ktorého žiaci stavali časť robotického modelu (vľavo), výsledný robotický model dotvorený podľa vlastných predstáv (vpravo)

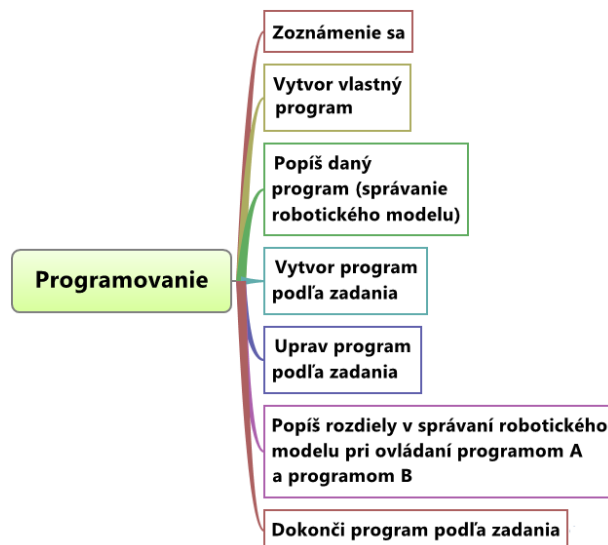
Konštrukcia vlastného robotického modelu sa tiež nezaobišla bez určitých požiadaviek na vybrané komponenty z robotickej stavebnice. Žiaci vo svojom prvom vlastnom robotickom modeli museli využiť motor (pozri Obrázok 4), pričom sa mohli inšpirovať dvomi návodmi na stavbu robotických modelov z predchádzajúcich hodín (pozri Obrázok 2). V ďalších vlastných modeloch mali žiaci použiť okrem motora aj jeden zo senzorov.



Obrázok 4. Prvé vlastné robotické modely žiakov

### 3.2.2 Programovanie robotického modelu

Z hľadiska programovania robotického modelu sa žiaci stretli so siedmimi typmi úloh (pozri Obrázok 5), v ktorých sa zoznamovali so softvérom na ovládanie robotickej stavebnice a možnosťami programovania senzorov, vytvárali vlastný program, popisovali zadaný program, vytvárali program podľa zadania, popisovali rozdiely medzi programami a dotvárali program podľa zadania.




Obrázok 5. Rôzne typy úloh pri programovaní robotického modelu

Pri type úlohy Zoznámenie sa sme zaradili frontálnu formu výučby, kde vybratá vyučujúca oboznamovala žiakov so softvérom, jeho základnými ikonami príkazov (ikony príkazov slúžiacie na spustenie, zastavenie motora, nastavenie rýchlosti a času točenia motora) a s možnosťami využívania senzorov pri programovaní robotického modelu. Pričom vyučujúca využívala projektor na názorne zobrazenie spôsobu ovládania softvéru. Pri zoznamovaní s možnosťami využívania senzorov kládla žiakom otázky ako napríklad: „Ktorá ikona v softvéri označuje pohybový senzor? S akými ikonami ju môžeme spojiť? Ktoré ikony z nich


sme už využívali a na čo slúžia?“ Takto sme sa snažili docieľiť, aby zoznamovanie sa s daným senzorom prebehlo riadeným spôsobom, ale s určitou mierou objavovania. Pri zvyšných typoch úloh žiaci nielen, že konkrétne programy vytvárali v softvéri a opisovali v pracovnom liste, ale aj každú úlohu slovné opisovali vyučujúcej, prípadne výskumníkom. Príklad úlohy, kde žiaci slovné opisovali správanie robotického modelu pri spustení daného programu do pracovného listu a úlohy, v ktorej žiaci opisovali rozdiely medzi správaním robotického modelu pri ovládaní programom A a programom B vidíme na Obrázku 6.

**1** Matúš najskôr skúša ovládať nový prostriedok pomocou starého programu:




Po spustení programu pozorne sledujte program aj dopravný prostriedok.  
Popíšte správanie dopravného prostriedku:

**2** Matúš zisťuje ako sa správa prostriedok po spustení ďalších programov. Vytvorte programy A a B na obrazovke.  
Popíšte ich a napíšte aj aké hlavné rozdiely sú medzi týmito programami A a B.



A



B

Rozdiely medzi A a B:

**Obrázok 6.** Ukážka pracovného listu znázorňujúca dva typy úloh

Podrobnejší opis jednotlivých typov úloh z hľadiska programovania robotického modelu uvádzame v [14].

### 3.2.3 Prezentovanie robotického modelu

Z hľadiska prezentovania robotického modelu sa žiaci stretli s dvomi typmi úloh (pozri Obrázok 7): buď svoju prácu (vytvorený robotický model a jeho program) opisovali vyučujúcej, alebo ju opisovali nielen vyučujúcej, ale aj celej triede. Takto žiaci získavali napríklad podnety pre ďalšiu úpravu svojej práce alebo zvýšenie motivácie. Primárne však išlo o spätnú väzbu k vytvoreným robotickým modelom a programom aj od svojich spolužiakov.



**Obrázok 7.** Kategórie aktivít pri práci s robotickým modelom s dvoma typmi úloh pri jeho prezentovaní

Prezentovanie vytvoreného vlastného robotického modelu a prezentovanie programov na jeho ovládanie (vlastných programov, ale aj programov zostavených podľa zadania) sa ukázalo ako značne podstatné. Pri vytváraní vlastného modelu mali žiaci za úlohu vysvetliť aj jeho účel, pre ktorý sa ho rozhodli postaviť v rámci zadanej témy (Najlepší pomocník, Inteligentný sluha, Postava z rozprávkovej krajiny, ...). Žiaci mali napríklad

vysvetliť, prečo je ich robotický model najlepším pomocníkom. Teda mali vysvetliť, pri akých činnostiach im pomáha a zároveň aj opísať jeho správanie – mali vysvetliť program vytvorený na jeho ovládanie. Žiaci taktiež písomne opisovali aj programy zadané v pracovných listoch, ktoré slovné vysvetľovali vyučujúcej (prípadne výskumníkom). Pri takomto „dvojtom“ výklade programu mali žiaci takmer okamžitú spätnú väzbu k ich predstave o fungovaní konkrétnych programov a mohli si tak upevniť alebo upraviť poznatok o fungovaní konkrétnych postupností príkazov v určitom programe. Vysvetľovanie nie len programov na ovládanie robotického modelu, ale aj prezentovanie celkovej práce na ňom je preto dôležitou súčasťou vyučovacích hodín s edukačnou robotikou.

## 3.3 Hodnotenie aktivít

V rámci bežného vyučovania informatiky využívala vybraná vyučujúca bodový systém. My sme však vytvorili hodnotenie, v ktorom sme sa rozhodli žiakov klasifikovať pomocou známky. Preto na základe určitej známky z nami navrhnutého hodnotenia priradila vyučujúca žiakom príslušné bodové ohodnotenie.

Pri prvých dvoch vyučovacích hodinách sme nehodnotili prácu žiakov. Hodiny slúžili hlavne k zoznamovaniu sa a ujasňovaniu si pojmu robot a k zoznamovaniu sa so softvérom a s robotickou stavebnicou, pričom mali hlavne motivačný charakter pre žiakov.

Pre zvyšné vyučovacie hodiny sme pripravili **rubriku**, ktorá poskytuje základ pre konzistentné a komplexné posúdenie stavu poznania žiakov. Ide o istý druh metodiky hodnotenia pomocou tabuľky kritérií. Žiacu prácu sme podľa tejto rubriky hodnotili z troch hľadísk:

- stavba,
- programovanie,
- prezentovanie robotického modelu.

Na jej základe žiaci získavali za každú oblasť určitý počet bodov (0 až 4 body) a ich súčtom získali výslednú známku. Táto rubrika slúži primárne na hodnotenie vlastného robotického modelu (pozri Obrázok 7). Pokiaľ žiaci stavali robotický model podľa návodu, tak sa hodnotenie zameriavalo hlavne na oblasť programovania a prezentovania robotického modelu. Rubriku budeme tiež upravovať v ďalších iteráciách výskumu.

Počet bodov	Konštrukcia	Program	Prezentácia
4	Obsahuje všetky žiadané súčasti a je pevná (nerozpadá sa pri spúšťaní programu).	Obsahuje všetky žiadané príkazy, ktoré sú spojené žiadaným spôsobom.	Obsahuje presný popis všetkých príkazov v programe.
3	Obsahuje všetky žiadané súčasti, ale rozpadá sa pri spúšťaní programu.	Neobsahuje všetky žiadané príkazy, ale zvyšné príkazy sú spojené žiadaným spôsobom.	Obsahuje presný popis len niektorých príkazov v programe.
2	Neobsahuje všetky žiadané súčasti, ale je pevná (nerozpadá sa pri spúšťaní programu).	Obsahuje všetky žiadané príkazy, ktoré však nie sú spojené žiadaným spôsobom.	Obsahuje čiastočný popis všetkých príkazov v programe.
1	Neobsahuje všetky žiadané súčasti a rozpadá sa pri spúšťaní programu.	Neobsahuje všetky žiadané príkazy a príkazy ani nie sú spojené žiadaným spôsobom.	Obsahuje čiastočný popis len niektorých príkazov v programe.
0	Neobsahuje žiadne súčasti.	Neobsahuje príkazy.	Neobsahuje popis príkazov v programe.

**Obrázok 7.** Rubrika na hodnotenie vlastného robotického modelu

V rámci hodnotenia prvého vlastného robotického modelu sme využili navyše vrstovnícke hodnotenie vo forme „tajného“ hlasovania o najzaujímavejší robotický model Najlepšieho pomocníka. Na lístok, ktorý každý žiak odovzdal, boli napísané názvy dvoch robotických modelov, ktoré sa mu najviac páčili.

### 3.4 Identifikované chyby žiakov vedúce k úprave materiálov

V tejto časti opisujeme výsledky analýzy aktivít s edukačnou robotikou z troch rozoberaných hľadísk. Sústredili sme sa hlavne na tie typy aktivít, pri ktorých sa žiaci najčastejšie dopúšťali chýb. Na základe nich budeme môcť vytvoriť výstižnejšie a pre žiakov zrozumiteľnejšie zadania úloh z hľadiska stavby, programovania aj prezentovania robotického modelu.

#### 3.4.1 Chyby pri stavbe robotického modelu

V priebehu stavby robotického modelu sa žiaci dopúšťali chýb už pri jeho stavbe podľa návodu, kde napríklad niekoľkokrát použili nesprávne LEGO kocky s chybnou dĺžkou. Takejto chybe sa pokúsime predísť tak, že *pri každej LEGO kocke označíme jej počet „bodiek“ prislúchajúcim číslom*. Pri stavbe netriviálneho modelu Autíčka žiaci mnohokrát chybné pripevňovali jeho vrhnú časť (otáčali ju v horizontálnom smere o 180 stupňov). V tomto prípade v návode *výrazne označíme spôsob pripevňovania vrchnej časti autíčka*.

Najčastejšie sa však objavovali nezrovnalosti pri stavbe vlastných robotických modelov, v ktorých mali žiaci využiť konkrétne komponenty stavebnice (pri prvom vlastnom modeli to bol motor). V tomto prípade sme žiakom umožnili využiť *i návody na stavbu robotických modelov z predchádzajúcich hodín*. Tu sa objavila neschopnosť spolupracovať v skupine pri návrhu a stavbe robotického modelu – v prípade dvoch skupín ani uľahčenie s využitím návodov nepostačovalo, pretože ani v jednej z týchto skupín sa jej členovia nemohli dohodnúť na jednom spoločnom vyhovujúcom modeli, a tak sa snažili postaviť každý svoj vlastný model. V tomto prípade sme usúdili, že by mohlo byť *prínosné pozmeniť zloženie členov jednotlivých skupín*. Počas nasledujúcej stavby vlastných robotických modelov sa táto zmena ukázala ako efektívna a spomínaný fenomén sa už nezopakoval.

