

KONŠTRUKCIONIZMUS. OD PIAGETA PO ŠKOLU V DIGITÁLNO M VEKU

PROF. RNDR. IVAN KALAŠ, PH.D., PAEDDR. MARTINA KABÁTOVÁ, PH.D., MGR. KATARÍNA MIKOLAJOVÁ, RNDR. PETER TOMCSÁNYI

ABSTRAKT

V našom príspevku sa budeme venovať prezentácii dvoch teórií učenia sa detí a žiakov, ktoré natrvalo ovplyvnili náš prístup k projektovaniu a implementácii predmetov informatika a informatická výchova, a tiež náš pohľad na význam digitálnych technológií pre modernú školu a vzdelávanie v 21. storočí. Domnievame sa, že kvalitný učiteľ informatiky a informatickej výchovy by mal poznať Piagetov konštruktivizmus a Papertov konštrukcionizmus – iba tak môže pochopiť význam logovskej kultúry v jej najširšom slova zmysle, bez ohľadu na to, či sa dnes prejavuje v programovacom prostredí Logo, v edukačnej robotike, v jazyku Scratch alebo v inej podobe.

Kľúčové slová: konštruktivizmus, konštrukcionizmus, logovská kultúra, edukačné programovanie

ÚVODNÁ MOTIVÁCIA

Čím dlhšie a systematickejšie premýšľame o cieľoch a formách informatiky a informatickej výchovy v modernom vzdelávaní, tým častejšie si kladieme otázky o tom, ako sa žiaci učia, ako môžeme plánovať, implementovať a hodnotiť ich komplexný rozvoj, akú úlohu môžu alebo majú hrať v tomto procese digitálne technológie, napr. v podobe edukačnej robotiky či edukačného programovania, ako môžeme čo najefektívnejšie prispieť k rozvoju žiackych kompetencií pre 21. storočie.

V 80-tych rokoch sme sa zapojili do medzinárodnej **logovskej komunity** a ako mladí programátori a nadšení pedagógovia sa postupne oboznamovali s dielom jej najvýznamnejšieho teoretika Seymoura Paperta. Študovať, rozumieť a tvorivo využívať myšlienky **konštrukcionizmu** však nie je jednoduché, pretože – ako pri každej komplexnej teórii – vedieť sa „jej očami“ pozrieť na súčasnosť a premýšľať o budúcnosti znamená kvalitne sa oboznámiť aj s jej koreňmi a východiskami.

Potrebu určitej zmeny, ďalšieho kroku dopredu, sme pocítili aj po 11. konferencii EuroLogo v Bratislave v roku 2007 a pri príprave jej pokračovania v Paríži sme pristúpili nie len k zmene názvu, ale aj k jasnejšej orientácii na *nové pedagogiky v digitálnom veku*. Prečítajme si niekoľko úryvkov z príhovoru Richarda Nossa členom logovskej komunity:

*V časoch, kedy vzniklo Logo (a neskôr konferencia EuroLogo), vytvorila sa na počet malá ale silná skupina pedagógov a počítačových vedcov, ktorí začali vnímať Logo ako viac než obyčajný programovací jazyk. Predstavovalo pre nich spôsob **uvažovania o učení sa**, metaforu toho, ako sa môžeme učiť efektívne. Učiť sa, ale tiež konštruovať, zdokonaľovať či zdieľať spôsobmi, pre ktoré Logo poslúžilo ako dobrý príklad. ... Najdôležitejší sen tvorcov Loga a ich štúpcov však zostal prevažne nespĺnený. Snívali o tom, že Logo môže iniciovať transformáciu – nie iba **spôsobov**, ako sa učíme, alebo **metód**, akými nás učia – ale predovšetkým toho, čo sa dá učiť a naučiť.*

*Môžeme poukázať na niektoré pozoruhodné úspechy. Ak nám však ide o **podstatnú zmenu**, nemôžeme prehliadať, že vznikli nové skupiny, nové hranice, ktoré treba prekonať, a nové oblasti poznávania, ktoré treba prizvať a vymeniť si s nimi poznanie. Je čas, aby sa EuroLogo posunulo ďalej. A práve to chceme vyjadriť aj zmenou mena konferencie na Konštrukcionizmus. Konštrukcionizmus, slovo, ktorým Seymour Papert označil pedagogiku postavenú na vytváraní a zdieľaní fyzických, virtuálnych aj intelektuálnych štruktúr – jasné príspevky k transformácii, ktorú chceme podporiť...*

R. Noss, London Knowledge Lab, Institute of Education, University of London
www.aup.edu/news/downloads/constructionism2010_Noss.pdf

Inšpirovaní úspechom konferencie Constructionism 2010, pozri [3], sme systematickému štúdiu tejto témy venovali celý nasledujúci zimný semester katedrového seminára. V krátkom prednáškovom bloku ju chceme priblížiť aj účastníkom konferencie Didinfo 2011.

Východiskom Papertovho konštrukcionizmu bol, prirodzene, Piagetov konštruktivizmus¹. Aj keď je lákavé – a potrebné ísť pri štúdiu aj hlbšie do minulosti, k Deweymu, Montessori, Freinetovi a ďalším, okolnosti nám to pre túto príležitosť nedovoľujú, našim východiskom bude Piaget a Vygotsky. Ďalej sa pozrieme na to, ako sa konštruktivizmus prejavuje dnes, a to v modernej didaktike matematiky, pozri [11], pretože (a) Hejný zohral významnú úlohu pri formovaní našich predstáv o poznávacom procese detí, a tiež (b) modernú didaktiku matematiky považujeme za najbližší zdroj inšpirácií pri rozvoji didaktiky informatiky a informatickej výchovy.

¹ Papert navrhol jednoduchú mnemotechnickú pomôcku, ako si uľahčiť rozlišovanie týchto dvoch krkolomných slov: V slove konštruktivizmus zdôrazníme písmeno V - konštruktivizmus - prípadne ho zjednodušené označíme **V-teória**. V slove konštrukcionizmus urobíme to isté s druhým písmenom n: konštrukcionizmus alebo **N-teória**.

Celý zvyšok nášho bloku budeme venovať vzniku jazyka Logo a krátkej histórii edukačného programovania, logovskej kultúre, vlastnému konštrukcionizmu a súčasným prejavom spomínaných fenoménov, ktoré súvisia s modernými programovacími prostrediami pre žiakov. Budeme uvažovať aj o tom, kam asi smeruje detské edukačné programovanie a aké je jeho miesto v predstavách o výchove a vzdelávaní detí a žiakov pre život v 21. storočí.

1 PIAGET A VYGOTSKY: MODERNÁ TEÓRIA UČENIA SA DETÍ

Venujme sa najprv novodobým východiskám moderného pohľadu na učenie sa (mladých ľudí). Aj keď je lákavé a produktívne hľadať ešte staršie zdroje a prevratné práce najvýznamnejších autorov, našim štartovacím bodom budú Piaget a Vygotsky – revoluční psychológovia detskej mysle, ktorí pôsobili v 20. storočí. Významne pritom využijeme práce súčasnej teoreticky konštruktivizmu E. Ackermann, pozri napr. [1].

1.1 Jean Piaget (1896 — 1980) a konštruktivizmus

Jean Piaget bol významným psychológom detského myslenia. Venoval sa vývinovej psychológii a epistemológii, dnes by sme povedali, že jeho odborom bola kognitívna veda. Piagetova práca je komplexné dielo, ktoré sumarizuje a podáva odpovede na mnohé otázky týkajúce sa poznávania, vedomia, mentálnych aktivít a nadobúdania vedomostí a zručností. Svoju pozornosť venoval skúmaniu poznávacích procesov od najrannejších fáz vývinu dieťaťa. S deťmi strávil mnoho času – viedol psychologické experimenty na overenie svojich teórií a často sa s deťmi rozprával. Spoločné dialógy pomohli Piagetovi odhaliť mnohé tajomstvá detskej mysle.

Jeden z jeho **experimentov bol zameraný na skúmanie usporiadania** – deti rôzneho veku dostali za úlohu usporiadať 8 paličiek rôznej dĺžky tak, aby boli zoradené od najmenej po najväčšiu. Dieťa v *predoperačnom štádiu* (3,5 roka) nedokáže usporiadať celú postupnosť. Vie porovnať dve paličky medzi sebou a povedať, že jedna je kratšia ako druhá, ale nevie to spojiť s rovnakou informáciou o ďalšej dvojici paličiek – tomuto dieťaťu ešte chýba pochopenie tranzitívnosti operácií a bude trvať niekoľko ďalších rokov, než sa v jeho mysli vytvorí reprezentácia usporiadania viacerých objektov. Iné dieťa vo veku 6,5 roka sa už nachádza v *štádiu konkrétnych operácií* – úlohu dokáže vyriešiť iba metódou pokus-omyl, a tiež nemá v mysli úplnú reprezentáciu usporiadania. Prejaví sa to tak, že nevie popísať, ako úlohu riešilo. Posledné dieťa v experimente má 9 rokov a už sa nachádza v štádiu formálnych operácií – úlohu efektívne, s ľahkosťou a precízne vyrieši, navyše dokáže presne popísať svoj postup a odpovedať na doplňujúce otázky (Koľko paličiek je kratších ako najdlhšia z nich? Odpovedá správne, aj keď asistentka zakryje časť paličiek papierom – to znamená, že *rozumie usporiadaniu na abstraktnej úrovni*.)

Podľa Piageta má detské myslenie svoju logiku a vnútornú štruktúru, aj keď sa často môže dospelým zdať, že deti sa mýlia. Ich myslenie je však prispôsobené aktuálnemu vývinovému štádiu a dobre odráža potreby dieťaťa v čase, keď jeho najdôležitejšou úlohou je vyvíjať sa a učiť sa nové veci.

Piaget je autorom teórie učenia sa – **konštruktivizmu**. Teória učenia sa vysvetľuje a predvída javy, fakty, udalosti a pozorovania v rámci procesu učenia sa. Pre vyučovaciu prax poskytujú teórie učenia sa hodnotné informácie, pretože ak získame podrobný popis a vysvetlenie procesu učenia sa, dokážme na základe týchto poznatkov navrhnúť vyučovacie metódy, intervencie pre vyučovanie alebo prostredia pre učenie sa, ktoré budú lepšie reflektovať potreby detí.

