

VYBRANÉ ASPEKTY VZDĚLÁVÁNÍ V OBLASTI PROGRAMOVÁNÍ

KUBRICKÝ Jan, ČR

Resumé

Příspěvek se zabývá vybranými aspekty vzdělávání v oblasti programování. Analyzuje vybraná témata příslušných odborných publikací a studia na pomezí informálního a neformálního vzdělávání.

Klíčová slova: vzdělávání, ICT, programování, odborné publikace.

THE CHOSEN ASPECTS OF EDUCATION IN THE AREA OF PROGRAMMING

Abstract

The paper deals with the chosen aspects of education in the area of programming. The autor analyse topics of technical publications and study in informal education.

Key words: education, ICT, programming, technical publications.

1 Úvod

Jedním z charakteristických znaků vzdělávání v oblasti ICT je zvyšující se míra pronikání do pomezí informálního a neformálního vzdělávacího prostředí. Prostředky ICT se stávají díky nepřetržitému vývoji běžným uživatelům přístupnější a existuje celá řada odborných publikací věnujících se tématům - jak tyto prostředky ovládnout a využít pro řešení běžných či méně tradičních situací. Díky těmto publikacím je možné získat komplexní poznatky z nejrůznějších oblastí moderních ICT a dosáhnout, nebo se minimálně přiblížit úrovni znalostí získaných prostřednictvím formálního vzdělávání. V tomto smyslu jsou v popředí zájmu oblasti programování a tvorba webových aplikací (v této stati budeme tvorbu webových aplikací uvažovat jako jednu ze součástí oblasti programování). Zastáváme názor, že právě snadná uchopitelnost obsahu při studiu (odborné publikace) a aplikaci (dostupná vývojová prostředí) programování je jedním z důvodů, že se tato oblast stále více etabluje jako součást vzdělávání na rozhraní informálního a neformálního prostředí.

2 Odborné publikace

Publikace věnující se daným tématům ICT jsou z výše naznačeného úhlu pohledu velmi ceněné a žádané, a dle charakteristiky jejich zpracování je můžeme pracovní rozdělit do čtyř hlavních kategorií (v celku respektující celistvost výkladu dané problematiky):

1. **Základní výkladové příručky** – zaměřeny od základů na co nejširší možný výklad všech prvků, vlastností a funkcí příslušné problematiky.
2. **Dílčí výkladové příručky** – podrobně zaměřeny na dílčí část příslušné problematiky.
3. **Multi-výkladové příručky** – zaměřeny na komplex témat příbuzných systémů a jejich vzájemného využití.
4. **Aplikační příručky** – obsahující primárně výklad konkrétních aplikací příslušné problematiky, a to jak samostatně nebo i z pozice propojení příbuzných témat.

V poslední době se velkému zájmu těší zejména aplikační příručky, které čtenářům předkládají hotové techniky k okamžitému použití. Publikace již v názvu obsahují hesla typu *praktické příklady, hotová řešení, koncepty a techniky, konkrétní využití*. Aplikační příručky jsou určeny zejména pokročilejším čtenářům, obeznámeným se základy dané problematiky. Ale i přesto se autoři těchto publikací snaží prezentovat nezbytné začátečnické minimum, na něž okamžitě navazují praktické příklady. V hierarchii systematičnosti studia jsou publikace tohoto typu v našem pracovním rozdělení na čtvrtém místě, obrazně na nejvyšším stupni pyramidy poznávání v této oblasti - jak ilustruje obr. 1.

V rámci formálního vzdělávání přichází studium aplikačních příruček na řadu většinou až jako poslední, po osvojení znalostní báze základních prostředků celé problematiky. Vycházíme tak z didaktické zásady systematičnosti, která se podle M. Kurelové (3) realizuje tehdy, když ve výuce postupujeme od snazšího k náročnějšímu. V programování se opíráme o základní schémata vztažená z matematiky, příslušného programovacího jazyka a postupně prostřednictvím složitějších prvků, jako jsou podmínky, cykly a funkce budujeme konkrétní aplikace.



Obr. 1: Hierarchie odborných publikací.

Na pomezí neformálního a informálního vzdělávání jsou ale aplikační příručky většinou první, nikoli jednou z posledních typů publikací, ke kterým se zájemci o problematiku v počátcích svého studia uchylují. Zkoumají možnosti využití, vzorové příklady a vstřebávají pouze minimální teoretické znalosti nezbytné pro aplikaci praktických příkladů. Teprve posléze při potřebě hlubšího poznání se vrací k výkladovým příručkám a prohlubují nabyté znalosti. Mnohé indicie z praxe nám napovídají, že se tento postup obecně nemusí nutně jevit jako nesprávný. Ve výuce objektově orientovaného programování nachází své opodstatnění.

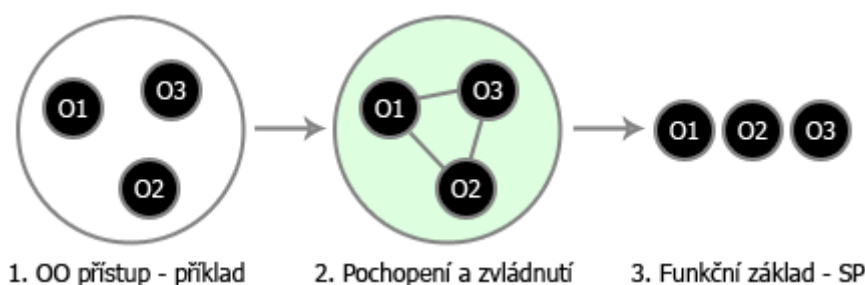
3 Objektově orientované programování

Naznačený trend je patrný ve výuce Objektově Orientovaného Programování (dále jen OOP). OOP lze vlastně považovat za průřezové téma programování, neboť velká část programovacích jazyků je orientována objektově a principy OOP jsou tak přenositelné mezi více vývojových prostředí.