#### 3.4.2 Chyby pri programovaní robotického modelu

Pri programovaní robotických modelov sa žiaci najčastejšie dopúšťali chýb pri úlohe, v ktorej mali do pracovného listu písomne opísať správanie robotického modelu pri ovládaní daným programom. V takejto úlohe žiaci buď neopísali každú ikonu v programe, alebo napríklad pri zmene rýchlosti nenapísali, na akú konkrétnu rýchlosť sa mení, ale napísali „spomali“. Jedna skupina žiakov napríklad napísala „senzor sa prispôbi rýchlosti“, zatiaľ čo mala presnejšie popísať tak, ako iná skupina „čím ide panáčik bližšie k senzoru, tým ide vrtuľa rýchlejšie...“. Pri úprave pracovných listov pre žiakov sa tak budeme snažiť *výstižnejšie a jasnejšie popísať požadované výstupy od žiakov, pričom pridáme aj ukážku programu s jeho popisom*.

Pri vytváraní programu podľa zadania žiaci tvorili nesprávne postupnosti príkazov alebo im v týchto postupnostiach chýbali niektoré príkazy. Pri úprave takejto úlohy *zvýrazníme, že pri tvorbe programu je nutné postupovať presne podľa zadania*

*a výslednú postupnosť je nutné skontrolovať každým členom skupiny*.

Pri tvorbe vlastných programov s určitými podmienkami (nutnosť využitia ikon príkazov zo zadania) žiaci nevyužívali niektoré žiadané ikony príkazov, napríklad nepoužili cyklus alebo nastavenie času otáčania motora. Žiaci síce vedeli, ktoré ikony majú použiť, avšak pokiaľ im tieto ikony nekorešpondovali s ich vlastnou predstavou o správaní ich robotického modelu, tak ich nevyužili. V tomto type úlohy *opäť zvýrazníme, ktoré ikony príkazov žiaci majú využívať, pričom žiakom predostrieme aj ukážku jedného typu programu, ktorý spĺňa všetky požiadavky*. Uvádzaním ukážok sme sa doteraz zámerne vyhýbali, pretože mohli ovplyvňovať riešenia žiakov a zamerať ich pozornosť na jeden konkrétny typ úloh, ktorým bol daný príklad.

#### 3.4.3 Chyby pri prezentovaní robotického modelu

Z hľadiska prezentovania jednotlivých úloh žiaci robili chyby najmä pri vysvetľovaní zadáných programov z pracovných listov, ale aj pri vysvetľovaní svojich vlastných programov. Podobne ako pri písomnom popise programov v pracovných listoch, tak i pri slovnom opise programov žiaci nevysvetlili buď všetky ikony programu, alebo ich vysvetlili nepresne. Niekedy to bolo spôsobené nepozornosťou, nepresnosťou, ale i nevedomosťou. Žiaci často vytvárali zaujímavé vlastné programy, avšak pri prezentácii celkovej práce žiaci len spustili program bez konkrétnych pripomienok k správaniu robotického modelu. Preto sme sa ich snažili povzbudzovať k vysvetľovaniu správania ich modelov. V rámci úpravy takejto úlohy *poskytneme žiakom ukážku prezentovania celej práce na určitom robotickom modeli v úvode konkrétnej vyučovacej hodiny ako spôsob „ideálneho“ predvedenia prezentácie robotického modelu a vysvetlenia jeho správania*.

## 4. VZDELÁVACIE CIELE INFORMATIKY

Pri navrhovaní učebných činností pre žiakov bolo našim cieľom vytvoriť také kognitívne činnosti, ktoré narábajú s pojmami, ako je hľadanie, skúmanie, zisťovanie niečoho, čo môže byť nieisté v poznávacom procese. Snažili sme sa žiakom napomáhať konkrétne poznatky vytvárať, nie ich iba odovzdávať.

Pri stavbe robotického modelu, a to prevažne v rámci stanovenej témy, si žiaci najskôr sformulovali zadaný problém – uvažovali ako bude robotický model vyzerať, na čo bude slúžiť a aké bude mať správanie. Žiaci si takýmto spôsobom mohli nadobúdať schopnosti potrebné na výskumnú prácu, teda i schopnosti potrebné na realizovanie jednoduchého projektu.

Pri určovaní správania robotického modelu, pri jeho programovaní, si žiaci nadobúdali schopnosti algoritmizovať zadaný problém a rozvíjali si svoje programátorské zručnosti. Konkrétnejšie, žiaci pri programovaní robotického modelu používali jazyk na definovanie jeho správania, teda aplikovali pravidlá konštrukcie jednoduchého edukačného programovacieho jazyka k stavebnici LEGO WeDo tak, že priamou manipuláciou zostavili ikony príkazov do postupnosti. Túto postupnosť vyhodnocovali, hľadali v nej chyby a upravovali ju. Taktiež interpretovali rozdiely medzi dvomi rôznymi postupnosťami príkazov a stanovili počet opakovaní určitej postupnosti ikon príkazov pomocou hodnoty.

Pri práci na robotickom modeli žiaci spolupracovali v skupinách a svoju prácu prezentovali vyučujúcemu i celej triede. Takto si

mohli rozvíjať nielen schopnosti kooperácie a komunikácie, ale aj svoje schopnosti verejne vystupovať a referovať o svojej práci.

Žiaci si ďalej pri práci na robotickom modeli rozvíjali aj svoju tvorivosť, logické myslenie i zodpovednosť za prácu tvorenú v skupine.

## 5. ZÁVER

V článku sme poskytli opis návrhu aktivít pre edukačnú robotiku, ktorý iteratívne vytvárame v rámci nášho dizertačného výskumu. Venovali sme sa rôznym aktivitám, ktoré sme realizovali so žiakmi 5. ročníka ZŠ s robotickou stavebnicou LEGO WeDo. Tieto aktivity sme rozdelili do troch kategórií a to stavba, programovanie a prezentovanie robotického modelu. Pričom každú kategóriu sme ďalej bližšie opisovali a analyzovali. Na základe analýzy dát z realizovaných aktivít sme identifikovali chyby a navrhli úpravy konkrétnych úloh do ďalšej iterácie nášho výskumu vývojom. Celkovo sme upravili rozsah aktivít a zloženie typov úloh v rámci jednej vyučovacej hodiny, štylizáciu i obsah konkrétnych zadání úloh a návody na stavbu robotických modelov. Veríme, že takto vytvorené aktivity pre edukačnú robotiku a metodické materiály pre učiteľov poslúžia v prvom rade na napĺňanie moderných vzdelávacích cieľov informatiky na 2. stupni ZŠ a k rozvoju viacerých dôležitých znalostí, schopností a zručností žiakov, ktoré využijú nie len vo vzdelávacom procese, ale aj pri realizácii voľnočasových aktivít a dokonca aj vo svojom budúcom povolani.

## 6. BIBLIOGRAFICKÉ ODKAZY

- [1] CAPPELLERI, David J, VITOROULIS, Nikolas. *The robotic decathlon: project-based learning labs and curriculum design for an introductory robotics course*. In *Education, IEEE Transactions on*. 2013, vol. 56, no. 1, p. 73-81.
- [2] DETSIKAS, Nikolaos, ALIMISIS, Dimitris. *Status and trends in educational robotics worldwide with special consideration of educational experiences from Greek schools*. In *5th International Conference on Informatics in Schools: Situation, Evolution and Perspectives*. 2011, p. 1-12. ISBN 978-3-642-24721-7.
- [3] BENITTI, Fabiane B.V. *Exploring the educational potential of robotics in schools: A systematic review*. In *Computers & Education*. 2012, vol. 58, p. 978-988. ISSN 0360-1315.
- [4] BERS, Marina U. et al. *Computational thinking and tinkering: Exploration of an early childhood robotics curriculum*. In *Computers & Education*. 2014, vol. 72, p. 145-157. ISBN 978-1-4129-5052-7.
- [5] ALIMISIS, Dimitris. *Educational robotics: Open questions and new challenges*. In *Themes in Science and Technology Education*. 2013, vol. 6, no. 1, p. 63-71.
- [6] SKLAR, Elizabeth I., JOHNSON, Jeffrey H., LUND, Henrik H. *Children learning from team robotics: Robocup junior 2000* [online]. [cit. 2014-1-23]. Dostupné na: <<http://demo.cs.brandeis.edu/papers/rcj2000-full.pdf>>
- [7] GURA, Mark. *Getting Started with LEGO Robotics: A Guide for K-12 Educators*. United States of America: International Society for Technology in Education, 2011. ISBN 978-1-56484-298-5.
- [8] ŠVARÍČEK, Roman et al. *Kvalitativní výzkum v pedagogických vědách*. Praha: Portál, 2007. ISBN 978-80-7367-313-0.
- [9] PASCH, Marvin et al. *Od vzdělávacího programu k vyučovací hodině*. Praha: Portál, 2005. 416 s. ISBN 80-7367-054-2.
- [10] KALAŠ, Ivan, MAYEROVÁ, Karolína, VESELOVSKÁ, Michaela. *Developing a learning taxonomy for educational robotics*. In *Journal of Technology and Information Education*. 2014, vol. 6, no. 1, p. 30-44. ISSN 1803-537X.
- [11] RUSK, Natalie et al., *New Pathways into Robotics: Strategies for Broadening Participation*. In *Journal of Science Education and Technology*, 2008. vol. 17, no. 1, p. 59-69.
- [12] PAPERT, Seymour. *The Eight Big Ideas of the Constructionist Learning Laboratory*. Unpublished
- [13] KABÁTOVÁ, Martina, PEKÁROVÁ, Janka. *Edukačná robotika pre prvý stupeň a budúci učitelia*. In *DidInfo 2008*. Banská Bystrica: Univerzita Mateja Bela. 2010, p. 26. ISBN 978-80-8083-952-9.
- [14] VESELOVSKÁ, Michaela, MAYEROVÁ, Karolína. *Využitie pohybového senzoru pri integrácii edukačnej robotiky do informatiky na 2. stupni ZŠ*. Rukopis. Publikácia prijatá na konferenciu *DidactIG 2015*. Liberec: Katedra aplikované matematiky, Fakulta prírodovedně-humanitní a pedagogická, Technická univerzita v Liberci, 2015.



# Čo sú Open Badges a ako ich použiť?

Dana Šuníková  
KZVI FMFI UK v Bratislave  
Mlynská dolina  
842 48 Bratislava  
Slovenská republika  
dana.sunikova@fmph.uniba.sk

Zuzana Kubincová  
KZVI FMFI UK v Bratislave  
Mlynská dolina  
842 48 Bratislava  
Slovenská republika  
zuzana.kubincova@fmph.uniba.sk

## ABSTRAKT

Motivácia študentov je dôležitým aspektom vzdelávania, ktorým sa musí zaoberať každý učiteľ. Webové technológie a sociálne médiá nám ponúkajú viacero nástrojov, zapojenie ktorých do vyučovania môže na študentov pôsobiť motivujúco. V tomto článku zoznamujeme čitateľa s používaním odznakov ako doplnkovej formy hodnotenia, predstavujeme koncept Open Badges a možnosti využitia týchto digitálnych odznakov v Moodle.

## ABSTRACT

Students' motivation is an important educational aspect which every teacher has to face. Web technologies and social media offer several tools which could have motivational effect on students when integrated into education. In this paper, we familiarize the reader with badges as an additional form of the assessment. We introduce the Open Badges concept as well as the possibility to use these digital badges in Moodle.

## Kľúčové slová

Open Badges, vzdelávanie, motivácia, odznaky, hodnotenie, Moodle

## Keywords

Open Badges, education, motivation, badges, assessment, Moodle

## 1. ÚVOD

Jedným z kľúčových faktorov ovplyvňujúcich úspešnosť výchovno-vzdelávacieho procesu na všetkých stupňoch i vo všetkých typoch vzdelávania je nepochybne motivácia. Má vplyv na záujem učiaceho sa o predmet štúdia, jeho aktivitu i sústredenosť v procese učenia sa i celkový prístup k študijným povinnostiam. Rozdiely v úrovni motivácie dvoch podobne nadaných jedincov môžu spôsobiť, že ani v rovnakých vzdelávacích podmienkach nepodajú podobné výkony [15].