Piaget tvrdí, že **učenie sa** je aktivita, ktorá hľadá rovnováhu medzi vyvíjajúcou sa vrodenu štruktúrou mysle a vonkajšími vplyvmi, ktorými sa človek snaží porozumieť a prispôbiť. Dieťa, ktoré do svojej štruktúry mysle zaraďuje nový poznatok, musí opustiť svoju fungujúcu predstavu o danom jave. To si vyžaduje viac než len to, aby sa stretlo s lepšou teóriou (napr. vďaka učiteľovi či rodičovi). Táto zmena môže vyplývať jedine z jeho vlastnej činnosti, zo skúsenosti. Preto Piaget hovorí, že *ľudská inteligencia je ukotvená v činnosti*.

Z Piagetovej teórie konštruktivizmu vychádzajú viaceré závery, pozri [1]:

- **Učenie nikdy nemôže byť transmisívne**, či sa nám to páči alebo nie. Deti jednoducho nevstrebávajú to, čo sa im hovorí. Namiesto toho interpretujú povedané v perspektíve svojich vlastných vedomostí a skúseností.
- **Vedomosť nie je informácia** (komodita, ktorá sa dá jednoducho odovzdať a prijať v nezmenenej podobe), vzniká len na základe skúsenosti.
- **Teórie učenia sa, ktoré ignorujú detské miskoncepce a aktuálny obsah mysle, sa mňajú účinkom**. Deti majú dobré dôvody na to, aby zostali pri svojich predstavách o svete okolo seba a nezáleží na tom, akú pravdivú a rozumnú teóriu im predložíme ako alternatívu. Učiteľ z nášho pohľadu nemá byť poskytovateľom univerzálnych pravd, ale sprievodcom, ktorý pomôže dieťaťu objavovať, realizovať sa, vymieňať si skúsenosti s ostatnými a napokon si rozšíriť svoje poznanie.

Mimoriadny príspevok Piageta k vzdelávaniu je tom, že sa zamerával na to, ako dieťa premýšľa, keď sa jeho doterajšie presvedčenia rozpadajú, keď sa vstrebáva alternatívne vysvetlenie a porozumenie, keď sa úprava a rozšírenie doterajších obrazov stáva pre dieťa nutnosťou.

Konštruktivizmus vidí nadobúdanie vedomostí a zručností ako aktívny proces, na ktorého počiatku sú vrodené štruktúry, ale tento proces je ovplyvnený prostredím, vnímaním, zvyklosťami aj sociálnym kontextom. Pre efektívne učenie sa musí žiak *dostať priestor pracovať s učivom*. Podľa daného vývinového štádia má v nižšom veku dôležité postavenie manipulácia s reálnymi objektmi, neskôr prídu na rad mentálne reprezentácie a abstraktné pojmy. Dôležité je poznať východiskový stav poznania žiaka, aby sa nové vedomosti mohli dostatočne kvalitne ukotviť v štruktúrach už získaných vedomostí. Výskumy ukázali, že aj deti v rannom veku majú prvotné naivné predstavy o mnohých javoch (napríklad vysvetlenia fyzikálnych, algebrických či geometrických javov). S týmito predstavami sa dá efektívne pracovať a využiť ich na výstavbu hodnotných poznatkov.

Piaget zachytáva najmä to, čo je spoločné a **typické pre myslenie** detí v konkrétnom vývinovom štádiu, nezaujímajú ho individuálne rozdiely medzi deťmi (súčasní odborníci mu práve tento nedostatok vytýkajú). Dalo by sa povedať, že objektom Piagetovho skúmania je *typické dieťa*, ktoré reprezentuje čo najväčšiu skupinu.

Piaget a konštruktivizmus významne prispeli k tomu, že sa psychológovia začali seriózne zaoberať detským myslením, jeho špecifikami a vývinom.

1.2 Vygotsky (1896 — 1934) a sociálny rozmer konštruktivizmu

Tento významný psychológ pôvodne vyštudoval právo. Venoval sa výskumu kognitívneho vývinu dieťaťa, jeho reči a kognitívnych procesov. Vo svojich vedeckých prácach vysvetľoval, ako interakcie dieťaťa s dospelými prispievajú k vývinu jeho schopností a zručností.

Jeho najvýznamnejší príspevok k teóriám učenia sa spočíva v tom, že pridal ku konštruktivizmu aj **sociálny rozmer** – podľa Vygotského sa dieťa učí a vyvíja jedine v kontakte s ostatnými – so spoločnosťou, v ktorom žije. Takto sa dieťa sústavne učí od svojich rodičov, učiteľov, súrodencov a kamarátov. Takéto učenie sa je však rôzne v rôznych kultúrach a rôznych spoločnostiach, je závislé na jazyku a iných vyjadrovacích prostriedkoch, ktoré používame a ktoré sú v rôznych kultúrach rozdielne. Vygotsky súhlasí s Piagetom v tom, že poznanie si musí skonstruovať každý sám svojou skúsenosťou – je to aktívny proces. Na rozdiel od Piageta však zdôrazňuje, že takýto zážitok dieťa najprv spracuje tak, že ho *zdieľa s druhými, a až potom samo pre seba*.

Vygotského zaujal rozdiel v tom, čo sa môže dieťa naučiť samo bez pomoci a čo sa môže naučiť za asistencie niekoho dospelého alebo skúsenejšieho. *Aktuálny vývinový stupeň* predstavujú úlohy, ktoré dokáže dieťa riešiť samostatne. *Potenciálny vývinový stupeň* predstavujú úlohy, ktoré dokáže vyriešiť za pomoci niekoho iného (rodiča, učiteľa, kamaráta, spolužiaka a pod.). Vygotsky tvrdí, že potenciálny vývinový stupeň je pri hodnotení schopností dieťaťa omnoho dôležitejší. Rozdiel medzi spomínanými vývinovými stupňami nazval **zóna najbližšieho vývinu** (angl. ZPD – zone of proximal development). Rodičia a učitelia môžu pomôcť dieťaťu učiť sa tým, že mu budú zadávať a riešiť s ním úlohy, ktoré ležia v zóne jeho najbližšieho vývinu – neustále tak budú posúvať hranice jeho poznania a schopností.

2 HEJNÝ: KONŠTRUKTIVIZMUS A MATEMATIKA

Teórie Piageta a Vygotského ovplyvnili pohľad na moderné vzdelávanie mnohých didaktikov a učiteľov. Z predmetov nám blízkych je to celkom prirodzene aj matematika a jej pre nás najvplyvnejší didaktik M. Hejný. Jeho dielo, pozri napr. [11], je pre nás vzrušujúcou odpoveďou na otázku, ako vyzerá konštruktivizmus dnes.

Neobyčajne zaujímavá publikácia *Dítě, škola a matematika* obsahuje hneď niekoľko tém, ktoré odhaľujú miesto a podobu konštruktivistických postupov v modernom vyučovaní matematiky. Kniha podáva pomerne ostrú kritiku inštruktívneho a transmisívneho vyučovania, vedenia žiakov k formalizmu, ktoré podľa autorov málokedy vedú k vytvoreniu živého poznatku. Ukazujú veľké množstvo príkladov z reálnej praxe, ktoré riešia rovnaké didaktické situácie konštruktivisticky.

Autori do detailov rozoberajú jednotlivé fázy poznávacieho procesu:

- motiváciu,
- izolované modely,
- univerzálne modely,
- kryštalizáciu a
- automatizáciu.

Veľkú pozornosť venujú tvorbe univerzálnych modelov a *abstrakčným zdvihom, ktoré sú hybnou silou tvorby nových živých vedomostí a zručností*. Na záver publikácie autori zhŕňajú svoje pedagogické presvedčenie vychádzajúce z konštruktivismu takto:

- Matematické vzdelávanie bude užitočné a zmysluplné, ak bude rozvíjať a pestovať schopnosť samostatného a kritického myslenia. V dôsledky prevládajúceho formálneho prístupu sú tieto cenné príležitosti často nevyužitú.
- Matematika bude užitočná, ak bude súčasťou ľudskej kultúry a bude pomáhať riešiť každodenné problémy. Musíme opustiť predstavu, že matematika je výnimočná a hľadať ju v súvislosti s inými aspektmi bežného života.
- Matematické vzdelávanie bude mať zmysel, ak bude pestovať zvedavosť, klásť otázky a prispievať ku kritickému postoj. Nedávajme deťom odpovede na otázky, ktoré nepoložili.
- Matematika bude užitočná, ak bude rozvíjať pracovné návyky detí. Matematika môže byť hrou a tvorivou činnosťou, nemá byť drežúrou.

Aj keď niektoré problémy súvisiace s vyučovaním matematiky majú korene v špecifikách tohto predmetu, mnohé postrehy a dôkazmi podporené názory sa dajú aplikovať vo všetkých predmetoch a na informatike tým viac, že má s matematikou viaceré spoločné črty.

3 PAPERT, LOGO A DIGITÁLNY VEK: NOVÝ KONŠTRUKTIVIZMUS

Koncom 60-tych rokov 20. storočia sa odohrali prevratné udalosti nielen v hudbe, filme či architektúre, ale aj v informatike a – aj vďaka Piagetovi a Vygotskému – jej vplyve na **predstavy o vzdelávaní** detí: vizionári digitálneho veku, ako Feurzeig, Papert a Kay vyjadrili svoje presvedčenie, že digitálne technológie – nech už akokoľvek v tom čase vyzerali – patria do rúk deťom a žiakom, pretože v sebe skrývajú obrovský potenciál pre *iné učenie sa*, pre *inú školu*. Papert nás ako prvý upozornil na to, že digitálny vek sa stane **revolúciou v učení sa detí**.

V tejto kapitole sa stručne zoznámime s históriou jazyka Logo, ktoré bolo (v našej oblasti) kľúčovým produktom tohto obdobia, a výsledným efektom jeho rozšírenia do desiatok krajín sveta, ktorý dnes nazývame **logovská kultúra**.

3.1 Vznik Loga

Vedúci autorského tímu jazyka Logo W. Feurzeig popisuje v [6] vznik nového jazyka Logo v roku 1966. Oboznamuje nás aj s prvými skúsenosťami s ním, ktoré prvýkrát publikovali v správe [5]. Logo pôvodne vyvíjali v komerčnej firme Bolt, Beranek and Newman, Inc. (BBN) v tesnej spolupráci s Massachusetts Institute of Technology (MIT) ako prostriedok pre **nový spôsob vyučovania matematiky**, ktorý by využil počítače, teda najmodernejšiu techniku prístupnú v tom čase. Podľa [6] ich požiadavky na nový jazyk boli:

- *Po veľmi krátkej príprave by mali už tretiaci základnej školy zvládnuť jeho použitie pre jednoduché úlohy.*
- *Jeho štruktúra musí obsahovať dôležité matematické pojmy s minimálnou potrebou programátorských konvencií.*
- *Musta sa v ňom dať vyjadriť matematicky bohaté nenumerické aj numerické algoritmy.*

Podľa [5] očakávané prínosy použitia programovacieho jazyka ako nástroja pre pochopenie matematiky zahŕňali osvojenie si exaktného myslenia a vyjadrovania sa, špecifický spôsob prenikania do podstaty množstva zásadných matematických pojmov, ako aj použitie počítača ako *matematického laboratória*. Išlo teda o také poňatie výučby matematiky, v ktorom by žiak mohol experimentovať s matematickými pojmami vďaka počítaču a vďaka existencii jemu primeranému programovaciemu jazyku.

Keďže Logo vyvíjali v čase, keď netextová interakcia s počítačmi bola viac v rovine sci-fi než bežnou realitou, jeho prostredie je čisto textové. Z technického hľadiska je Logo syntakticky zjednodušenou verziou programovacieho jazyka LISP. Z neho zdedilo spracovanie slov a zoznamov (viet), ako aj *dialógové spracovanie príkazov*, teda to, že každý vstupný riadok sa ihneď vykoná a nie je potrebné najprv napísať ucelený program. Aj preto dostal nový jazyk názov Logo – od gréckeho slova *λόγος* (vo význame *slovo*, ale aj *vyjadrenie myšlienky* či *samotná myšlienka*).

Už v prvých implementáciách sa však rozvíjalo aj jeho použitie na riadenie robotov, a to práve reálnych robotov pripojených k počítaču, ktoré boli vybavené aj perom na kreslenie. Nazývali sa Floor Turtle (korytnačka chodiaca po podlahe), pozri [16]. Neskôr, keď to dovoľil rozvoj hardvéru, Logo pribralo aj virtuálneho robota. V spomínanej knihe² ju autor nazýva **Light Turtle** (korytnačka vytvorená svetlom, svetelným lúčom na obrazovke) a dá sa riadiť rovnako ako Floor Turtle, len príkazy vykonáva oveľa rýchlejšie a presnejšie.

Aj keď sa korytnačka na obrazovke objavila v Logu až neskôr, postupom času sa stala najznámejšou charakteristickou črtou tohto jazyka, viac pozri napr. v [26]. Kvalita zobrazenia korytnačky a ňou nakreslených obrázkov na obrazovke (korytnačia grafika) sa vyvíjali tak, ako sa vyvíjali technické možnosti počítačov. Najprv bola korytnačka trojuholník na obrazovke a vedela kresliť len čiary jednou farbou. Neskôr pribudlo viac farieb a korytnačka sa zmenila na obrázok pohybujúci sa na obrazovke (angl. *sprite*). Neskôr pribudla možnosť použiť viac korytnačiek v jednom programe, a tiež možnosť zmeniť obrázok reprezentujúci každú z nich. Moderné verzie Loga (ako napr. Microworlds Pro, Terrapin Logo či Imagine Logo) umožňujú programovať interakcie korytnačiek s počítačovou myšou, ako aj interakcie korytnačiek navzájom. Niektoré verzie Loga vystúpili z dvojrozmerného priestoru a ponúkajú možnosť trojrozmerného kreslenia a zobrazovania objektov (napr. Elica Logo).

Pre rozvoj programovacieho jazyka Logo boli dôležité ďalšie práce jeho spoluautora S. Paperta a jeho spolupracovníkov z MIT, kam sa po roku 1970 presunulo ťažisko výskumu a vývoja Loga do novovytvoreného Logo Laboratory. Papert prišiel na MIT zo Ženevy, kde spolupracoval s J. Piagetom a bol ovplyvnený jeho konštruktivistickým pohľadom na učenie sa. Vďaka tesnému prepojeniu Loga a teórie konštrukcionizmu sa začalo Logo vnímať ako *filozofia vzdelávania a neustále sa vyvíjajúca rodina programovacích jazykov, ktoré podporujú jej realizáciu*, pozri [6], kde cituje H. Abelsona.

Podľa [2] Feurzeig a Papert predstavovali dva odlišné pohľady na Logo, a to pohľad **reformistov** a pohľad **revolucionárov**. Feurzeig a jeho ďalší kolegovia z BBN v ňom videli silný nástroj na zlepšenie (reformovanie) výučby v škole. Papert a jeho blízki spolupracovníci nevideli preň miesto v súčasnom ponímaní školy, videli Logo ako možný nástroj *revolúcie v školstve*, pozri [2], str. 245-6:

...videli Logo ako počítačové prostredie, ktoré by malo podporovať deti, aby robili veci úplne novými spôsobmi.

Spomínaný presun ťažiska vývoja jazyka Logo na MIT posilnilo revolucionárov. Logo sa teda spájalo s filozofiou radikálnej reformy školy. Tak to zostalo, kým sa Logo vnímalo ako výskumný projekt a používalo sa len na niekoľkých vybraných školách. Po nástupe mikro počítačov do (amerických) škôl v prvej polovici 80. rokov a po publikovaní knihy [16] sa Logo začalo šíriť oveľa intenzívnejšie. Prispeli k tomu aj komerčné firmy, ktoré vyvinuli svoje vlastné verzie Loga. Avšak ich sľuby o radikálnej zmene v školách po zavedení ich produktu boli často nadsadené. Navyše, firmy videli trh hlavne na prvom stupni základných škôl (v zmysle slovenského vzdelávacieho systému), hoci pôvodný výskum prebiehal so siedmakmi na Junior High School (druhý stupeň ZŠ, nižšie gymnázium). Preto Logo získalo nálepku „detského programovacieho

² Viac o nej budeme hovoriť v nasledujúcej časti tejto kapitoly.

jazyka“. Stávali do popredia len jeho možnosti kreslenia, a to niekedy až do takej miery, že sa redukovalo na základnú korytnačiu grafiku.

Pre všetky tieto dôvody ako aj pre zmenu klímy v americkom školstve a pre nespĺnené sľuby nastal v druhej polovici 80. rokov istý útlm a prvotné nadšenie pre Logo pomaslo. Skupina profesora Paperta v ďalšom období zamerala svoje aktivity na pokusy s Logom vo vzdelávacích systémoch iných krajín (napr. v Brazílii, Thajsku, Argentíne, vid' [19]), jednak na prácu s Logom mimo školy [12].

V tomto období sa Logo rozšírilo aj do niektorých ďalších krajín (Veľká Británia, Austrália). Z nám bližších krajín sa v druhej polovici 80. rokov takouto „logovskou krajinou“ stalo vďaka výrobe klonov počítačov Apple II Bulharsko a vďaka rozšíreniu 8-bitových mikropočítačov americkej výroby aj Poľsko a Maďarsko. V týchto krajinách s istým oneskorením prechádzalo použitie Loga v školách podobným vývojom, ako v USA.

3.2 Papert, matematika a Mindstorms

Objav knihlače sám o sebe nestvoril poéziu. Ulahčil však jej šírenie, a tým prispel k zrodu nových básnikov. Parný stroj sám o sebe nestvoril priemyselnú revolúciu. To ľudia vyrábajú stroje. A keď človek dokáže stroje revolučne zmeniť, stávajú sa symbolom tejto zmeny.

Počítače nespôsobia revolúciu vo vzdelávaní. Zabudnime (na chvíľu) na počítače, podstatné je zmeniť kultúru! Papert nie je mechanický determinista či technokrat. Chcel by skôr prehodnotiť tradičnú štruktúru vzdelávania a použiť k tomu ako pomôcku počítač.

Bill Kerr (Game Maker community, Austrália)

S. Papert publikoval v roku 1980 svoju najvýznamnejšiu knihu *Mindstorms*³, pozri [16], v ktorej čitateľom predstrel novú víziu vzdelávania pre budúcnosť: jej základným nástrojom by sa mali stať digitálne technológie. Predstavil tu tiež programovací jazyk Logo⁴ a jeho potenciál pre poznávací proces detí – obzvlášť v matematike. V tejto kľúčovej knihe moderného vzdelávania v digitálnom veku zaviedol pojem *mikrosvety* (angl. microworlds) a dal krídla slovnému spojeniu *veľké myšlienky* (angl. powerful ideas). Mikrosvety chápe ako príležitosti, v ktorých sú veľké myšlienky fyziky, matematiky, jazykovedy a ktorejkoľvek oblasti poznávania prítomné spôsobom, ktorý umožní žiakovi naučiť sa ich, objaviť ich prirodzeným spôsobom, podobne ako sa dieťa naučí hovoriť.

Samotné Logo a jeho korytnačiu geometriu označil za prototyp takéhoto mikrosveta (pozri [16], str. 117):

... korytnačka vytvára kompletný svet, v ktorom sú niektoré otázky podstatné a iné nie. ... táto myšlienka sa dá ďalej rozvíjať a tak skonštruovať veľa takýchto "mikrosvetov", každý so svojimi vlastnými predpokladmi a obmedzeniami. Deti sa dozvedia, aké to je skúmať vlastnosti vybraného mikrosveta bez toho, aby ich vyrušovali nepodstatné otázky. Tým sa učia prenášať svoj spôsob skúmania zo svojho osobného sveta do formálnej oblasti konštruovania vedeckých teórií.

V nadpise jednej z kapitol nazýva mikrosvety *inkubátormi poznania* a demonštruje ich na príklade Dynaturtle. Je to korytnačka, ktorá okrem pozície implementuje aj rýchlosť a hmotnosť, prípadne aj vzájomné pôsobenie viacerých takýchto korytnačiek. Tento mikrosvet ponúka ako alternatívu ku klasickej výučbe Newtonovskej fyziky. Detailnejšie skúma použitie mikrosveta Dynaturtle napr. diSessa v [24].

Keď deti objavujú korytnačiu geometriu, keď vytvárajú svoje prvé vlastné príkazy, keď spoznávajú možnosti, ktoré poskytujú príkazy s parametrami, keď začnú stavať „pyramídy“ vlastných príkazov (**stvorec**, **stvorec :S**, **trojuholník**, **domcek**, **domcek :S**, **radDomov**, **radDomov :N**, **ulica...**), objavia sériu veľkých myšlienok. Takýto posun od známeho k novému je v učení sa detí kľúčový. Učia sa rešpektovať silu veľkých myšlienok. Učia sa, že *najväčšou myšlienkou je samotná myšlienka veľkých myšlienok* (str. 76).