V nižšom školskom veku prevažuje tzv. vonkajšia motivácia – učenie je motivované napr. odmenou alebo trestom, snahou uspokojiť učiteľa či rodičov a pod. Pre učenie je však cennejšia vnútorná motivácia, ktorá pôsobí dlhodobejšie [9]. Učiaci sa je v takomto prípade motivovaný vlastnou zvedavosťou a záujmom o predmet, učí sa pre svoje vlastné uspokojenie.

Motivácia však nie je nemenný faktor – je ovplyvniteľná a môže sa vyvíjať. V prostredí organizovaného vzdelávania hrá pri motivovaní žiakov dôležitú úlohu učiteľ. Zapájaním vhodných motivačných princípov môže zvyšovať vonkajšiu ale i vnútornú motiváciu svojich žiakov. Medzi významné metódy rozvíjania motivácie patrí okrem pozitívneho hodnotenia napr. aj vyučovanie hrou, kde ako motivačný faktor pôsobí súťaživosť, radosť z hry, uvoľnená atmosféra a pod. [15].

V posledných rokoch je v súvislosti s motivovaním k učeniu pomerne často skloňovaným výrazom „gamifikácia“ (z angl. gamification) definovaná ako využitie herných mechanizmov, dynamiky a konštrukcií v mimoherných kontextoch [3]. Zjednodušene povedané, ide o zapájanie herných prvkov do vyučovania. V podstate je to spojenie dvoch vyššie spomenutých metód rozvoja motivácie realizované rôznymi spôsobmi, napr. vytváranie rôznych rebríčkov, súťaženie viacerých skupín v rámci triedy, pozitívne hodnotenie prostredníctvom alternatívneho odmeňovania čiastkových úspechov (inak, ako známku) a pod.

Jednou z možností zavedenia herných prvkov do vyučovania je použitie odznakov a iných ocenení ako alternatívnej, resp. doplnkovej formy hodnotenia. Súčasným trendom v tejto oblasti je využívanie digitálnych odznakov [1, 2, 4, 11, 12]. Vzhľadom k stále širšiemu uplatneniu digitálnych technológií, internetu a sociálnych médií v škole i v bežnom živote študentov je pre nich digitálna forma odznakov prirodzeným konceptom.

V tomto príspevku sa budeme zaoberať charakteristikou digitálnych odznakov, príkladmi ich použitia vo vyučovaní, predstavíme koncepciu *Open badges* a ukážeme možnosti ich použitia v Moodle.

## 2. ODZNAKY

Podľa Lobkowitza [8] siaha história odznakov do staroveku. Rímski vojaci ich nosili ako vyznamenanie, prípadne mohli znamenať hodnotu alebo príslušnosť k oddielu [16]. V časoch stredovekých rytierov boli odznaky heraldickými symbolmi, ktoré sa nosili ako šperky a neskôr ako symbol ukončenia púte alebo znak politickej príslušnosti [4].

Viaceri autori [1, 2, 4, 11 a ďalší] poukazujú na posun v používaní odznakov v prostredí skautskej komunity, kde začali slúžiť ako znak dosiahnutého úspechu v nejakej presne určenej oblasti, napr. získanie nejakej konkrétnej zručnosti alebo vedomosti na vopred definovanej úrovni. Neskôr, s nástupom video-hier a počítačových hier sa odznakmi začalo vyjadrovať dosiahnutie nejakej úrovne hry, prekonanie prekážky, zdolanie výzvy. Rozšírenie internetu umožnilo získané odznaky aj prezentovať navonok iným užívateľom, vytvárať rebríčky a pod.

Vyššie spomínaným spôsobom sú odznaky chápané a používané aj v súčasnom kontexte sociálnych médií, ako symbol splnenia úlohy, dosiahnutia úspechu, resp. získania skúsenosti, pričom takéto odznaky môže udeliť nejaká inštitúcia, organizácia, skupina alebo aj jednotlivec. Začali sa preto používať ako motivácia aj v širšom kontexte, dokonca aj v oblastiach, ktoré primárne nie sú spojené so získavaním nových vedomostí a zručností. Dobrým príkladom je ich použitie v elektronickom bankovníctve. Napr. majiteľovi účtu v mBank sú za jeho rôzne úkony spojené s nastavovaním a používaním jeho účtu v rámci internet-bankingu, s používaním platobnej karty, so sporením a pod. pridelené rôzne odznaky (viď obr. 1). Banka chce zrejme týmto



Obrázok 1. Odznaky v internet-bankingu v mBank.

spôsobom motivovať svojich klientov k častejším finančným transakciám na ich účte.

Pre nás je však podstatné, že vyššie spomínaný model udeľovania odznakov v prostredí skautingu, prípadne v počítačových hrách, dobre korešponduje s odmeňovaním v prostredí vzdelávania.

### 3. ODZNAKY VO VÝUČBE

Uplatnenie odznakov vo vzdelávacích aktivitách v škole je pomerne priamočiare. Študenti dostávajú od učiteľa za vyriešené úlohy, dosiahnuté vedomosti, zručnosti alebo iné úspechy symbolickú cenu podobne, ako v súčasnosti bežne dostávajú známku alebo certifikát. Symbol ocenenia – odznak – potom študenti môžu sprístupniť aj iným, aby im tak dali vedieť o svojich vedomostiach alebo zručnostiach.

Veľmi jednoduchou implementáciou odznakov môže byť triedna nástenka so zoznamom študentov, na ktorú učiteľ pripína udelené odznaky – najlepšie obrázky vystihujúce danú aktivitu. Tento jednoduchý prístup má však aj značné nevýhody, napr. odznaky na nástenke môže hocikto premiestniť alebo ich odtiaľ odstrániť, odznaky neodrážajú presne aktivity, za ktoré boli udelené a ak má byť nástenka dostatočne výpovedná, treba ju dopĺňať o zadania, čím sa pri ich väčšom množstve môže stať neprehľadnou.

V súčasnosti, keď sú digitálne technológie neodmysliteľnou súčasťou nášho každodenného tak profesionálneho ako aj osobného života a presadzujú sa stále viac aj vo vyučovaní, sa preferovanou formou odznakov používaných v rámci hodnotenia stávajú digitálne odznaky. Digitálny odznak je vizuálna reprezentácia výkonu študenta certifikujúca jeho zručnosti a schopnosti, ktorá je dostupná on-line. V jednoduchšej forme to môže byť napr. iba obrázok, ale v poslednej dobe sa stále viac používajú digitálne odznaky, ktoré obsahujú aj metadáta (o.i. aj linky-odkazy), ktoré môžu poskytnúť detailné vysvetlenie toho, čo sa držiteľ odznaku naučil, vrátane popisu zadania a odkazu na výsledok jeho práce.

Konkrétna metodika využitia odznakov vo vyučovaní môže byť rôzna. Odznaky

- možno použiť ako alternatívnu alebo doplnkovú formu hodnotenia;
- možno použiť na hodnotenie všetkých alebo len vybraných aktivít na hodine;
- možno udeľovať priamo za výkony vo vzdelávacích aktivitách a/alebo za iné ako vzdelávacie aktivity (napr. pomoc spolužiakovi pri riešení úlohy);
- možno udeľovať za aktivity v rámci alebo nad rámec študijných povinností;

- možno udeľovať automaticky po splnení nejakej aktivity alebo na základe schválenia výsledku danej aktivity učiteľom;
- môže udeľovať len učiteľ alebo si ich (alebo nejakú časť z nich) môžu udeľovať žiaci navzájom medzi sebou;
- možno udeľovať jednotlivcom alebo skupinám;
- možno udeľovať po jednom alebo môžu byť pripravené v sériách pokrývajúcich po častiach celú nejakú oblasť, atď.

Predchádzajúce alternatívy použitia odznakov môžu priniesť rozdielne výsledky. Učiteľ by sa mal pri ich aplikovaní rozhodnúť na základe cieľov, ktoré chce v triede dosiahnuť a tiež ich prispôbiť špecifikám tej ktorej študijnej skupiny.

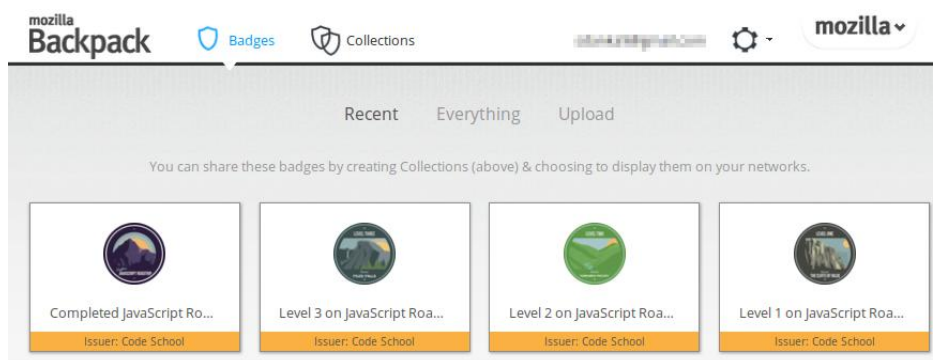
Využívanie digitálnych odznakov v škole môže mať pozitívny vplyv na vyučovací proces vo viacerých ohľadoch:

**Zvyšovanie motivácie pre učenie sa.** V prípade používania odznakov pôsobí na žiakov motivačne viacero faktorov. Jedným z princípov zvyšovania motivácie je *bezprostredná spätná väzba* po podanom výkone. Odznaky udeľované priebežne ju môžu zabezpečiť.

Zbierať odznaky a vidieť ich zobrazené na svojom vzdelávacom profile je pre žiakov nielen hravé a zábavné, ale vnáša do učenia sa aj *prvok súťaživosti*, ktorý motivuje žiakov zbierať ďalšie odznaky. To však dosiahnu len ďalším štúdiom a plnením študijných povinností, takže zbieranie odznakov ich nakoniec vedie k získavaniu ďalších vedomostí a zručností.

V prípade používania sérií digitálnych odznakov, môže prostredníctvom svojich metadát každý z odznakov ukazovať na ďalší krok, ktorý treba spraviť pre získanie nasledujúceho odznaku a tým lepšie objasniť a zjednodušiť žiakovi *postupné získavanie poznatkov* v príslušnej oblasti pokrytej danou sériou odznakov [4].

**Sledovanie stavu a pokroku v učení sa.** Priebežné udeľovanie odznakov poskytuje učiacim sa nielen bezprostrednú spätnú väzbu, ale dáva im aj ich okoliu i možnosť *priamo vizuálne sledovať ich progres* v učení sa. Digitálny odznak môže prostredníctvom svojich metadát nielen popisovať zadanie, ale cez hyperlinku odkazovať priamo na hodnotenú prácu, za ktorú bol pridelený, prípadne nepriamo na kolekciu iných údajov potrebných na vyhodnotenie výsledku úlohy, za ktorú bol pridelený. Toto umožňuje podchytiť a oceniť aj také zručnosti, ktoré bežne nebývajú predmetom hodnotenia, napr. sociálne zručnosti, kolaboratívnu prácu a pod. Možno k nim potom priradiť digitálny odznak, ktorý prostredníctvom odkazu na video zobrazí záznam príslušných zručností, resp. aktivít [4].



Obrázok 2. Odznaky účastníka kurzu na portáli Code School zobrazené v úložisku Mozilla-Backpack.

Digitálne odznaky môžu taktiež podporovať budovanie identity a rozvoj vzťahov v sociálnych sieťach. Profil, v ktorom sa zobrazujú aj digitálne odznaky, umožňuje budovať reputáciu jeho majiteľa a pomôcť ostatným nájsť profily s podobnými záujmami alebo zručnosťami.

**Dôkaz dosiahnutých úspechov.** Digitálne odznaky so svojimi metadátami a linkami na výsledky práce študentov poskytujú jednoduchú a transparentnú možnosť prezentovania získaných vedomostí a zručností otvorenému publiku a pri vhodnej implementácii odznakov aj možnosť hodnotenia práce týmto publikom.

Výhodou takéhoto prístupu je dostupnosť študijných výsledkov spolu s popismi zadania aj konkrétnymi výstupmi prípadným záujemcom, ktorými môžu byť napr. škola na vyššom stupni vzdelávania, budúci zamestnávateľ, profesionálna alebo záujmová sociálna sieť a pod.