Na záver tejto časti prezentujeme ucelenú definíciu pojmu mikrosvet, zostavenú na základe prác viacerých autorov⁵:

Čo je logovský mikrosvet

Je to starostlivo navrhnutá, otvorená, interaktívna softvérová aplikácia zameraná na učenie sa určitej témy v určitej oblasti. Takéto aplikácie vytvárajú pre deti, žiakov či študentov príležitosti pre to, aby spoznávali nové pojmy a konštruovali si nové znalosti. Keď deti alebo žiaci pracujú s mikrosvetom, rozvíjajú si poznanie prostredníctvom *aktívneho zapojenia sa a učenia sa objavovaním*. Aj keď sa na poznávací proces orientuje mnoho softvérových aplikácií, mikrosvety sú špecifické tým, že podporujú úplné *pohltenie pri učení sa*, pričom citlivo rešpektujú kognitívne a motivačné stavy učiaceho sa. Mikrosvet zvyčajne obsahuje takéto komponenty:

- súbor výpočtových alebo vizuálnych objektov, ktoré modelujú matematické či iné vlastnosti z predmetnej oblasti,
- prístup k viacsobným reprezentáciám podstatných vlastností modelu,
- spôsob, ako kombinovať objekty alebo operácie do zložitejších štruktúr (podobne ako v jazyku kombinujeme slová do viet),
- súbor aktivít a zadaní, ktoré sú zakomponované alebo naprogramované v mikrosvete. Cez ne je dieťa, žiak či študent motivovaný riešiť problémy, dosahovať ciele a pod.

³ Toto slovo je umelé, jeho autorom je S. Papert a radšej sa ani nepokúsime ho preložiť. Zostaňme na úrovni metafor a analógií: ak víchor (angl. windstorm) dokáže meniť krajinu, potom nové digitálne technológie môžu významne zmeniť spôsob a efektívnosť toho, ako sa učia žiaci. Môžu spôsobiť búrku či revolúciu v ich poznávacom procese.

⁴ V predchádzajúcej časti sme už spomenuli, že aj vďaka Papertovmu bestselleru *Mindstorms* sa Logo masívne rozšírilo do celého sveta.

⁵ V tejto definícii sme voľne využili napr. aj [4] a [23].

3.3 Logovská kultúra

Keď uvažujeme o Logu a jeho úlohe pri formovaní názorov na moderné vzdelávanie, hľadáme odpoveď na otázku, čo je také výnimočné práve na tomto programovacom jazyku. V prvom rade si uvedomme, že rozsah komunity ľudí – učiteľov, edukátorov a programátorov na celom svete – a hĺbka zmien v ich názoroch⁶ a postupoch, ktoré s používaním Loga súvisia, si vyslúžili spoločné označenie *logovský charakter*, *logovský duch*, alebo najčastejšie **logovská kultúra**⁷. Pokúsme sa teda (najmä na základe [18]) charakterizovať, čo je logovská kultúra a prečo skoro nikdy nenachádzame podobný jav okolo iných softvérových prostredí pre edukačné programovanie.

Zjednodušená odpoveď vysvetľuje logovskú kultúru ako *programovací jazyk Logo a filozofiu poznávania*. Logovská kultúra je však viac než programovanie v Logu a je viac než filozofia poznávania. To, o čo sa pokúšame v malých *logovských mikrosvetoch* alebo v celonárodných projektoch v rôznych krajinách sveta, je snaha poskytnúť žiakovi príležitosť *hľadať riešenie, ktoré funguje, vyriešiť jeho problém, vytvoriť produkt*, ktorý je dôležitý pre neho i pre druhých... Pre logovskú kultúru je kľúčové **učenie sa činnosťami** – pomocou programovacieho jazyka, zamerané na tvorbu **zmysluplného produktu**, rozšírené o premýšľanie a rozprávanie o tom, čo tvorím. Takýmto produktom môže byť grafická kompozícia (a jej zodpovedajúci program), počítačová hra, mozaika vytvorená v mikrosvete, hudobná kompozícia, animovaný „živý obraz“ a pod.

V logovskej kultúre nehovoríme o neúspechoch a omyloch, logovská kultúra **považuje omyly za dôležité** kroky v procese poznávania, za významnú príležitosť, ako lepšie pochopiť to, čo sa pokúšame urobiť alebo vyriešiť. Logisti odmietajú tradičnú predpojatosť školy starostlivo rozlišovať správne a nesprávne odpovede či riešenia. To, samozrejme, neznamená, že každé riešenie je správne. Akceptovanie omylov však má podporiť cestu k cieľu. Život totiž nie je o tom, že máme poznať správnu odpoveď, ale o tom, že máme vyriešiť určitý problém tak, aby toto riešenie fungovalo.

Pokúsme sa teraz pomocou vymedziť **základné charakteristiky** logovskej kultúry. Pomôžu nám aj pri lepšom pochopení Papertovej teórie konštrukcionizmu, pretože komentujú podstatne širší okruh problémov, než len programovanie v jazyku Logo.

- Nie Logo samotné je základným stavebným kameňom logovskej kultúry⁸, ale je ním *programovanie*.
- Deti dokážu programovať v prekvapujúco skorom veku.
- Deti ľubovoľného veku a s ľubovoľným sociálnym zázemím dokážu viac, než zvyčajne očakávame. Len im dajme do rúk vhodné nástroje a príležitosť.
- Príležitosť však znamená viac, než len prístup k počítačom. Potrebujú pracovať v *atmosfére, ktorá podporuje učenie sa* a individuálne projekty a umožňujú kontakt s *veľkými myšlienkami* (tie sme skúmali v predchádzajúcej časti).
- To znamená ešte vyššie nároky na učiteľa. Domnievame sa však, že to zároveň znamená aj zaujímavejšiu a tvorivejšiu prácu a veríme, že väčšina učiteľov uprednostňuje *tvorivosť* pred *nižšou náročnosťou*.
- Aj učiteľ však musí dostať príležitosť učiť sa. To si vyžaduje *čas a intelektuálnu podporu*.
- Podobne, ako predpokladáme, že deti dokážu viac, než sa od nich očakáva, rovnakú dôveru máme aj voči učiteľom.
- Veríme konštruktivistickému prístupu k učeniu sa.
- Ideme však ďalej: vytvárame nový konštruktivistický prístup, a to nie len k učeniu sa, ale aj k životu.
- Veríme, že žiak sa môže stať dobrým *učiacim sa*, že sa môže naučiť učiť sa. Aj učiteľ sa preto musí veľa učiť – pred žiakmi a spolu so žiakmi⁹.
- Domnievame sa, že učenie sa by malo byť užitočné pre žiaka už *teraz*, a nie až *niekedy neskôr*.
- Na to potrebujeme bohatú zbierku projektov a aktivít, v ktorých by sa každé dieťa mohlo stretnúť s veľkými myšlienkami, ktoré mu pomôžu pripraviť sa na život v 21. storočí.

Skutočný prínos jazyka Logo spočíva v troch nutných podmienkach pre vznik kultúry, ktorými sú **komunita**, **čas** a **potenciál**. Ak sa pozrieme do budúcnosti, je celkom prirodzené očakávať nové a silnejšie programovacie systémy na podporu vzdelávania. Vidíme však, ako dlho už čakáme na sformovanie novej kultúry v edukačnom programovaní. Ako nás upozorňuje Papert v [18] na str. xvi:

Takýto proces potrebuje čas. A zatiaľ všetko naznačuje, že potenciálni kandidáti na vedúce postavenie v nových kultúrach zastávajú princípy logovskej kultúry. Toto tvrdenie nevyplýva z arogantnej viery, že tvorcovia logovskej filozofie sú najmädrejší zo všetkých. Vyplýva z presvedčenia, že logovská filozofia vlastne nebola vynájdená, ale je výrazom oslobodenia procesu učenia sa od umelých prekážok pred digitálnych znalostných technológií.

Seymour Papert v [18]

⁶ Ilustrujú to napr. aj stovky monografií a článkov, ktoré skúmajú tento fenomén, a to nielen z technickej alebo edukačnej stránky, ale aj z pohľadu psychológie a filozofie poznávania.

⁷ Z mnohých možných definícií slova kultúra, pozri [10], sa nám tu asi najviac hodí táto: *Súbor zdieľaných postojov, hodnôt, cieľov a postupov, ktoré charakterizujú určitú inštitúciu, organizáciu alebo skupinu.*

⁸ Ďalej v tomto článku budeme hovoriť o relatívne novom programovacom jazyku, ktorý je prinajmenšom silným kandidátom na nasledujúci, moderný jazyk logovskej kultúry. Máme na mysli jazyk Scratch.

⁹ Túto myšlienku podporíme ešte jedným zaujímavým citátom z [18]: *Najlepší spôsob, ako sa stať dobrým tesárom, je zažiť dobré tesárčenie spolu s majstrom tesárom. Analogicky, najlepší spôsob, ako sa stať dobrým učiacim sa, dobrým žiakom, je zažiť dobré učenie sa spolu s majstrom učenia sa.* Všimnite si, že Papert nenapísal ... s majstrom učenia, ale ... s majstrom učenia sa.

4 KONŠTRUKCIONIZMUS

Papert vytvoril už v čase svojho pôsobenia v Ženeve (kde spolupracoval práve s Piagetom) originálnu víziu vzdelávania a pomenoval ju **konštrukcionizmus**, pretože vychádzal z konštruktivizmu. Zaoberal sa tým, ako by mala škola vyzerat', keby bola skutočne založená na moderných teóriách učenia sa. Kládol si otázky, ako predefinovať vyučovanie a jeho ciele v závislosti na nových informačných technológiách, ktoré sa už vtedy objavovali (treba si však uvedomiť, že sa to odohralo už pred viac ako štyridsiatimi rokmi a počítače vtedy mali veľmi ďaleko od dnešných malých a výkonných strojov s farebnými displejmi s obrovským rozlíšením). Papert bol propagátorom vtedy ešte vskutku vizionárskej myšlienky, že počítače a digitálne technológie patria do rúk deťom a môžu byť prínosné pre proces učenia sa.

Konštrukcionizmus vychádza z názoru, že ľudia sa najefektívnejšie učia vtedy, keď si svoje vedomosti sami aktívne vytvárajú – konštruujú. Ani podľa Paperta nie je možné, aby učiteľ dokázal odovzdať vedomosti žiakom, v tomto názore sa zhoduje s Piagetom aj Vygotskym.

Konštrukcionizmus je **teória učenia sa**, ale aj **vzdelávacia stratégia**. Na rozdiel od ostatných teórií učenia sa, ktoré popisujú proces získavania poznatkov a zručností iba na základe kognitívnych procesov, konštrukcionizmus berie do úvahy aj význam nadšenia pre učenie sa. Tvrdí, že žiaci sa lepšie zapoja do procesu učenia sa, ak budú pracovať na niečom, čo má pre nich osobný význam. Konštrukcionistické učenie sa podporuje aj vývin rôznych typov inteligencií a učebných štýlov, ponúka rôznorodé reprezentácie vedomostí a učiva, pozri [8].

Konštrukcionizmus sa vzťahuje k vyučovaniu, ktoré je zamerané na žiaka, využíva aktívne metódy, podporuje vlastnú tvorbu a tvorivosť. Vyučovanie by malo podporovať celkový vývin schopností dieťaťa tak, že mu poskytne produktívne prostredie a možnosti pre

- interaktívne, praktické činnosti, ktoré vychádzajú z jeho záujmov,
- objavovanie a učenie sa vytváraním (angl. *learning by making*) niečoho, čo má význam aj pre iných,
- spoluprácu, spoločné učenie sa s inými deťmi, ale aj s dospelými,
- riešenie otvorených zadaní a riešenie otvorených problémov.

Otvorené problémy a projekty sú v konštrukcionistickom vyučovaní kľúčové. Aj ich zadanie by malo vychádzať od riešiteľa – toto spĺňajú napr. aj programátorské úlohy, v ktorých si má žiak vymyslieť zadanie i riešenie nejakého problému, ktorý ho zaujme. Často sa aktivity tohto typu objavujú v **edukačnej robotike**, kde je veľká voľnosť pri vytváraní robotického modelu a pri jeho programovaní.

Konštrukcionizmus požaduje, aby sa **rola učiteľa** zmenila, aby sa stal pomocníkom žiakov na ceste k aktívnemu vytváraniu vlastných konceptov, sietí vedomostí, nadobudaniu zručností. Učiteľ by mal žiakovi pripraviť podmienky a viesť ho k objavovaniu, pomáhať mu, aby prišiel na riešenia sám. Jedine takto nadobudnuté vedomosti a zručnosti sú hodnotné a využiteľné vo svete, ktorý sa neustále mení. Učiteľ by mal byť v prvom rade sám dobrým žiakom. Iba tak sa môže dieťa od neho naučiť učiť sa.

Piaget zdôrazňoval, detailne skúmal a popisoval predovšetkým vnútornú štruktúru, organizáciu poznania a stabilitu systému nášho vnútorného poznania na rôznych stupňoch vývinu. Papert sa viac zaujímal o dynamiku tohto systému. Skúmal situácie, kedy dochádza k jeho narušeniu a posunu na ďalšiu úroveň. Zaoberal sa tým, ako môže rôzny kontext vplývať na formovanie poznania. Podľa Paperta kľúč k učeniu sa je v ponorení sa do situácie a zohľadnení rôznych premenných, ktoré ju ovplyvňujú, pozri [1].

Konštrukcionizmus – slovo s N a konštruktivizmus – slovo s V majú spoločný pohľad na proces učenia sa ako na budovanie znalostí prostredníctvom aktívnej činnosti... My konštrukcionisti ešte pridávame myšlienku, že toto sa deje zvlášť efektívne v konkrétnom kontexte, v ktorom je učiaci sa vedome a aktívne zapojení a vytvára nejaký artefakt, ktorý á význam aj pre niekoho druhého – či už to je hrad z piesku na pláži alebo teória vesmíru.

S. Papert et al., [17]

Piagetova teória prehladá dôležitosť kontextu a využiteľnosti poznania aj význam dostupného materiálu či médií pre učenie sa, rovnako neprikladá dôležitosť ani individuálnym rozdielom medzi deťmi, či rôznym štýlom učenia sa. Tieto nedostatky dopĺňa teória konštrukcionizmu.

Papert sám označil konštrukcionizmus za teóriu, ktorá presahuje konštruktivizmus: zamýšľal sa totiž aj nad obsahom vzdelávania, nad tým, **čo** (a nielen ako) **máme učiť** budúcich občanov, aby sa mohli z nich stať dospelí, ktorí budú mať pozitívny prístup k spoločnosti a budú vedieť využívať v praktickom živote to, čo sa naučili v škole. Tento politický rozmer mal v Papertovej práci dôležité postavenie – bol autorom aj niekoľkých reformných dokumentov, pozri napr. [21].

Zhrňme teraz niekoľko základných princípov konštrukcionistického učenia sa (najmä podľa [7] a [20]):

1. **Učenie sa je aktívny proces**, pri ktorom dieťa využíva svoje zmysly, na základe ktorých si v mysli konštruuje významenamy.
2. **Keď sa deti učia, učia sa aj to, ako sa učiť** – metakognitívne zručnosti sú neoddeliteľné od procesu učenia sa.
3. **Kľúčové udalosti pri konštrukcii znalostí sa odohrávajú v mysli** – aktivity musia zamestnať ruky aj myseľ.
4. **Jazyk ovplyvňuje učenie sa** – tieto dva fenomény sú neoddeliteľne prepojené.
5. **Učenie sa je sociálna aktivita** – dieťa sa učí v kontexte spoločnosti, v spojení s ostatnými ľuďmi, či už sú to učitelia, spolužiaci, kamaráti, rodina alebo komunita.
6. **Učenie sa je závislé od kontextu** – dieťa sa neučí ideálne abstraktné izolované poznatky, ale veci, ktoré súvisia s jeho životom.

7. Dieťa sa nemôže učiť bez toho, aby stávalo na svojich **predchádzajúcich vedomostiach**. Pre vytvorenie živej vedomosti je nevyhnutné, aby zapadla do existujúcej štruktúry v mysli.
8. **Učenie sa vyžaduje čas** – často sa musíme k myšlienkam vracieť, obracať ich v mysli, testovať ich a hrať sa s nimi, aby sa stali skutočne živými vedomosťami. Mali by sme sa venovať aj aktivitám, ktoré sú časovo veľmi náročné.
9. **Motivácia je kľúčovou ingredienciou učenia sa. Tento proces sa odvíja od záujmov učiaceho sa** – pokým nevieme, prečo sa učíme, nebudeme vedieť, kde takúto vedomosť využiť a stratiť pre nás zmysel.
10. **Náročná zábava** (angl. *hard fun*) a **zábavné učenie sa** (angl. *playful learning*) – úlohy by mali byť dostatočne zábavné, jednoduché a prístupné na to, aby dieťa zaujali, ale aj dostatočne ťažké na to, aby udržali jeho pozornosť a záujem.
11. Možnosť **robiť chyby a poučiť sa z nich** – deťom nezabudnime ponechať priestor na ich vlastné riešenia, pri ktorých často robia chyby, prostredníctvom dialógu s nimi prichádzajme na to, v čom je problém, a pomáhajme im s riešením.
12. **Tímová práca**, kolaborácia, rozdelenie si rolí v tíme a spolupráca na riešení – deti sa učia, ako si zorganizovať prácu v tíme, ako rozdeliť úlohy, niektoré zadania nie je možné vyriešiť bez pomoci ostatných.
13. **Učiteľia sa učia tiež** - nie je možné, aby bol učiteľ pripravený na všetky problémy, ktoré môžu nastať, tiež často rieši neznáme úlohy a učí sa nové veci spolu s deťmi.

Všimnime si, že tento zoznam bodov sa čiastočne zhoduje s podobným zoznamom, ktorý sme uviedli v predchádzajúcej časti venovanej logovskej kultúre – oba tieto javy sú neoddeliteľne prepojené a stoja za nimi ľudia, ktorí sú súčasťou logovskej kultúry, ale zároveň sú aj zástancami konštrukcionistického učenia sa.

5 SÚČASNOSŤ A BUDÚCNOSŤ EDUKAČNÉHO PROGRAMOVANIA

V predchádzajúcej kapitole sme sa dozvedeli o konštrukcionizme a princípoch konštrukcionistického učenia sa. Pripomeňme a zdôrazníme tri veci: (a) konštrukcionizmus vznikol v kontexte premýšľania o modernom učení sa matematiky, (b) jeho tvorca či tvorcovia však zdôrazňujú kľúčový presah do celého poznávacieho procesu detí, dokonca ešte viac: kým Piaget skúmal to, ako sa dieťa učí, Papert zaujíma aj to, čo sa učí a prečo. Otvorene hovorí, že konštrukcionizmus nie je iba nový prístup k učeniu sa, ale k celému životu, (c) na báze úvah o novom učení sa vznikol programovací jazyk Logo a okolo neho celá kultúra – logovská kultúra, ktorá vybuodovala a využíva (detské) programovanie ako nástroj na objavovanie veľkých myšlienok.

Nás ako didaktikov informatiky celkom prirodzene zaujíma aj tento **programátorský aspekt konštrukcionizmu**, a to minimálne z dvoch dôvodov: (a) ako absolventi „vedeckého informatického štúdia“ sme sa o Piagetovi, Vygotskom či Papertovi a ich moderných teóriách učenia sa dozvedeli cez **edukačné programovanie** a úprimne musíme priznať, že pred 30-timi rokmi sme Logo vnímali viac-menej výlučne ako programovacie prostredie, až neskôr a postupne aj ako nástroj na zmenu myslenia a nástroj na iné, nové učenie sa, (b) naďalej a v zhode s názormi spomínaných osobností ako Feurzeig, Papert, Kay, Resnick, diSessa, Kafai, Noss a ďalších sme vždy považovali a naďalej považujeme edukačné programovanie za dôležitú súčasť všeobecného vzdelania, za vynikajúcu príležitosť na rozvoj tvorivosti, logického myslenia, kolaborácie a vyšších poznávacích funkcií. A bez najmenších pochybností ako **kľúčovú súčasť moderného informatického vzdelania** na základnej a strednej škole.