Ako ukazujú viaceré štúdie [napr. 1, 4, 12], používanie odznakov vo vyučovaní motivuje študentov pokračovať v učení sa, dokonca aj v prípade, keď je vzdelávacia aktivita v konflikte s inými prioritami študenta. Aj napriek tomu, že odznaky sú vonkajšou odmenou, ukazuje sa, že môžu slúžiť ako vnútorná motivácia [11].

## 4. OPEN BADGES

Cieľom iniciatívy Open Badges od Mozilly je nahradenie životopisov odznakmi, ktoré reprezentujú schopnosti používateľa a ich obsah je verifikovateľný – je možné zistiť, za čo bol odznak udelený a kto ho udelil. Odznaky je možné udeľovať ako v akreditovaných inštitúciách, tak aj v neformálnom vzdelávaní, a preto ich môžu získať aj ľudia, ktorí inak na vzdelanie nemajú peniaze (vysoké školy sú v mnohých štátoch platené a veľmi drahé). Na Open Badges je výnimočné najmä to, že je to otvorená platforma, takže každý môže odznaky vydávať, zobrazovať a získať. Vďaka tomuto – a tomu, že sú čoraz populárnejšie a vydáva ich stále viac inštitúcií – je reálne, že časom bude väčšina digitálnych odznakov práve typu Open Badges a odznaky sa budú používať ako odmeny stále viac.

Odznaky Open Badges fungujú nasledovne. Vydavateľ odznakov, rozhodne, aké kritériá musí splniť človek, ktorý chce odznak získať. Potom odznak vytvorí a online ponúkne záujemcom. Používateľ sa môže o odznaky uchádzať a keď splní kritériá, odznak mu bude udelený. Toto je možné spraviť ako manuálne tak automaticky, napríklad za splnenie programátorských úloh na webstránke.

Používateľ zbiera odznaky za rôzne vedomosti, schopnosti, prípadne správanie. Tieto si vkladá do svojho digitálneho úložiska, nazývaného “Backpack” (Obr.2). Tu ich môže zoskupovať do kolekcií, pre ktoré vie nastaviť, aby boli verejné, a potom ich zobraziť napríklad na sociálnych sieťach. Na odznaky je možné kliknúť, a tak zobraziť detailné informácie o kritériách, ktoré bolo potrebné splniť na získanie odznaku. Tieto kritériá a ďalšie informácie, ako napríklad vydavateľ, prípadne konkrétne riešenie úlohy, ak bol odznak získaný za vypracovanie zadania, sú kódované v metadátach odznaku. Ak chce niekto zistiť informácie o držiteľovi odznaku, jeho práci a schopnostiach, prezrie si detailnejšie tieto metadáta poskytnuté vydavateľom [5].

V odznakoch momentálne nie je možné vyhľadávať. Pokiaľ sa však z projektu vyvinie ekosystém, ako jeho autori chcú, je veľmi pravdepodobné, že odznaky časom indexované budú. Potom bude napríklad možné vyhľadávať ľudí podľa množiny zručností, ktoré ovládajú [5].

Systém odznakov Open Badges stojí na niekoľkých základných typoch používateľov:

- Žiaci, študenti a ktokoľvek, kto chce odznaky získať a následne ich ukázať svetu. O odznaky sa treba uchádzať a následne splniť kritériá na ich získanie. Odznak je viazaný ku konkrétnej e-mailovej adrese, ktorá je identifikáciou príjemcu v systéme.
- Vydavatelia odznakov – rôzne inštitúcie formálneho aj neformálneho vzdelávania alebo ktokoľvek, kto vytvorí vlastný odznak.
- Zobrazovatelia odznakov – dávajú možnosť študentom zobraziť odznaky, ktoré označili ako verejné. Zobrazovatelia dokážu overiť odznaky a zobraziť ich na webstránke, v aplikácii, widgete alebo na sociálnej sieti.
- Tí, ktorí si odznaky prezierajú. Môžu to byť napríklad potenciálni zamestnávateľia, spolužiaci, škola, do ktorej sa študent hlási a pod.

Dôvodom na založenie iniciatívy Open Badges bolo niekoľko. Okrem už spomínanej ceny vzdelania je to celkový pohľad na vzdelávanie v dnešnom svete, ktorý sa mení. Učenie sa už nie je len o sedení v škole, ale rozširuje sa do rôznych kontextov, zážitkov a interakcií. Učenie je v dnešnom svete široké, sociálne, neformálne, kolektívne, kreatívne a celoživotné. Žiaci sú aktívni účastníci, nie len konzumenti. Vzdelávacie prostredie už nie je len jediná trieda alebo webstránka, ale pozostáva z veľa priestorov, ktoré sa nachádzajú v širšom, zosieťovanom, distribuovanom a



rozšíriteľnom prostredí. V takomto prostredí sa žiakom núkajú rôzne spôsoby, ako nadobúdať kompetencie a zdokonaľovať svoje zručnosti [13].

Za iniciatívu Open Badges hovorí okrem iného argument, že titul získaný v nejakom študijnom programe ešte nevypovedá o konkrétnych vedomostiach. Napríklad to, že niekto získal titul bakalára v informatike nemusí nutne znamenať, že vie pracovať s JavaScriptom, je dobrý projektový manažér, pracoval na open source projekte alebo dokáže dobre pracovať vo virtuálnom tíme [14]. Odznaky typu Open Badges sa naproti tomu získavajú za konkrétne zručnosti a vypovedajú presne o schopnostiach používateľa, a to ako o “hard skills”, tak o “soft skills”, teda aj o vedomostiach a naučených schopnostiach, aj o vlastnostiach, vďaka ktorým zamestnanci vedú efektívne a harmonicky vychádzať s inými ľuďmi. Ďalšia z výhod Open Badges je, že efektívne implementuje myšlienku pokus-omyl. Pri získavaní odznaku sa žiaci učia novým veciam a snažia sa, kým ho nezískajú.

Ozývajú sa aj kritické hlasy. Ich hlavnými argumentmi je, že vzdelanie zadarmo je stále zadarmo práve preto, že nemá žiadnu komerčnú hodnotu. Keď vďaka odznakom bude mať komerčnú hodnotu, prestane byť zadarmo. Ďalšia dôležitá vec je motivácia ľudí vzdelávať sa, pracovať na sebe, športovať. Vo svete, kde za toto nedostanú odmenu v podobe odznaku, to robia kvôli vedomostiam, dobrému pocitu, zábave. Keď bude za všetko možné získať odznak, hrozí riziko, že takéto aktivity budú používatelia robiť iba kvôli odznakom [10].

Na internete sa dajú získať rôzne druhy odznakov za rôzne aktivity, online aj offline. Open Badges odznaky je možné získať napríklad za:

- **absolvovanie online kurzu** alebo semestra na online vzdelávacej inštitúcii, napríklad na MLHS Online Academy<sup>1</sup>, kde je potrebné vytvoriť svoj študijný plán, zúčastňovať sa týždenných stretnutí a na konci štúdia prezentovať nové poznatky. Peaceful Playgrounds<sup>2</sup> ponúka online kurzy zamerané na prácu s deťmi hlavne mimo vyučovacieho času, ale ponúka napríklad aj kurz pre učiteľov na prevenciu šikany. Za absolvovanie kurzov získa účastník certifikát a odznak typu Open Badge. Tieto kurzy sú však platené.
- **účasť na konferencii alebo workshope.** MCN podporuje používanie digitálnych technológií v kultúrnych organizáciách. Okrem iného zorganizovali sériu workshopov<sup>3</sup>, za účasť na ktorých bolo možné získať rôzne druhy odznakov.
- **odovzdané zadanie ohodnotené ostatnými študentmi.** Peer 2 Peer University<sup>4</sup> je online vzdelávacia inštitúcia, ktorá sa zameriava na formovanie komunity vzdelávajúcich sa. Na tomto portáli je mnoho odznakov, z ktorých niektoré si udeľujú študenti navzájom. Občas je treba, aby sa viacerí študenti vyjadrili, že zadanie bolo splnené. Veľa takýchto zadaní sa “odovzdáva” formou blogu, napríklad treba vytvoriť HTML kód,

napísať o ňom blogový príspevok a čakať, kým ho ostatní ohodnotia.

- **odovzdané zadanie ohodnotené lektormi kurzu.** Na úspešné absolvovanie kurzu Introduction to Openness in Education<sup>5</sup> treba študovať materiály, hľadať nové zdroje, vytvárať učebné pomôcky a tie zdieľať s ostatnými. Tiež je potrebné sa zapojiť do diskusie cez Twitter. V kurze sa dajú získať tri rôzne odznaky. Jeden je za splnenie všetkých požiadaviek, druhý za nadštandardnú prácu aspoň v troch častiach kurzu a tretí za diskusiu, v ktorej študent najprv obhajuje otvorenosť vo vyučovaní pred nadriadeným a následne o nej publikuje článok na blogu. V kurze Badges: New Currency for Professional Credentials<sup>6</sup> účastníci postupne vytvárajú ekosystém Open Badges. Jednotlivé vypracované zadania odovzdávajú na kontrolu lektorom kurzu a dostávajú za ne odznaky Open Badges. Riešenia sa dajú prerábať a odovzdávať opakovane, pokiaľ nie sú účastníci s hodnotením spokojní. Pracovať je možné jednotlivo aj v tímoch.
- **odovzdané zadanie ohodnotené automaticky.** Portál BuzzMath<sup>7</sup> slúži na precvičovanie matematiky na základnej a strednej škole. Dá sa na ňom zaregistrovať ako študent, učiteľ alebo rodič. Učiteľ vytvorí kurz, do ktorého pozve študentov. Študenti vedú riešiť rôzne úlohy zoskupené do sérií. Za každú správne vyriešenú sériu úloh získa študent hviezdičku. Pokiaľ je študent zapísaný v nejakom kurze a vyriešil všetky série úloh týkajúce sa daného učiva, otvorí sa mu výzva, ktorú keď splní, dostane odznak. Týchto odznakov je na stránke momentálne jedenásť (Obr. 3 (a)). Portál Code School<sup>8</sup>, ponúkajúci vzdelávanie v informatike, funguje na podobnom princípe, je tu však iba rola študenta. Ten si vyberie, ktorou „cestou“ sa chce vydať, teda či sa ide učiť JavaScript, Ruby, prípadne inú technológiu. Každá táto cesta obsahuje niekoľko kurzov a v každom kurze je niekoľko úrovní. Odznaky sa dajú získať za úroveň aj kurzy.
- **„process knowledge“**, teda za kreativitu, kolaboráciu, kritické myslenie a podobne. Na už spomínanom portáli BuzzMath sa nachádzajú aj odznaky, ktoré udeľuje učiteľ za pozitívny prístup a správanie, vďaka ktorému dosiahnu študenti v matematike úspech. Momentálne ich je dostupných trinásť (Obr. 3 (b)). Coderbits<sup>9</sup> je webová stránka, kde si môže každý automaticky vytvoriť svoje portfólio prelinkovaním iných webových stránok, ktoré používa. Udeľujú sa tu odznaky za aktivitu na prelinkovaných stránkach, napríklad za ambície, komunikáciu, vplyv, iniciatívu, vzdelanie, produktivitu, kvalitu, sebarozvoj, zručnosti alebo tímovú prácu.