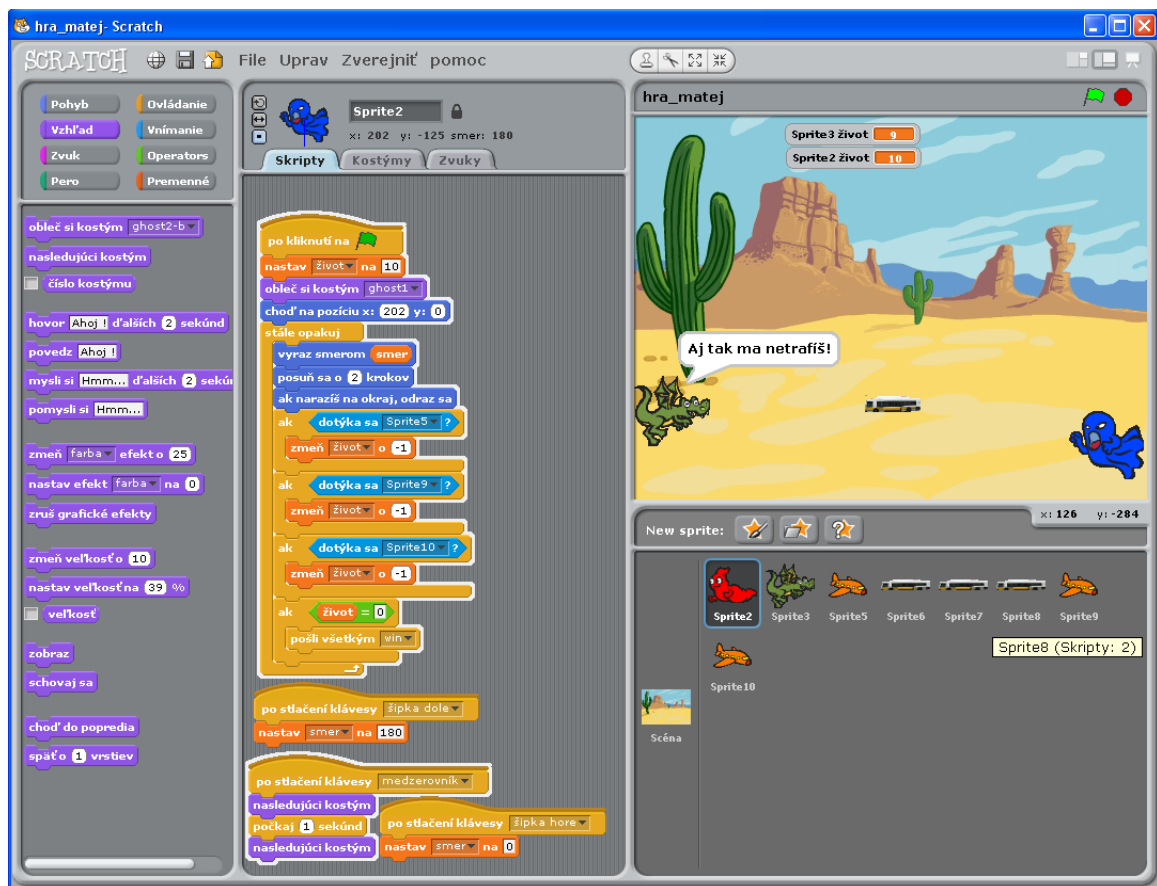
Záverčnú časť nášho pásma preto chceme venovať pohľadu na súčasnosť a budúcnosť edukačného programovania. Stručne sa zoznámime s dvoma relatívne novými programovacími prostrediami a budeme sa pýtať, kam nás posunula logovská kultúra a aká môže byť jej budúcnosť, jej pokračovanie? Ak dnes so záujmom sledujeme vznik a šírenie nových programovacích prostredí, zaujíma nás najmä to, aký je osud edukačného programovania v kontexte moderného vzdelávania a rozvoja kompetencií pre 21. storočie.

5.1 Scratch

V istom slova zmysle môžeme Scratch považovať za novú generáciu v edukačnom programovaní. V súčasnej dobe je Scratch rozšíreným programovacím jazykom u mladšej skupiny používateľov, ktorí sa s programovaním stretávajú prvýkrát. Vychádza z rovnakých myšlienok logovskej kultúry a konštrukcionizmu ako Logo, pozri [13], alebo prostredie E-toys, ktorému sa budeme stručne venovať v časti 5.2.

V porovnaní s jazykom Logo však Scratch prináša dva principiálne rozdiely. Prvý z nich nie je novátorský a poznáme ho aj z iných programovacích jazykov. Môžeme však povedať, že sa ho v jazyku Scratch podarilo úspešne rozvinúť a implementovať. Týka sa otázky syntaxe: v prostredí Scratch sa **nepotrebuje učiť syntax jazyka**, čo je pre začínajúceho používateľa z hľadiska motivácie pozitívny faktor. V tomto prostredí totiž programujeme spájaním už *hotových dielikov*, na ktorých sú zapísané príkazy alebo iné prvky programu (pozri stredný panel na obr. 1). Dieliky majú rôzne tvary podľa toho, či reprezentujú príkaz, štruktúru alebo výraz. Farba dielika závisí od kategórie, do ktorej prislúcha (*Pohyb, Vnimanie, Zvuky* a pod.). Najdôležitejšie je teda to, že tvary dielikov sú navrhnuté tak, aby sa nedali spojiť také dieliky, ktoré sa k sebe syntakticky „nehodia“.

Druhý rozdiel sa týka vlastnej filozofie projektu Scratch, konkrétne jeho **sociálneho rozmeru**. Tvorcovia jazyka Scratch chceli umožniť používateľom, aby mohli svoje projekty verejne publikovať, zdieľať a hlavne **učiť sa od seba navzájom**. Túto myšlienku realizovali prostredníctvom portálu, na ktorom vznikla v roku 2007 dnes už obrovská on-line Scratch komunita (pozri scratch.mit.edu, viac napr. aj v článku [14]). Úspešné naplnenie tohto zámeru potvrdzuje viac ako 120 000 registrovaných používateľov a priemerne 1 500 uploadovaných projektov denne. K marcu 2011 bolo na Scratch portáli dostupných spolu viac ako 1 600 000 projektov z celého sveta.



Obr. 1 Okno programu Scratch

Vráťme sa však späť k filozofii tohto projektu. Papert prišiel s myšlienkou, podľa ktorej by programovacie prostredie malo mať **nízky prah** – pre začínajúceho používateľa by malo byť jednoduché doň „vstúpiť“, a **vyšoký strop** – zároveň by malo umožňovať vytvárať stále komplexnejšie a náročnejšie projekty, pozri [22]. Tvorcovia jazyka a prostredia Scratch sa rozhodli k týmto požiadavkám pridať ešte jednu, a to požiadavku **širokých stien** – možnosť vytvárať bohatú škálu rôznorodých projektov, ktoré by oslovili žiakov s rozdielnymi záujmami a s odlišnými štýlmi učenia sa. Túto požiadavku autori naplňajú aj cez možnosť práce s multimédiami – do projektu vieme jednoducho importovať obrázky, zvuky alebo hudbu. Scratch má tiež vlastný editor kreslenia a nástroj na nahrávanie zvuku.

Pre nás sú zaujímavé tie vlastnosti jazyka a prostredia Scratch, ktoré mu podľa jeho tvorcov majú pomôcť oslovit' a zaujať široké spektrum detí a mladých ľudí a motivovať ich k tomu, aby prostredníctvom programovania pracovali s **programovateľnými médiami** (interaktívnymi animáciami, príbehmi, obrázkami, video, hrami a pod., pozri [15]). Týmto kľúčovým vlastnostiam jazyka a prostredia Scratch sa vo svojom článku venujú v [13]. Na tomto mieste sa ich v niekoľkých bodoch pokúsime stručne zhrnúť¹⁰.

- Prostredie programu Scratch tvorí **len jedno okno**, ktoré je **používateľovi stále k dispozícii**. V tomto prostredí nenájdeme žiadne iné „plávajúce“ okná alebo palety – kľúčové komponenty tak ostávajú pre používateľa stále viditeľné. Tvorcovia sa snažili minimalizovať počet takých okien, ktoré sa zobrazujú až na požiadanie (napr. editor kreslenia). Najpoužívanejšie dieliky sa nachádzajú poruke – navrchu palety (v ľavej časti okna programu).
- Scratch síce rozlišuje fázu editovania a behu programu, ale rozdiel medzi nimi je veľmi malý. Navyše, používateľ môže aj počas editovania skúšať beh malých kúskov programu. Môže tak napr. jednoducho skúmať a objavovať nové dieliky, ktoré zatiaľ nepozná – po kliknutí na konkrétny dielik sa vykoná zodpovedajúci príkaz. Dielik pritom nemusíme ani preniesť do časti scenárov objektu, stačí naň kliknúť rovno na palete, kde je umiestnený. Dieliky s parametrami majú prednastavenú konkrétnu hodnotu, aby mohol používateľ bezprostredne objavovať ich funkcionálnosť. Hodnotu, ktorú vráti niektoré dieliky (funkcie), môžeme zobrazovať napríklad pomocou „bublín“ (podobne ako v komikse, viď Obr. 1).

Rôzne **parametre príkazov** (napr. otoč sa o x stupňov) **vieme meniť** aj počas ich vykonávania v programe – objekty na túto zmenu dynamicky reagujú. Učiaci sa tak môže rýchlejšie a lepšie porozumieť daným princípom a spojitostiam

¹⁰ Pre lepšie pochopenie je dobré si ozrejmiť: v tomto prostredí pracujeme výlučne s objektmi (či už je to samotná scéna alebo postavy či predmety umiestnené na nej). Ku každému objektu prislúchajú jeho **scenáre** (stredný panel na obr. 1) a **premenne**. Zaujímavosťou je vyriešená komunikácia medzi objektmi – v scenári jedného objektu nemôžeme priamo „oslovit“ iný objekt, vieme ale poslať konkrétnu správu všetkým ostatným objektom. Po prijatí správy môže iný objekt aktivovať svoj vlastný scenár.

a vytvoriť si lepší mentálny model problému. Rovnako môžeme kedykoľvek odobrať alebo pridať dielik do práve spusteného programu, čo má bezprostredný vplyv na jeho správanie.

Tvorcovia nechceli, aby používatelia najprv „vybudovali“ program alebo projekt a až hotový ho potom spúšťali a testovali. Naopak, zámerom tvorcov bolo, aby projekt **budovali a zároveň neustále skúšali a overovali**.