1 <http://blogs.mtlakes.org/onlineacademy/>

2 <http://www.peacefulplaygrounds.com/>

3 <http://mcnpro.org/>

4 <https://p2pu.org/en/>

5 <https://learn.canvas.net/courses/4>

6 [https://www.coursesites.com/webapps/Bb-sites-course-creation-BBLEARN/courseHomepage.htmlx?course\\_id=\\_264998\\_1](https://www.coursesites.com/webapps/Bb-sites-course-creation-BBLEARN/courseHomepage.htmlx?course_id=_264998_1)

7 <https://www.buzzmath.com/>

8 <https://www.codeschool.com/>

9 <https://coderbits.com/>



Obrázok 3. Odkazy z portálu Buzzmath. (a) za vyriešené úlohy; (b) za „process knowledge“

- **iné aktivity, online**, napríklad na stránke Physiopedia<sup>10</sup> sa dá veľa odznakov získať za aktívnu účasť na vytváraní a zveľaďovaní obsahu tohto portálu. Odkazy udeľujú administrátori.
- **iné aktivity, offline**. Michigan State University<sup>11</sup> aj Seton Hall University<sup>12</sup> ponúkajú odznaky za účasť na rôznych podujatiach. Študent potvrdí účasť oskenovaním QR kódu. Do programu Books@Work<sup>13</sup> sa prihlasujú skupiny zamestnancov jednej firmy. Lektori im priebežne vyberajú knihy na čítanie, a to rôzne žánre, vrátane beletrie, a každý týždeň sa zapájajú do diskusie o čítanej knihe. Odnak sa dá získať za aspoň 3 mesiace v programe, pokiaľ sa účastník zúčastnil aspoň na desiatich stretnutiach z dvanástich a odpovedal na tri krátke otázky o knihách a jeho dojmach z nich. V SAPO Campus<sup>14</sup> sa odznaky dajú použiť rôznymi spôsobmi – na identifikovanie profilu osoby, zvýraznenie ľudí, ktorí sa zúčastnili na aktivite organizovanej inštitúciou, na uznanie najlepšej práce v triede alebo dokonca pre učiteľa, ktorý vyhral školskú karaoke súťaž.

Z uvedeného je zrejme, že odznaky Open Badges majú rôznorodé využitie a postupne začínajú žiť vlastným životom. Viaceré príklady ukazovali možnosti ich udeľovania buď priamo v škole, alebo za doplnkové vzdelávacie aktivity. Keďže však všetky uvedené portály používajú výhradne angličtinu, nedajú sa v našom prostredí využiť priamo, učitelia sa na nich môžu len inšpirovať. V nasledujúcej časti uvádzame príklad lokalizovaného prostredia, v ktorom taktiež možno využiť odznaky Open Badges.

## 5. OPEN BADGES V MOODLE

Moodle je vzdelávacia platforma navrhnutá tak, aby poskytovala pedagógom, administrátorom a študentom robustný, bezpečný a integrovaný systém na vytvorenie personalizovaného vzdelávacieho prostredia [6]. V Moodle administrátor vytvorí kurz, v ktorom pedagógovia pridávajú pre študentov rôzne

aktivity. Medzi aktivity patria prednášky, ankety, diskusné fóra, rôzne zadania a mnohé iné.

Moodle ponúka aj možnosť ako ukázať prospech a odmeniť úspech, a tou sú práve odznaky. Odkazy môžu byť udelené na základe rôznych kritérií a sú plne kompatibilné s Open Badges od Mozilly. Odkazy vytvorené v Moodle sa dajú zobrazíť v používateľskom profile, ale aj poslať do digitálneho úložiska „Backpack“. Verejné kolekcie odznakov z úložiska je možné zobrazíť v profile používateľa v Moodle [7]. Existujú dva typy odznakov – tie, ktoré sa vzťahujú k aktivitám na stránke a tie, ktoré sa vzťahujú k aktivitám v konkrétnych kurzoch. My sa budeme ďalej venovať druhému typu odznakov.

Odkazy môže pridávať do kurzu aj učiteľ (Obr. 4). Údaje, ktoré treba odznaku vyplniť, sú názov, popis, obrázok, ďalej meno a prípadne e-mailová adresa vydavateľa. Odnaku udeľovanému za zručnosti, ktoré treba obnovovať, sa dá nastaviť doba jeho platnosti.

Odkazy v Moodle sa delia do troch kategórií podľa kritérií, ktoré treba splniť na ich získanie.

- **Odkazy, ktoré udeľuje učiteľ**. Pri nich treba nastaviť, ktorí používatelia ich vedia udeliť podľa role, ktorú v systéme zastávajú. Na výber sú možnosti *Manager*, *Teacher*, *Non-editing teacher*. V systéme sa nedá vytvoriť odznak, ktorí by si mohli študenti udeľovať navzájom.
- **Odkazy za ukončenie kurzu**. Tu je možné nastaviť najhoršiu známku, pri ktorej ešte odznak bude udelený a dátum, do ktorého musí byť kurz hotový.
- **Odkazy za ukončenie aktivity**. Pri tomto type odznaku je treba vybrať aktivity, za ktorých ukončenie získajú študenti odznak a taktiež treba aktivite nastaviť, kedy je uznaná za splnenú.

Odkazy po ich sprístupnení študentom nie je možné ďalej upravovať. Jediná výnimka nastáva, ak ich znova zneprístupníme a odznak nebol ešte nikomu udelený. Pokiaľ ho získal aspoň jeden študent, odznak už nijako nezmeníme.

Medzi zaujímavé možnosti patrí ešte vymazanie odznaku. Dá sa vybrať z dvoch možností, a to či chceme, aby sa daný odznak vymazal každému používateľovi, alebo nie. Pokiaľ chceme odznak vymazať, ale už pridelené odznaky tohto typu













10 [http://www.physio-pedia.com/Main\\_Page](http://www.physio-pedia.com/Main_Page)

11 <http://badges.msu.edu/>

12 <http://tltc.shu.edu/badges/>

13 <http://www.booksatwork.org/>

14 <http://campus.sapo.pt/blog/oficialteamblog>

Meno	Badge status	Criteria	Recipients	Akcie
 Aktivita	Available to users	• Awarded by: Teacher	0	  
 Domáca úloha	Not available to users	• Complete: "Assign - Domáca úloha 03"	0	  
 Pomoc	Available to users	• Awarded by: Teacher	0	  

Obrázok 4. Tabuľka odznakov pripravených učiteľom v Moodle

používateľom zachovať, budú môcť s predtým prideleným odznakom naďalej manipulovať, avšak novú inšanciu tohto odznaku už nebude možné získať. V prípade, že chceme odznak používateľom vymazať, nebude sa im už zobrazovať v profile a stratia k nemu úplne prístup. Ak ho predtým poslali do externého úložiska, odtiaľ vymazaný nebude, avšak nebude možné dostať sa k informáciám (pôvodne uloženým v metadátach), za čo bol získaný.

## 6. ZÁVER

V tejto práci sme sa venovali použitiu odznakov vo výučbe. Najprv sme popísali, čo odznaky sú, ako vznikli, na aký účel sa používajú, popísali sme rôzne metodiky ich použitia a pozitívny vplyv používania odznakov na vyučovací proces. Ďalej sme predstavili odznaky typu Open Badges, ako fungujú, ich výhody a nevýhody a kde sa momentálne používajú. Taktiež sme uviedli možnosti ich použitia v systéme Moodle.

Ako ukazujú mnohé štúdie, odznaky sú dobrým nástrojom motivácie študentov. Preto začínajú byť v súčasnosti široko využívané rôznymi vzdelávacími inštitúciami. Myslíme si, že je vhodné začať ich zaraďovať do vyučovania aj na našich školách, a to či už v jednoduchších formách, napríklad ako obrázky, alebo využitím infraštruktúry Open Badges.

Tento príspevok je súčasťou pilotnej štúdie nášho výskumu, v ktorom sa chceme ďalej uberať práve týmto smerom. V tomto polroku robíme výskum na gymnáziu, kde nasadzujeme do jednej triedy fyzické odznaky a od budúceho školského roka plánujeme na portáli matfyz.sk implementovať digitálne odznaky typu Open Badges a začať ich využívať v niekoľkých triedach gymnázia. Odznaky budeme vydávať aj zobrazovať.

## 7. POĎAKOVANIE

Príspevok je súčasťou výskumu v projekte VEGA 1/0948/13.

## 8. BIBLIOGRAFICKÉ ODKAZY

- [1] ABRAMOVICH, S., SCHUNN, Ch., HIGASHI, R. M. *Are badges useful in education?: it depends upon the type of badge and expertise of learner*. Education Tech Research Dev (2013) 61:217–232. DOI 10.1007/s11423-013-9289-2
- [2] BARKER, B. *Digital Badges in Informal Learning Environments*. ICIW 2013, The Eighth International

Conference on Internet and Web Applications and Services, 2013.

- [3] DETERDING, S. et al., *Gamification: Toward a definition*. CHI 2011, Gamification Workshop. Vancouver, Canada. Dostupné na: <http://hci.usask.ca/uploads/219-02-Deterding.-Khaled.-Nacke.-Dixon.pdf>
- [4] GIBSON, D., OSTASHEWSKI, N., FLINTOFF, K., GRANT, S., KNIGHT, E. *Digital badges in education*. Education and Information Technologies, Springer Science+Business Media New York 2013, DOI 10.1007/s10639-013-9291-7
- [5] GOLOGOSKI, E. *Motivating the Learner: Mozilla's Open Badges Program*, Stanford University 4
- [6] [https://docs.moodle.org/28/en/About Moodle](https://docs.moodle.org/28/en/About_Moodle)
- [7] <https://docs.moodle.org/28/en/Badges>
- [8] LOBKOWITZ, F. *Encyklopedie řádů a vyznamenání*, LIBRI, Praha 1999, ISBN 80-85983-71-0
- [9] MAREŠ, J. *Pedagogická psychologie*. In: J.Průcha: Pedagogická encyklopedie, Portál 2009
- [10] REID, A. *Digital Digs*. 15 September 2011. Referenced at <http://www.alex-reid.net/2011/09/welcome-to-badge-world.html> on February 7, 2012. 3
- [11] RUGHINIS, R. *Talkative objects in need of interpretation. re-thinking digital badges in education*. CHI'13 Extended Abstracts on Human Factors in Computing Systems. ACM, 2013.
- [12] SANTOS, C., et al. *Students' Perspectives on Badges in Educational Social Media Platforms: The Case of SAPO Campus Tutorial Badges*. Advanced Learning Technologies (ICALT), 2013 IEEE 13th International Conference on. IEEE, 2013
- [13] THE MOZILLA FOUNDATION, PEER 2 PEER UNIVERSITY Open Badges for Lifelong Learning - Exploring an open badge ecosystem to support skill development and lifelong learning for real results such as jobs and advancement, Working Document, in collaboration with The MacArthur Foundation
- [14] WATTERS, A. *Mozilla's Open Badges Project: A New Way to Recognize Learning*. MindShift. 10 August 2011. Referenced at <http://mindshift.kqed.org/2011/08/mozillas-open-badges-project-a-new-way-to-recognize-learning/#more-14421> on February 5, 2012.
- [15] ŽILINCOVÁ, M. *Motivácia žiaka*, Centrum pedagogicko - psychologického poradenstva a prevencie Námestovo, dostupné online: <http://www.cppap.orava.sk/METODIKY/Motiv%C3%A1cia%20%C5%BEiaka.pdf>
- [16] <http://cs.wikipedia.org/wiki/Odznak>



*Postre*

*Posters*

# Desaťročie výučby informatiky na Univerzite tretieho veku na UMB v Banskej Bystrici

Jan Klimeš, Ivan Brodenec, Dana Horváthová, Jarmila Škrinárová  
katedra informatiky FPV UMB  
Tajovského 40  
974 01 Banská Bystrica  
Slovensko

[jan.klimes@umb.sk](mailto:jan.klimes@umb.sk), [ivan.brodenec@umb.sk](mailto:ivan.brodenec@umb.sk),  
[dana.horvathova@umb.sk](mailto:dana.horvathova@umb.sk), [jarmila.skrinarova@umb.sk](mailto:jarmila.skrinarova@umb.sk)

## ABSTRAKT

Cieľom článku je priblížiť štúdium informačných a komunikačných technológií v rámci Univerzity tretieho veku na Univerzite Mateja Bela v Banskej Bystrici od jeho začiatku v roku 2003 až do súčasnosti.

## ABSTRACT

The aim of the article is to present the study of information and communication technologies within the University of the Third Age at Matej Bel University in Banská Bystrica from the beginning in 2003 until now.

## Kľúčové slová

Univerzita tretieho veku, vzdelávanie seniorov, informačné a komunikačné technológie.

## Keywords

University of the Third Age, Education of seniors, information and communication technologies.