- Program alebo podprogram, ktorý sa práve vykonáva, je zvýraznený bielym orámovaním (viď stredný panel na Obr. 1). Používateľ tak môže zistiť, kedy a ako dlho sa daný program vykonáva. Ak sa v programe vyskytuje chyba (napr. delenie nulou), farba orámovania sa zmení na červenú a konkrétny dielik sa rovnako orámuje na červeno. Program tiež môžeme krokovať výberom možnosti z menu programu, vtedy príkaz (dielik), ktorý sa práve vykonáva, svieti žltou farbou.
- V programe Scratch sa **nepoužívajú žiadne chybové správy**. Program vzniká skladaním už hotových dielikov s rôznymi príkazmi do väčších blokov. Dieliky rôznych skupín (výraz, štruktúra alebo príkaz) majú odlišný tvar, dajú sa teda spojiť len v prípade, že je to formálne správne. Tvorcovia sa snažili vytvoriť jazyk, v ktorom bude oveľa jednoduchšie zostaviť program, ktorý „niečo“ vykonáva, ako program, ktorý je nefunkčný. Aj v prípade „zlého“ (nekorektného) vstupu sa program snaží vykonať niečo zmysluplné (napr. veľkosť objektu sa môže pohybovať len v rozumných hraniciach, a to aj v prípade, že zadáme číslo mimo týchto stanovených hraníc). Takýto prístup samozrejme nevylučuje chyby. Ak chceme vytvoriť zmysluplný projekt, sme nútení rozmýšľať, v každom prípade je to spôsob, ako sa vyhnúť poklesu motivácie učiaceho sa, ktoré môže byť spôsobené častým hlásením chýb.
- V tomto prostredí sa pracuje s premennými podobne ako v jazyku Boxer (pozri napr. [25]) – premenné sú reprezentované konkrétnymi objektmi, s ktorými môžeme manipulovať. Hodnotu každej premennej (aj zoznamu) vieme sledovať prostredníctvom jej vlastného monitora, ktorý je umiestnený na scéne (viď pravý horný panel na Obr. 1). Môžeme tak napr. pozorovať, ako sa zmení hodnota premennej po vykonaní konkrétneho príkazu (napr. *zmeň x o -2*).
- **Množina príkazov je minimalizovaná**, ale iba do tej miery, aby používateľ aj tak mohol vytvárať široké spektrum rôznorodých projektov. Každý nový dielik zaberá ďalšie miesto na obrazovke a znamená pre používateľa potencionálne dlhšie hľadanie. Pri výbere dielikov do palety príkazov preto tvorcovia každú voľbu dobre zvážili.

Aj keď je v súčasnosti Scratch značne populárny a mohol by znamenať istý prelom v edukačnom programovaní, iba budúcnosť ukáže, či sa práve on stane nositeľom či pokračovateľom logovskej kultúry. V každom prípade už dnes môžeme konštatovať, že autori tohto moderného nástroja robia všetko preto, aby sa tak stalo.

5.2 Alan Kay a Squeak Etoys

V tejto časti nášho príspevku sa budeme venovať detskému programovaciemu jazyku s trocha dlhšou históriou – prostrediu Squeak Etoys (či skráteno iba Etoys). Vedúcim jeho autorského kolektívu je ďalší významný vizionár druhej polovice 20. storočia – A. Kay. V roku 1968 sa ako mladý výskumník zoznámil s Papertom a programovacím jazykom Logo, odvtedy študoval diela Piageta, Vygotského, Brunera a ďalších konštruktivistov a konštrukcionistov. Od roku 1970 pracoval vo výskumnom ústave PARC Xerox, kde sa podieľal na vývoji programovacieho jazyka Smalltalk. Prispel k rozpracovaniu myšlienky **objektového programovania**, je autorom alebo spoluautorom prvého grafického počítačového rozhrania a grafických okien. Kaya považujú za jedného z autorov myšlienky **mobilného vzdelávania**. Jeho najslávnejším projektom je **Dynabook** z konca 60-tych rokov – predstava, v ktorej už pred viac ako 40 rokmi načrtnol dnešný tablet PC.

Prostredie Squeak Etoys je postavené, rovnako ako Scratch, na textovom programovacom jazyku Squeak. Na rozdiel od Scratchu, v ktorom môžeme na prvý pohľad rozpoznať Logovské dedičstvo, Etoys viac odzrkadľuje svoju implementačnú platformu – jazyk Squeak. Etoys teda možno chápať ako grafickú nadstavbu jazyka Squeak. Program vytvárame z pravouhlých kartičiek a textovo ich dopisujeme len v prípade potreby – ak špecifikujeme argumenty, mená nových procedúr a premenných a pod.

Ľavá časť obrázku 2 ukazuje popis objektu Car (väčšie okno) a jednu jeho metódu (menšie okno). Objekt Car je červené autíčko, ktorého metóda *script1* mu káže chodiť po obrazovke a odrážať sa od inej než bledomodrej farby. Vidíme, že prostredie Etoys pôsobí „dospelejším“ a zložitejším dojmom než Scratch. Horný okraj oboch okien procedúry obsahuje množstvo ikon, pomocou ktorých môžeme dosiahnuť mnohé funkcie prostredia. Kompletný popis objektu Car pôsobí na prvý pohľad dosť zložito. Vidíme aj to, že v programe explicitne oslovujeme objekty (Car forward by...). Etoys teda, na rozdiel od základnej verzie prostredia Scratch, umožňuje v metóde jedného objektu vyvolať metódu iného objektu.

V článku [9] vyjadril Kay (podľa www.squeakland.org/about/people/ je členom "Etoys Vision Team") základnú myšlienku prostredia Etoys takto:

Poskytnúť a rozšíriť väčšinu počítačových potrieb pomocou jednoduchého a silného používateľského rozhrania, ktoré „beží všade a na všetom“ a umožňuje koncovým používateľom „pristupovať ku všetkému a meniť všetko“.

Princíp *meniť všetko* ilustruje pravá časť obrázku 2.



Obr. 2 Vľavo demo prejekt prostredia E-toys (popis v texte). Obrázok vpravo je prevzatý z [9] a ukazuje, že aj okienka procedúr sú rovnaké vizuálne objekty ako autíčka: aj im môžeme predpísať pohyb a otáčanie

5.3 Kam smeruje edukačné programovanie

V predchádzajúcich častiach sme prezentovali dve zaujímavé programovacie prostredia, ktoré celkom určite patria – spolu s ďalšími jazykmi – medzi kandidátov, ktorí chcú ďalej niest' zástavu logovskej kultúry a byť na čele edukačného programovania. Aj keď v jednotlivých krajinách a komunitách existujú rôzne prostriedky a rôzne stratégie v edukačnom programovaní, na prezentáciu sme vybrali dva významné a „mienkotvorné“ prístupy: jeden pochádza z „rodiska“ Loga a teórie konštrukcionizmu, druhý vznikol na západnom pobreží USA, v kolektíve, ktorý vedie ďalší z kľúčových vizionárov myšlienky počítača ako nástroja na rozvoj myslenia v rukách detí:

Ešte zaujímavejšie však bolo premýšľať o tom (pred 40-timi rokmi, dodávajú autori), čo nové vlastne prináša počítač pre učenie sa. Veril som, že tak ako tlač postupne zmenila spôsob komunikácie medzi ľuďmi, aj počítač zmení spôsob, ako ľudia objavujú, testujú a prezentujú svoje myšlienky. Prirovnal som počítač k hudobnému nástroju: je to nástroj, ktorého tónmi sú myšlienky.

A. Kay v rozhovore z roku 2002 o histórii myšlienky Dynabook, pozri www.squeakland.org/resources/articles/article.jsp?id=1007

Napriek týmto zaujímavým projektom v kontexte edukačného programovania sme niekedy svedkami určitej skepsy, niekedy sa dokonca hovorí o *kríze edukačného programovania*. Vidíme, že skutočne komplexné a silné prostredia pre edukačné programovanie (ako napr. Boxer, pozri [25], ale tiež moderné implementácie jazyka Logo a čiastočne aj Etoys) sa presadzujú len ťažko a často ich používajú iba výskumníci alebo malé komunity detí a ich učiteľov. Zamyslime sa v tejto záverečnej časti nad tým, aké problémy a aké obavy sa v tomto kontexte zvyknú uvádzať:

- Možno vznikli prílišné očakávania, ktoré nasledovalo určité sklamanie.
- Možno je programovanie v jeho plnom rozsahu príliš zložitú pre nasadenie vo všeobecnom vzdelaní a jedinou cestou je niečo z neho obetovať.
- Možno sa ešte nenašlo to „pravé“ detské programovacie prostredie, ktoré by sprostredkovalo aj menej triviálne pojmy prístupnou formou (premenné, parametre, rekurziu, zoznamy ...).
- Možno majú z programovania obavy skôr učители než žiaci, možno nie sú pre túto kompetenciu dobre pripravení. Všimnime si napríklad, aké problémy robí učiteľovi, ktorý sa učil programovať v „pred objektovej“ ére, preorientovať sa na moderné objektové programovacie prostredie.

Medzi učiteľmi sú populárnejšie také produkty, ktoré by sme mohli zaradiť niekde medzi dve krajné polohy, kde na jednom konci sú uzavreté a relatívne jednoduché mikrosvety bez možnosti ich zmeny (ako napr. Mravec či Baltík), a na druhom konci zjednodušené prostredia, ktoré si udržia otvorenosť (len) za cenu obetovania niektorých dôležitých informatických konceptov (ako napr. Scratch či Turtle Art). Niekde medzi týmito polohami nájdeme aj prostredie Živý obraz zo vzdelávania v projekte DVUi.

Klasické textové programovanie pomocou klávesnice je v týchto moderných prostrediach zvyčajne nahradené nejakým grafickým spôsobom budovania programu: od skladania textu na kartičkách či dielikoch (na ktorých je ešte stále čitateľný text programu) po čisto ikonické programovanie (kde sú príkazy nahradené obrázkami, ktoré sú jasné a zrozumiteľné pre autorov, často však už menej pre používateľov), prípadne určitú kombináciu týchto dvoch prístupov.