## 1. ÚVOD

Vzdelávanie na Univerzite tretieho veku (UTV) je organizované v rámci celoživotného vzdelávania občanov. UTV Univerzity Mateja Bela v Banskej Bystrici (UMB v BB) je členom Asociácie univerzít tretieho veku (ASUTV) na Slovensku a tým je zároveň prepojená aj na medzinárodné organizácie tohto charakteru, ako je: Medzinárodná asociácia univerzít tretieho veku (AIUTA), Európska federácia starších študentov (EFOS) a Európska federácia starostlivosti o starších (EURAG).

UTV UMB v BB bola založená v septembri roku 1992. Za vyše 23 rokov prešla táto ustanovizeň mnohými fázami svojho vývoja. Počas týchto rokov tu študovalo vyše pätnásť tisíc záujemcov. UTV UMB sa tak stala druhou najväčšou univerzitou tretieho veku zo 16 UTV na Slovensku, ktorá ponúka záujemcom o štúdium približne 20 študijných odborov, ktoré každoročne aktualizuje podľa záujmu a požiadaviek študentov. Viac informácií o štúdiu na: [http://www.umb.sk/umb/umbbb.nsf/page/Univerzita\\_tretieho\\_veku\\_UMBp1](http://www.umb.sk/umb/umbbb.nsf/page/Univerzita_tretieho_veku_UMBp1). Štúdium dopĺňajú rôzne odborné exkurzie, návštevy galérií, múzeí, divadla, či spoločné umelecko-spoločenské podujatia.

## 2. ZAČIATKY ODBORU IKT

Od roku 2003 sa podarilo rozbehnúť študijný odbor Informačné a komunikačné technológie (IaKT), ktorého odborným garantom sa stal doc. Ing. Jan Klimeš, CSc, ktorý do harmonogramu prednášok jedného roka zaradil šesť hlavných tém súčasnej informatiky. Tieto sa neskôr rozšírili o ďalšie aktuálne témy na ďalšie roky štúdia a kolektív prednášajúcich sa rozšíril o ďalších pracovníkov najmä katedry informatiky (Horváthová, Škrinárová, Siládi, Huraj, Siláči,

Brodenec, Sebinová, Lehoťan, Martincová, Turanská). Autori študijného plánu vychádzali zo zásady, že štúdium by nemalo mať charakter kurzu, ktoré vtedy poskytovali mnohé firmy na ovládnutie niektorých aplikačných programov - Word, Excel a pod. Študentom odboru sme chceli ponúknuť aj určité množstvo teoretických poznatkov, aby hlbšie porozumeli práci technického i programového vybavenia a boli pripravení aj na novinky, ktoré sa v tomto vednom odbore neustále objavujú. Správnosť tejto koncepcie sa potvrdila najmä v prípade študentov, ktorí ešte neboli v dôchodkovom veku a poznatky získané štúdiom odboru IKT uplatnili v svojom povolani.

## 3. ŠTÚDIUM IKT DNES

Dnes štúdium pozostáva z dobrovoľného prípravného ročníka pre tých, ktorí nemali žiadne skúsenosti s prácou s počítačom. Nasledujú tri ročníky, zamerané najmä na úvod do IKT, získavanie potrebných zručností v práci s hardvérom, základným softvérom, adresármí a súbormi, internetom, textovým editorom, tabuľkovým kalkulátorom, grafickým editorom, multimédiami a programom na tvorbu prezentácií. Tieto tri ročníky končia študenti obhajobou absolventskej práce na tému, ktorú si sami zvolia. Do dnešného dňa ukončilo toto štúdium už 52 absolventov.

Pre veľký záujem boli zostavené ďalšie nadstavbové ročníky venované angličtine v IKT, fotografovaniu a spracovaniu fotiek a využitiu IKT v architektúre.

Všetci študenti majú k dispozícii počítač, na ktorom si môžu priebežne overovať vysvetľované poznatky. Tým je obmedzený aj počet študentov - jednak dostupným počtom počítačov, ale aj kapacitou vyučujúceho, ktorý sa často venuje jednotlivým študentom individuálne. Je žiadúce, ak majú študenti prístup k počítaču aj doma a môžu si tak získané poznatky upevňovať, prípadne riešiť zadané úlohy.

## 4. ZÁVER

UTV otvára široké možnosti vzdelávania pre občanov staršieho veku a láme bariéry pred využitím moderných technológií. Uspokojuje túžbu po vedomostiach, ponúka aktivity pre kvalitné využitie voľného času, pozitívne ovplyvňuje priebeh starnutia a kvalitu života. Plní aj rehabilitačnú funkciu, spojenú so znovuoobjavením a udržiavaním fyzických i psychických síl záujemcov. Prispieva k rozvoju záujmov, potrieb a zážitkov starších ľudí, vedie ich k aktívnej komunikácii s okolím, niekedy aj k zmene hodnotovej orientácie, často ich odpútava od vlastnej choroby k znovu nájdeniu životného optimizmu. Neoceniteľnou skutočnosťou, sú nové priateľstvá, ktoré sa v priebehu štúdia nadväzujú, a ktoré pretrvávajú aj po skončení štúdia.

# Implementácia IKT do vzdelávania na NHF EU v Bratislave

Agneša Gašperanová

Katedra aplikovanej informatiky a výpočtovej techniky, Národohospodárska fakulta, Ekonomická univerzita v Bratislave  
Dolnozemska cesta 1, 852 35 Bratislava, Slovensko, [agnes.gasperanova@euba.sk](mailto:agnes.gasperanova@euba.sk)

## ABSTRAKT

Cieľom výučby informatiky na NHF EU v Bratislave je naučiť študenta orientovať sa v oblasti IKT, s dôrazom na efektívnu prácu v prostredí základných aplikácií MS Office.

## ABSTRACT

The focus of informatics courses is to acquaint students with the Information Systems, focusing on work effectively with basic MS Office applications.

## Kľúčové slová

kancelársky balík, textový editor, tabuľkový kalkulator

## Keywords

Text processing, spreadsheet, electronic presentation

## 1. ÚVOD

Informačné a komunikačné technológie (IKT), sú dôležitým nástrojom spracovanie údajov. Absolventi Národohospodárskej fakulty Ekonomickej univerzity (NHF EU) patria do kategórie koncových používateľov. Predmet informatika je zameraná na vyučovanie a prehĺbenie zručností študentov, vo využívaní základných programov kancelárskeho balíka MS Office.

## 2. Výučba IKT na NHF EU

Študenti bakalárskeho štúdia NHF EU absolvujú predmety IKT v rámci spoločného základu. Predmet je v prvom ročníku povinný, v zimnom semestri je prednáška a cvičenie, v letnom semestri cvičenie, v treťom semestri štúdia je predmet povinne voliteľný.

Prednášky poskytujú všeobecný prehľad, definíciu základných pojmov, potrebných pre koncového používateľa IKT v oblasti administratívy a spracovania údajov. Cieľom predmetu je aj oboznámiť študentov s aktuálnymi informáciami z oblasti IKT

## 3. PRAKTICKÁ PRÍPRAVA

Na cvičeniach využívame najrozšírenejšie aplikačné programové vybavenie elektronickej kancelárie, MS Office.

### 3.1 Textový editor

Textový editor je väčšinou prvá aplikácia kancelárskeho balíka, s ktorým sa používateľ oboznámi. Študenti prichádzajú zo stredných škôl so základnými zručnosťami, úlohou cvičení je rozšírenie týchto zručností o pokročilejšie nástroje.

Štýly sú efektívnym nástrojom úpravy dokumentu. Cvičenia venujeme využívaniu a úprave štýlov. Objekty v textovom dokumente (grafika, tabuľky...) slúžia na skvalitnenie a oživenie dokumentu. Hromadná korešpondencia, jeden z najefektívnejších nástrojov textového editora, vyžaduje väčší časový priestor na cvičeniach.

### 3.2 Tabuľkový kalkulator

Zručnosť v spracovaní číselných údajov pomocou tabuľkového kalkulatora je kľúčovým momentom pri uplatnení sa našich absolventov. Tvorba a formátovanie tabuľky, zostrojenie jednoduchých vzorcov je príprava na využívanie zložitejších nástrojov. Prispôbenie odkazov vo výrazoch umožní efektívnu

tvorbu vzorcov, ale zmysluplné využívanie absolútnych a miešaných adries robí niektorým študentom veľké problémy.

Grafické zobrazenie údajov tabuľky je atraktívny a prehľadný nástroj, patrí medzi najobľúbenejšie kapitoly. Výber zdrojových údajov a vhodného typu grafu sú kľúčovým momentom.

Excel ponúka veľké množstvo funkcií, nemôžeme však všetky precvičiť. Nevyhnutné je, aby študenti dokázali nástroj správne použiť, potom v prípade potreby nájdú vhodnú funkciu na riešenie úlohy. Preberáme základné funkcie z kategórií, ktoré sú dôležité pre ekonomické výpočty (logické, vyhľadávacie, finančné, databázové), vytvárame jednoduché aj vnorené funkcie.

Databázové nástroje slúžia na spracovanie rozsiahlych tabuliek, základom je usporiadanie tabuľky a filtrovanie, automatický a rozšírený filter.

Kontingenčná tabuľka a kontingenčný graf sú efektívne analytické nástroje spracovania databázovej tabuľky. Umožnia získať prehľadné informácie z databázy. Vytvorenie a úprava tabuľky je otázkou niekoľkých minút, tabuľky sú navyše ľahko modifikovateľné. Kritickým bodom návrhu je umiestnenie položiek databázy do vhodnej oblasti polí. Nástrojom grafického zobrazenia údajov kontingenčnej tabuľky je kontingenčný graf.

## 3.3 Elektronická prezentácia a databáza

Cvičenie tretieho semestra je rozdelené na dve samostatné časti, prvá polovica semestra je venovaná tvorbe elektronickej prezentácie. Stručný obsah cvičení: zobrazenie a úprava predlohy prezentácie, výber motívu, vytváranie prezentácie, vkladanie snímok, úprava obsahu, prechody snímok a animácie, príprava na publikovanie. Okrem technickej stránky sa venujeme aj kvalite prezentácie. Študenti na konci bloku pripravujú a prednesú spolužiakom a vyučujúcemu vlastnú prezentáciu.

Druhá polovica semestra je venovaná základom práce sa databázou v programe MS Access. Úlohou je oboznámiť študentov so základnými pojmami relačnej databázy: tabuľka, relácia. Predmet poskytuje stručný pohľad na základné nástroje aplikácií, neusilujeme sa o úplnosť, ale cvičenie dáva dostatočné informácie k tomu, aby si študenti dokázali rozšíriť a zdokonaľiť svoje zručnosti vo využívaní aplikácie samoštúdiom.

## 4. ZÁVER

Predmety využívania IKT sú momentálne zaradené do prvých semestrov bakalárskeho štúdia. Zmeny v ekonomickej oblasti, neustále rozširovanie využívania IKT v národnom hospodárstve ukazujú, že to nestačí. Plánujeme rozšíriť ponuku predmetov IKT do výučby inžinierskeho štúdia, to je ale zatiaľ hudba budúcnosti.

## 5. BIBLIOGRAFICKÉ ODKAZY

- [1] Stránka NHF EU v Bratislave. [online] Dostupné na: <http://nhf.euba.sk/katedry/katedra-aplikovanej-informatiky-a-vypoctovej-techniky>.

Tento príspevok vznikol s podporou grantovej agentúry VEGA v rámci projektu VEGA 1/0975/15, Makroekonomické a mikroekonomické prejavy a dôsledky inflácie a deflácie.

# Medzipredmetové vzťahy v predmete semestrálny projekt

Jana Jacková, Ľudovít Trajtel'

Katedra informatiky FPV UMB

Tajovského 40

974 01 Banská Bystrica

Slovensko

jana.jackova@umb.sk, ludovit.trajtel@umb.sk

## ABSTRAKT

V príspevku predstavíme povinný predmet *semestrálny projekt* (ďalej SP) z hľadiska jeho začlenenia do bakalárskeho študijného programu *aplikovaná informatika* na Fakulte prírodných vied Univerzity Mateja Bela v Banskej Bystrici. Študenti učiteľstva informatiky v kombinácii predmetov môžu tento predmet študovať ako výberový.

## ABSTRACT

In this article, we introduce a compulsory subject *Semestral Project* from the point of view its inclusion in the bachelor study program *Applied Computer Science* at the Faculty of Natural Sciences, the Matej Bel University in Banská Bystrica. Students of *Computer Science Teacher Training* can study this subject as elective.

## Kľúčové slová

bakalársky program, informatika, projekt, medzipredmetové vzťahy, peer review, Moodle, PeerWise

## Keywords

bachelor program, computer science, project, cross-curricular relations, peer review, Moodle, PeerWise

## 1. PREDMET SEMESTRÁLNY PROJEKT

Podľa [1, s. 99-104] sa predmet SP vyučuje v prvých troch semestroch (ďalej SP1, SP2, SP3), ako uvádzame v Tabuľke 1. Cieľom SP je, aby študent vedel „samostatne pracovať na projektovej úlohe, samostatne spracovať dokumentáciu, prezentovať a obhájiť odbornú prácu z oblasti informatiky.“ [2]

**Tabuľka 1. Bakalárske štúdium aplikovanej informatiky**  
(spracovali autori podľa [1, s. 99-104])

predmety <sup>1</sup>	ZIMNÝ SEMESTER	LETNÝ SEMESTER
1. ročník	matematická analýza 1 algoritmy a štruktúry údajov 1 úvod k informatike <b>semestrálny projekt 1</b> informačné technológie špeciálny jazykový seminár AJ	matematická analýza 2 diskrétna matematika programovanie 1 DTP systémy internetové technológie 1 odborná angličtina <b>semestrálny projekt 2</b>
2. ročník	programovanie 2 algoritmy a štruktúry údajov 2 databázové systémy 1 <b>semestrálny projekt 3</b> počítačové systémy 1	programovanie 3 operačné systémy 1 počítačová grafika 1 jazyky a syntaktická analýza ...výber témy bakalárskej práce
	PV, V ...podľa výberu študenta z ponuky	
3. ročník	operačné systémy 2 počítačové siete 1 multimédiá <b>bakalársky projekt</b>	<b>bakalárska práca</b> počítačové systémy 2 softvérové systémy 1 informačná bezpečnosť 1 družicové navig. a poloh. systémy
	PV, V ...podľa výberu študenta z ponuky	

<sup>1</sup> typ: P – povinný, PV – povinne voliteľný, V – výberový

Na začiatku semestra učiteľ zorientuje študentov v témach odboru informatika; používa pri tom študijný program [1, 2], klasifikačný systém ACM [3], ponuku ostatných učiteľov katedry ap. Každý študent si vyberá tému, ktorú si dá schváliť učiteľovi predmetu. Študent môže v SP2 a SP3 pokračovať v téme z predchádzajúceho semestra alebo si vyberá novú tému; v SP2 a SP3 môže tiež pracovať na spoločnej téme so spolužiakom ako dvojica.

Výučbu predmetov SP1 a SP2 podporujeme týmito kolaboratívnymi aktivitami: (a) k prvým finálnym verziám projektov si študenti robia vzájomnú spätnú väzbu formou peer review, (b) pri prezentovaní tém používajú aj wiki stránku v prostredí Moodle a (c) pomocou systému PeerWise [4] sa navzájom učia (každý tvorí k svojej téme otázky a vkladá ich do systému, odpovedá na otázky ostatných a komentuje ich).

## 2. MEDZIPREDMETOVÉ VZŤAHY

V SP rozvíjame horizontálne aj vertikálne medzipredmetové vzťahy študijného programu. V 1. ročníku sa SP1 a SP2 dobre dopĺňa s predmetmi angličtiny (Tabuľka 1), pretože študent pri vypracovaní projektu používa aj anglické zdroje. Pri tvorbe dokumentácie využíva tiež vedomosti a zručnosti, ktoré získava v predmetoch informačné technológie a DTP systémy.

Odvodzením dokumentácie k SP študent preukazuje, že pozná zásady tvorby odborných textov [2]. Výsledný dokument v SP1 má štruktúru a formálnu úpravu podobnú ako v bakalárskej záverečnej práci, v SP2 je to odborný článok podľa predpísanej šablóny a v SP3 to môže byť napr. dokumentácia k softvérovej aplikácii. Absolvovaním SP je študent lepšie zorientovaný v nosných témach odboru informatika, čo môže uplatniť pri výbere svojich predmetov typu PV a V, a tiež pri výbere témy svojej bakalárskej práce (pozri Tabuľka 1).

Objavovanie medzipredmetových vzťahov v procese didaktickej analýzy učiva býva pre tvorivých učiteľov všetkých stupňov škôl veľkým zdrojom inšpirácie. Ich premyslené uplatňovanie vo výučbe môže edukantom priniesť lepšie pochopenie medzipredmetových súvislostí, a tiež väčšie uvedomenie si zmysluplnosti toho, čo sa učia. Dôsledkom môže byť zvýšenie záujmu o štúdium a lepšie študijné výsledky.

## 3. BIBLIOGRAFICKÉ ODKAZY

- [1] *Sprievodca štúdiom 2014/2015*. Banská Bystrica : FPV UMB, 2014. 229 s. ISBN 978-80-557-0588-0. Dostupné tiež na: <<http://www.fpv.umb.sk/studium/sprievodca-studiom>>.
- [2] Katedra informatiky FPV UMB. *Informačné listy o predmetoch 2014/2015*.
- [3] ACM. *The 1998 ACM Computing Classification System* [online]. [cit. 2015-02-25]. Dostupné na: <<http://www.acm.org/about/class/ccs98-html>>.
- [4] JACKOVÁ, J. Enhancing Engagement in Computer Science Courses through Student Created and Curated Question Banks. In *ITAT 2014, Part II*. Prague : Institute of Computer Science AS CR, 2014. ISBN 978-80-87136-19-5, p. 146.

# Inovácie kurikula v informatike na ZŠ

Janka Majherová  
Katolícka univerzita v Ružomberku  
Hrabovská cesta 1  
03401 Ružomberok  
Slovensko  
janka.majherova@ku.sk

## ABSTRAKT

V príspevku sa venujeme analýze súčasného stavu obsahu vyučovania predmetu informatika v nižšom sekundárnom vzdelávaní na Slovensku. Zameriame sa najmä na výučbu algoritmizácie a programovania v súvislosti s inováciami v Štátnom vzdelávacom programe.

## ABSTRACT

In this paper, we analyse actual trends in content teaching computer science in primary and secondary education in Slovakia. We focus on the teaching of algorithms and programming in the context of innovation in the State Education Programme.

## Kľúčové slová

Vyučovanie informatiky, nižšie sekundárne vzdelávanie, štátny vzdelávaci program, programovanie

## Keywords

Teaching of computer science, lower secondary education, state education programme, programming

## 1. ÚVOD

V súčasnosti sú zverejnené Inovované štátne vzdelávacie programy pre primárne a sekundárne vzdelávanie platné od školského roku 2015/2016 [1]. Aj v predmete informatika autori navrhli menšie úpravy počtu hodín ako aj obsahu učiva. Upravili sa názvy tematických okruhov: okruh Informácie okolo nás sa nazýva Reprezentácie a nástroje, Komunikácia prostredníctvom IKT sa mení na Komunikáciu a spoluprácu, okruh Postupy, riešenie problémov je detailnejšie rozpisovaný na podtémy a nazýva sa Algoritmické riešenie problémov [2]. V súvislosti s týmito zmenami sme sa pokúsili analyzovať vývoj vyučovania informatiky, najmä algoritmizácie a programovania, na základných školách na Slovensku od prvej reformy v roku 2008.

## 2. PRIESKUM VÝUČBY algoritmizácie

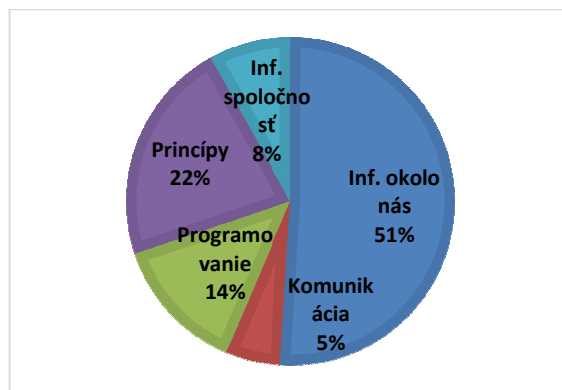
Na získanie dát sme použili metódu dotazníka a obsahovej analýzy textu v pedagogických denníkoch študentov odboru učiteľstvo informatiky na Pedagogickej fakulte KU v Ružomberku [3, s. 133].

Prvý prieskum sme realizovali v rokoch 2006-2008, kde sme porovnali výskyt výučby programovania s prácou s textom, tabuľkami, grafikou a s prezentáciami. Na základných školách sa v sledovanom období počas pedagogickej praxe študenti stretli s algoritmizáciou a programovaním v jazykoch Baltík alebo Imagine v piatich z pätnástich škôl.

Po istom časovom odstupe sme v roku 2013 realizovali prieskum na vzorke 66 učiteľov informatiky na ZŠ [podľa 4]. Zaujímala nás výučba programovania a konkrétne aký programovací jazyk učitelia používajú. Učitelia z celkového povinného počtu vyučovacích hodín venovali tematickému okruhu Postupy, riešenie problémov, algoritmické myslenie na 1. stupni ZŠ 17%

hodín a na 2. stupni ZŠ 20% hodín. Ako programovací jazyk preferovali na 1. stupni Baltík (53%), na druhom stupni Imagine (83% učiteľov). Prekvapivo málo učiteľov sa venuje príprave žiakov na súťaže v programovaní (iba 5,6 % na druhom stupni).

Pri prieskume v roku 2014 sme opäť analyzovali pedagogické denníky študentov, ktorí pozorovali výučbu alebo učili na 10 základných školách. Z počtu 150 vyučovacích hodín v predmete informatika na 2. stupni ZŠ sa najväčší priestor venoval výučbe celku Informácie okolo nás a princípom fungovania IKT. Programovaniu bolo venovaných 20 hodín, čo predstavuje priemerne 14% výučby (obr. 1).



Obrázok 1. Podiel výučby tém v predmete informatika na ZŠ

## 3. ZÁVER

Z analýzy dát prieskumov ako aj z rozhovorov a pozorovaní usudzujeme, že učitelia informatiky na základných školách majú stále rešpekt pred výučbou algoritmizácie a programovania. Pristupuje k tomu aj evidentný nezáujem žiakov o túto tému. Ten môže byť spôsobený aj nevhodnou metódikou výučby. Riešenie vidíme v propagácii vhodných metodických materiálov, motivácii učiteľov k príprave žiakov na súťaže ako aj kvalitnej didaktickej príprave budúcich učiteľov informatiky.

## 4. BIBLIOGRAFICKÉ ODKAZY

- [1] *Inovovaný štátny vzdelávaci program*. [online]. [cit. 2015-03-02]. Dostupné na: <<http://dl.acm.org/citation.cfm?doid=90417.90738>>.
- [2] SALANCI, L.: *Informatika v novom Štátnom vzdelávacom programe*. In DidInfo 2014. Banská Bystrica: UMB. ISBN 978-80-557-0698-6, s. 143-148.
- [3] Gavora, P.: *Úvod do pedagogického výskumu*. Bratislava: UK 2001. ISBN 80-223-1628-8
- [4] KONRÁDOVÁ, M.: *Tvorba kurikula v predmete informatika na druhom stupni ZŠ*. Záverečná práca. Ružomberok: PF KU 2013.

*Pracovné semináre*

*Workshops*



## **Tablety v školskom laboratóriu**

Mária Spišáková, Peter Spišák  
PMS Delta, s.r.o.

### **ABSTRAKT**

V rámci workshopu budú účastníci zoznámení s technológiou na bezdrôtové zdieľanie údajov a ovládanie meraní pomocou laboratórneho systému Vernier. Počas workshopu uskutočníme niekoľko laboratórnych pokusov, ktoré budú účastníci sami vyhodnocovať pomocou aplikácie Vernier Graphical Analysis priamo na tabletoch, namerané grafy, hodnoty a ovládanie merania budeme uskutočňovať pomocou Vernier LabQuest Viewer alebo pomocou webového rozhrania systémom Vernier Data Share.