Za prostredie, ktoré má v súčasnosti jasnú prevahu v dynamike svojho rozvoja a šírenia, môžeme smelo označiť Scratch. Len budúcnosť ukáže, či sa jeho rozšírením podarí naplniť myšlienky Paperta, diSessu či Kaya (po ich miernom zrealnení vďaka stretu s realitou) alebo nie. Veľa nám v blízkej budúcnosti napovie aj to, či sa v pripravovanej novej verzii autorom tohto prostredia podarí do prostredia pridať niektoré ďalšie ale kľúčovo dôležité abstrakcie (ako procedúry, resp. procedúry s parametrami), a to bez ujmy na jednoduchosti jazyka, ako sľubujú v [13].

ZÁVER

V našom príspevku sme stručne a skrátene prešli dlhú cestu od Piagetových výskumov detského poznávania, cez Vygotského a Paperta, až po súčasné programovacie prostredia, ktoré sa snažia vložiť deťom do rúk počítač a programovacie nástroje schopné pomáhať im pri poznávaní, pri objavovaní sveta a veľkých myšlienok. Predstavili sme hlavné charakteristiky konštruktivizmu a prečítali sme si, ako ho v súčasnosti vnímajú poprední didaktici matematiky. Zoznámili sme sa s históriou jazyka Logo a kultúrou, ktorá vznikla okolo komunity učiteľov a vedcov – logistov a priekopníkov využívania digitálnych technológií na učenie sa činnosťami, na tvorbu užitočných výstupov, na poznávanie a tvorbu. Zoznámili sme sa s Papertovou teóriou konštrukcionizmu, ktorá rozšírila Piagetov konštruktivizmus a všima si aj to, čo sa deti učia, prečo, kde a s kým, čo tvoria a čo objavujú – napr. aj pomocou digitálnych technológií.

Uvedomujeme si, že po slávnych a významných V- a N- teóriách učenia sa detí prídu (musia prísť) aj ďalšie. Na Papertovi napr. cítime, že neobsiahol vplyv internetu a sociálnych sietí na proces učenia sa. Ale jeho posun od konštruktivizmu k logovskej kultúre má obrovský význam a ovplyvnil a natrvalo tvaroval náš pohľad na moderné vzdelávanie v digitálnom veku.

Uvedomujeme si tiež, že ak chceme kvalifikovane premýšľať napr. o súčasnosti a budúcnosti edukačného programovania, musíme prehlbovať naše poznanie východísk a koreňov, musíme sa vracieť ku „klasikom“ našej oblasti a pokúšať sa porozumieť, ako uvažovali a kam hľadeli. Bez toho nemôžeme zodpovedne premýšľať o poznávanom procese detí a škole v digitálnom veku. A to, že naše predchádzajúce názory musíme prehodnotiť, nás naučil Seymour Papert.

LITERATÚRA

- [1] ACKERMANN, E. Constructivism(s): Shared roots, crossed paths, mutiple legacies. CLAYSON, J. E. – KALAŠ. I. (eds.) *Proceedings of Constructionism 2010 – 12th European Logo Conference*. [CD-ROM]. Bratislava : Comenius University, in association with The American University of Paris, 2010. ISBN 978-80-89186-66-2.
- [2] AGALIANOS, A. – WHITTY, G. & NOSS, R. The social shaping of Logo. *Social Studies of Science*, 36, pp. 241 – 267. 2006 [cit. 2010-03-10]. Dostupné na: citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.94.4204&rep=rep1&type=pdf
- [3] CLAYSON, J. E. – KALAŠ. I. (Eds.) *Proceedings of Constructionism 2010 – 12th European Logo Conference*. [CD-ROM]. Bratislava : Comenius University, in association with The American University of Paris, 2010. ISBN 978-80-89186-66-2.
- [4] EDWARDS, L. D. Microworlds as Representations. In: diSessa, A. A. – Hoyles, C. – Noss, R. – Edwards, L. D. (eds.) *Computers and Exploratory Learning*. Springer, 1995, p. 127 – 154. ISBN 978-3540592020.
- [5] FEURZEIG, W. et al. 1969. *Programming-Languages as a Conceptual Framework for Teaching Mathematics*. Final Report on the First Fifteen Months of the LOGO Project : Bolt, Beranek and Newman, Inc., Cambridge, MA. 1969. [cit. 2010-03-10]. Dostupné na: eric.ed.gov/ERICDocs/data/ericdocs2sql/content_storage_01/0000019b/80/33/6c/11.pdf
- [6] FEURZEIG, W. Toward a Culture of Creativity: A Personal Perspective on Logo's Early Years, Legacy and Ongoing Potential. In *Eurologo 2007 Proceedings*, Bratislava 2007. [cit. 2010-03-10]. Plná verzia prístupná len on-line www.eurologo2007.org/proceedings/KP-Feurzeig.pdf
- [7] HEIN, G. E. *Constructivist Learning Theory*. The Museum and the Needs of People, CECA (International Committee of Museum Educators) Conference, Jerusalem Israel, 15 – 22 October 1991. Dostupné na: <http://www.exploratorium.edu/IFI/resources/constructivistlearning.html>
- [8] KAFAI, Y. – RESNICK, M. (eds.) *Constructionism in practice*. Mahwah, New Jersey : Lawrence Erlbaum Associates, 1996. ISBN 0-8058-1985-1.
- [9] KAY, A. *Squeak Toys Authoring & Media*. 2010. Dostupné len on-line: www.squeakland.org/content/articles/attach/etovs_n_authoring.pdf
- [10] KROEBER, A. – KLUCKHOHN, C.: *Culture: A Critical Review of Concepts and Definitions*. New York, N.Y : Vintage Books, 1952.
- [11] KUŘINA, F. – HEJNÝ, M. *Dítě, škola a matematika*. Praha : Portál, 2009. ISBN 978-80-7367-397-0.
- [12] MALONEY, J. – BURD, L. – KAFAI, Y. – RUSK, N. – SILVERMAN, B. – RESNICK M. Scratch: A Sneak Preview. In *Second International Conference on Creating, Connecting, and Collaborating through Computing*. pp. 104 – 109. Kyoto, Japan, 2004. [cit. 2010-03-10]. Dostupné na: llk.media.mit.edu/papers/ScratchSneakPreview.pdf
- [13] MALONEY, J. – RESNICK, M. – RUSK, N. – SILVERMAN, B. – EASTMOND, E.: The Scratch Programming Language and Environment. In: *ACM Transactions on Computing Education*, Vol. 10, No. 4, Article 16, November 2010.
- [14] MIKOLAJOVÁ, K. *Vytváranie hier v prostredí Scratch - cesta k programovaniu na II. stupni ZŠ*. Rukopis. Článok bol prijatý na konferenciu DidInfo 2011. Banská Bystrica: Univerzita Mateja Bela, 2011.
- [15] MONROY-HERNÁNDEZ, A. – RESNICK, M. Empowering Kids to Create and Share Programmable Media. In: *Interactions – Pencils before pixels: a primer in hand-generated sketching*, Magazine, Vol. 15 Issue 2, 2008.
- [16] PAPERT, S.: *Mindstorms. Children, Computers And Powerful Ideas*. Basic Books, New York 1980. ISBN 0-465-04627-4.
- [17] PAPERT, S. – HAREL, I. Situating Constructionism. PAPERT, S. – HAREL, I.: *Constructionism*. Norwood, NJ : Ablex Publishing, 1991. ISBN 0893917869. s. 1 – 12. Dostupné na: www.papert.org/articles/SituatingConstructionism.html
- [18] PAPERT, S. What is Logo? And Who Needs It? In: *Logo philosophy and implementation*. Highgate Springs, Vermont: Logo Computer Systems Inc., 1999. ISBN 2-89371-494-3.
- [19] PAPERT, S. et al. *Logo Philosophy and Implementation*. Highgate Springs, Vermont: Logo Computer Systems Inc., 1999. ISBN 2-89371-494-3.

- [20] PAPERT, S. 1999. The Eight Big Ideas of the Constructionist Learning Laboratory. Nepublikovaný interný dokument. South Portland, Maine, 1999. Citované v STAGER, G. Papertian Constructionism and the Design of Productive Contexts for Learning. Plenary Session Paper – EuroLogo X, Warsaw, Poland, 2005.
Dostupné na: <www.stager.org/articles/eurologo2005.pdf>
- [21] PAPERT, S. – CAPERTON, G. *Vision for Education: The Caperton-Papert Platform*. An essay for 91st annual National Governors' Association meeting held in St. Louis, Missouri in August of 1999.
- [22] RESNICK, M. Scratch: Programming for All. *Communications of the ACM*, Vol. 52 No. 11, p. 60 – 67, 2009. ISSN:0001-0782.
Dostupné na: <<http://web.media.mit.edu/~mres/papers/Scratch-CACM-final.pdf>>
- [23] RIEBER, L. P. Microworlds. In: Jonassen, D. (ed.). *Handbook of Research for Educational Communications and Technology*. 2nd edition. Mahwah, N. J. : Lawrence Erlbaum Associates, 2004. p. 583 – 603. ISBN 0805841458.
- [24] diSESSA, A. Unlearning Aristotelian physics: A study of knowledge-based learning. *Cognitive Science*, 6, pp. 37 – 75, 1982 [cit. 2010-03-10]. Dostupné na: <<http://csjarchive.cogsci.rpi.edu/1982v06/i01/p0037p0075/MAIN.PDF>>
- [25] diSESSA, A. *Changing Minds: Computers, Learning, and Literacy*. MIT Press, 2000.
- [26] TOMCSÁNYI, P.: *Textové a grafické rozhrania pre vývoj malých edukačných aplikácií*. Pripravovaná dizertačná práca, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského, 2011.

AUTORI

KALAŠ, IVAN, PROF., RNDR., PHD.

Katedra základov a vyučovania informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského v Bratislave
Mlynská dolina, 842 48 Bratislava
kalas@fmph.uniba.sk

KABÁTOVÁ, MARTINA, PAEDDR., PHD.

Katedra základov a vyučovania informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského v Bratislave
Mlynská dolina, 842 48 Bratislava
kabatova@fmph.uniba.sk

MIKOLAJOVÁ, KATARÍNA, MGR.

Katedra základov a vyučovania informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského v Bratislave
Mlynská dolina, 842 48 Bratislava
mikolajova@fmph.uniba.sk

TOMSCÁNYI, PETER, RNDR.

Katedra základov a vyučovania informatiky, Fakulta matematiky, fyziky a informatiky, Univerzita Komenského v Bratislave
Mlynská dolina, 842 48 Bratislava
TOMSCANYI@FMPH.UNIBA.SK