## **Interaktívne testovanie žiakov v prostredí Samsung School a Socrative**

Robert Borbély, Peter Stach  
ZŠ Trieda SNP, Banská Bystrica

### **ABSTRAKT**

Účastníci workshopu sa naučia prakticky ovládať testovacie možnosti interaktívnych programov Socrative a Samsung School. Pri práci v malých skupinách troj- až štvorčlenných si vyskúšajú rolu učiteľa, ale aj žiaka.

- Ukážka testovania v prostredí aplikácie Socrative (20 min.)
- Ukážka testovania v prostredí aplikácie Samsung School (20 min.)
- Skupinová práca účastníkov workshopu (20 min.)

## **Spracovanie informácií na mobilných zariadeniach**

Ján Machaj, Miloš Bélik  
EDULAB, n.o.

### **ABSTRAKT**

V dnešnej dobe na nás číhajú informácie spoza každého rohu. Sú dostupnejšie, pestrejšie a intenzívnejšie ako kedykoľvek predtým. Dá sa povedať, že sú samozrejmosťou nášho života podobne ako nové technológie. V našom príspevku sa spolu pozrieme, ako si z dotykového zariadenia v podobe smartfónu alebo tabletu spraviť pomocníka pri spracovaní informácií z rôznych zdrojov. Predstavíme si aplikácie, ktoré pri tomto procese majú pomáhať a zjednodušovať prácu pri spracovávaní, triedení a zdieľaní informácií. Svojimi prednosťami a výhodami prekvapia viacerých, ktorí ešte ich funkcie nevyužívajú.

*Organizátori*  
*Organisers*

Katedra informatiky je pevnou súčasťou Fakulty prírodných vied Univerzity Mateja Bela. Má priznané právo udeľovať titul „bakalár“ a akademický titul „magister“ absolventom dennej formy štúdia a externej formy štúdia v študijnom programe Aplikovaná informatika a aj absolventom dennej formy štúdia v študijnom programe Učiteľstvo informatiky v kombináciách, ktoré sú zriadené podľa zákonných predpisov a uskutočňované na Fakulte prírodných vied Univerzity Mateja Bela. Katedra informatiky má priznané právo udeľovať akademický titul „doktor pedagogiky“ (v skratke „PaedDr.“) a akademický titul „doktor prírodných vied“ (v skratke „RNDr.“) absolventom študijných programov, ktorí získali titul „magister“, po vykonaní rigorózneho skúšky v študijnom odbore, v ktorom získali vysokoškolské vzdelanie alebo v príbuznom študijnom odbore uskutočňovanom na Fakulte prírodných vied Univerzity Mateja Bela. Katedra informatiky vykonáva aj výučbu mnohých celofakultných predmetov a celouniverzitných predmetov, ktoré sú buď povinnými, povinne voliteľnými alebo výberovými predmetmi na katedrách FPV UMB v Banskej Bystrici, resp. na iných katedrách; aj v rámci spoločného základu.

Zameranie edukačnej, vedeckej a výskumnej činnosti Katedry informatiky FPV UMB sa odvíja od hlavnej výskumnej línie fakulty, profilácie katedry a od záujmu jednotlivých členov katedry v symbióze s potrebou obohacovať predmety v študijných programoch o najnovšie poznatky vedy a výskumu v odbore.

Hlavnými smermi činnosti katedry sú:

- Teória vyučovania informatiky.
- Softvérové inžinierstvo.
- Umelá inteligencia.
- Aplikovaná informatika.

Katedru informatiky charakterizuje:

- vysoký dopyt po absolventoch našich odborov,
- široké spektrum uplatnenia absolventov v učiteľskej i odbornej praxi,
- kontakt so školami i s firmami z oblasti IT už počas štúdia,
- viacročná tradícia v príprave učiteľov a odborníkov pre oblasť informatiky,
- ústretové výučbové prostredie,
- tím skúsených akademických a vedeckých pracovníkov katedry,
- previazanosť teoretických a praktických poznatkov,
- primerane vybavené laboratóriá,
- možnosť absolvovať časť odborného štúdia na zahraničnej partnerskej univerzite.

Katedra je medzinárodne uznávaným a akceptovaným pracoviskom. Má kontakty s rôznymi domácimi aj zahraničnými pracoviskami a organizáciami, s ktorými dochádza k pravidelnej výmene učiteľov, študentov i ďalšej spolupráci v oblasti vedy a vzdelávania: Technická univerzita Ostrava, Metropolitní univerzita a ČVUT Praha, Česká republika; Oulu University of Applied Sciences, Oulu, Fínsko; Atilim University, Ankara a Fatih University, Istambul, Turecko; Utrecht University, Utrecht, Holandsko; Liepaja University, Liepaja, Lotyšsko; Corvinus University of Budapest, Maďarsko; Śląska Wyższa Szkoła Informatyczno-Medyczna, Chorzów, Poľsko; Technische Universität Chemnitz, Nemecko a ďalšie.

# METODICKO-PEDAGOGICKÉ CENTRUM

alokované pracovisko Banská Bystrica, Horná 97, 975 46 Banská Bystrica 1

**Metodicko-pedagogické centrum** je vzdelávacia inštitúcia pre učiteľov na Slovensku. Jej zriaďovateľom je Ministerstvo školstva Slovenskej republiky. Spolupracuje so štátnou školskou správou, Štátnou školskou inšpekciou, ŠPÚ, NÚCEM, vysokými školami a inými vzdelávacími a výskumnými inštitúciami na Slovensku i v zahraničí.

**Metodicko-pedagogické centrum** ponúka rozvoj kompetencií učiteľov materských, základných, stredných a špeciálnych škôl, školských zariadení:

- *na výkon riadiacich a pedagogických funkcií;*
- *v uplatňovaní inovácií z pedagogiky, psychológie, etiky, odboru, príbuzných vedných disciplín v riadení vyučovania a učenia sa žiakov;*
- *v uplatňovaní komunikačných technológií a didaktických prostriedkov vo vyučovaní;*
- *v tvorbe a riadení školských projektov – vrátane zahraničných projektov.*

Ponúka odborné a metodické materiály. Vydáva odborné a metodické časopisy **Pedagogické rozhľady**. V súčasnosti zvyšujeme ponuku priamej pedagogicko-poradenskej činnosti školám, školským zariadeniam a neprezenčného vzdelávania učiteľov aj prostredníctvom novovzniknutých regionálnych pracovísk.

**Metodicko-pedagogické centrum Ševčenkova 11, 850 05 Bratislava**  
**generálne riaditeľstvo**

Tel.: 02 48 20 94 11

## Regionálne pracoviská

### **BANSKÁ BYSTRICA**

Horná 97, 975 46 Banská Bystrica

**Tel.: 048 472 2999**

### **BRATISLAVA**

Ševčenkova 11, 850 05 Bratislava

**Tel.: 02 682 09910**

### **PREŠOV**

T. Ševčenku 11, 080 40 Prešov

**Tel.: 051 772 3451**

### **KOŠICE**

Ul. Zádielska 1, 040 01 Košice

**Tel.: 0917 516 196**

## Detašované pracoviská

### **NITRA**

Kozmonautov 5, 949 01 Nitra

**Tel.: 0917 516 191**

### **TRENČÍN**

Pod Sokolicami 14, 911 01 Trenčín

**Tel.: 032 744 3368**

### **ŽILINA**

Predmestská 1613, 010 01 Žilina

**Tel.: 0911 922 070**

### **TRNAVA**

Lomonosovova 2797/6

918 54 Trnava

**Tel.: 033 598 37 11**

## Centrum vedecko-technických informácií SR, ŠKOLSKÉ VÝPOČTOVÉ STREDISKO, BANSKÁ BYSTRICA

Školské výpočtové stredisko v Banskej Bystrici vzniklo 18. júla 1987 a v súčasnosti je pracoviskom Centra vedecko-technických informácií SR. Svoju činnosť realizuje v nasledovných oblastiach:

### **Informačná oblasť:**

ÚIPŠ – ŠVS v Banskej Bystrici prezentuje informácie o stredných školách v rámci prijímacieho pokračovania, informačný systém maturita, MS Select, programy na odvírovanie počítačov, výukové programy, výsledky súťaží.

Portál školská wikipédia - [wiki.svsbb.sk](http://wiki.svsbb.sk) poskytuje informácie o hardvéri, softvéri, videoprednáškach, videotutoriáloch, vyučovacích predmetoch. Wikiportál môžu používať učitelia, žiaci a verejnosť na vzdelávanie, vyhľadávanie informácií a podobne.

### **Spracovateľská oblasť, tvorba portálových riešení:**

Spracovanie informácií pedagogického procesu regionálneho školstva: prijímacie pokračovanie na stredné školy, vzdelávacie poukazy, maturity, celoštátne testovanie deviatakov, školské zahajovacie výkazy. Spracovanie je realizované na úrovni škôl, školských zariadení, Odboru školstva Obvodného úradu, VÚC, NÚCEM, MŠVVaŠ SR s podporou moderných otvorených portálových riešení, ktoré si vyžadujú minimálne finančné náklady.

### **Školiaca oblasť:**

Organizovanie vzdelávacích programov v oblasti IKT, určené pedagogickým a odborným zamestnancom škôl a školských zariadení. Vzdelávacie programy sú akreditované MŠVVaŠ SR v rámci aktualizácie vzdelávania a ich absolventi získavajú **kredity**. Z rovnakých akreditovaných programov ponúkame možnosť **overenia profesijných kompetencií**, získaných výkonom pedagogickej a odbornej činnosti alebo sebavzdelávaním. Tieto aktivity sú zamerané na základy práce s PC, MS Office, Open Office, elektronickú poštu a internet.

### **Metodická oblasť:**

Zabezpečuje metodickú pomoc základným a stredným školám pri spracovaní prijímacieho pokračovania na stredné školy, Centráram pedagogicko-psychologického poradenstva a prevencie pri organizovaní výstav a búr informácií pre žiakov končiacich povinnú školskú dochádzku ("Stredoškólák"). ZŠ a SŠ poskytuje odborné-metodické poradenstvo pri zavádzaní a využívaní IKT vo vzdelávaní.

### **Iné činnosti:**

V spolupráci s rôznymi organizáciami školstva a iných rezortov organizuje a zabezpečuje súťaže, podujatia miestneho, krajského, celoštátneho charakteru - školské, krajské kolá Zenit v programovaní, robotická súťaž FLL League.

**ADRESA: CVTI - ŠKOLSKÉ VÝPOČTOVÉ STREDISKO  
TAJOVSKÉHO 25  
975 73 BANSKÁ BYSTRICA  
<http://svs.edu.sk>**

*Sponzori a mediálni partneri*

*Sponsors and media partners*



# KVANT<sup>®</sup>



## *Quark*

*Magazín o vede a technike*

## PC REVUE



PMS *Delta*.sk



BANSKÁ BYSTRICA

Názov: **DidInfo 2015**  
Editor: doc. Ing. Ľudovít Trajtel', PhD.  
Technický editor: PaedDr. Ivan Brodenec, PhD., Ing. Dana Horváthová, PhD., Ing. Jana Jacková, PhD.  
Vydanie: 1. vydanie  
Vydavateľ: Univerzita Mateja Bela, Fakulta prírodných vied v Banskej Bystrici, Slovensko  
Rok: 2015  
Rozsah: 200 strán  
Formát: elektronický zborník  
Náklad: 80 ks  
ISBN: 978-80-557-0852-2

---

Title: **Proceedings of conference DidInfo 2015**  
Editor: doc. Ing. Ľudovít Trajtel', PhD.  
Technical Editor: PaedDr. Ivan Brodenec, PhD., Ing. Dana Horváthová, PhD., Ing. Jana Jacková, PhD.  
Edition: First edition  
Publisher: Matej Bel University, Faculty of Natural Sciences in Banská Bystrica, Slovakia  
Year: 2015  
Pages: 200  
Format: electronic proceedings  
Number of copies: 80  
ISBN: 978-80-557-0852-2