Podstata OOP spočívá v síti volně komunikujících objektů, nicméně funkční základ je budován prostřednictvím klasického strukturovaného programování. Strukturované programování (založeno na provádění posloupnosti příkazů metodou „shora-dolů“) tak není

technikou OOP úplně vytlačeno, pouze získalo novou, progresivní úlohu (6). Navíc mnohé programovací jazyky teprve otevírají své objektové možnosti. Taktéž zůstává, že strukturované programování se v mnoha vývojových prostředích stále hojně využívá odděleně a neztratilo téměř nic ze své účelovosti.

V tradičním modelu výuky programování „od základů k příkladům“ objevil se ale problém, který spočívá v potížích naučit se a pochopit OO přístup v momentě, kdy jsou vzdělávaní již schopni programovat a myslet prostřednictvím strukturovaného programování. Stejně tak v případě, kdy se výuka nejprve věnuje základům strukturovaného programování, přičemž objektový princip přichází na řadu až posléze. Čerpáme zde poznatky z praxe a z dlouholetých zkušeností a názoru předního odborníka-pedagoga R. Pecinovského, který se metodikou výuky OOP důkladně zabýval (viz. statě 1,2). Podstata jeho myšlenky tkví v rozdílném přístupu ve výuce a pokouší se jej ilustrovat obr. 2. Od samého začátku vštěpovat pomocí snadných příkladů objektově orientovaný model (**bod 1**), bez participování na výkladu strukturovaného programování. Funkční prvky strukturovaného programování - SP (**bod 3**) přicházejí na řadu až po pochopení a zvládnutí OO přístupu (**bod 2**).



Obr. 2: Přístup výuky OOP.

Ve výuce OOP R. Pecinovský v kontextu s odbornými publikacemi nepřímě zavrhuje započítí studia od základních výkladových příruček. Stejně tak postup výuky a výklad učitele. „*Nepříjemnou vlastností takto koncipovaného výkladu je, že studenti na počátku vstřebají základy strukturovaného programování, ty v jejich mysli zakoření, takže objektové rysy, s nimiž se seznámí v další části kurzu, se pak snaží naroubovat na tento strukturovaný kmen. Osvojení objektového myšlení jim pak většinou trvá výrazně déle, než kdyby s objekty pracovali ihned a nové poznatky usazovali do čistého, předchozími programátorskými zkušenostmi nepoznamenaného prostoru*“ (1, s.1).

4 Teoretický rozbor

Budeme-li uvažovat předchozí zkušenosti vzdělávaných, je zřejmé, že ačkoli je strukturované programování základním a funkčním prvkem OO přístupu a je z hlediska systematickosti výkladu základním prvkem od kterého by se mělo dále přecházet ke složitějšímu, je potřeba se v počátku výuky od něj úplně oprostit. Představuje v této fázi poznávání určitý negativní článek, komplikující pochopení nového.

Aktivizace minulých zkušeností při poznávání nového souvisí s pojmem *transfer učení* (4). Transfer učení je v odborné literatuře dále rozlišován na pozitivní a negativní (tzv. interferenci). Pozitivní transfer dle J. Průchy (4) vzniká tehdy, když se nově nacvičovaný

materiál nebo činnost podobá předcházejícím. V programování je např. patrný při žádoucím přenosu poznatkových schémat vztažených z matematiky, konkrétně např. matematické logiky. V případě negativního transferu minulá zkušenost komplikuje zapamatování a pochopení nového. K snížení interference je potřeba důkladně promýšlet koncepci výuky (5). Výuka OO přístupu v programování by měla být orientována na příklady, model reálného světa objektů a jeho asociaci se světem OOP, čímž jsme se zabývali např. ve stati (6).

5 Závěr

Předložená teorie, která se pokouší do oblasti formálního vzdělávání OOP implementovat nové trendy na rozhraní informálního a neformálního vzdělávání byla (1), (2), a je v maximální možné míře podrobena zátěži v praxi. Čerpání námětu z rozhraní informálního a neformálního vzdělávání s sebou nese určité riziko, pramenící často z empiricky nepodložených argumentů. Nicméně při hlubší analýze lze v některých oblastech tyto náměty dále zkoumat a rozvíjet. Stejně tak se jeví vhodné podrobně se zabývat obsahem a zpracováním příslušné odborné literatury.

6 Literatura

1. PECINOVSKÝ, R. Jak při výuce Javy opravdu začít s objekty [online]. <URL: http://vyuka.pecinovsky.cz/prispevky/Jak_pri_vyuce_zacit_s_objekty.pdf>.
2. PECINOVSKÝ, R. Výuka objektově orientovaného programování žáků základních a středních škol [online]. <URL: http://vyuka.pecinovsky.cz/prispevky/Vyuka_OOP_zaku_zakladnich_a_strednich_skol.pdf>.
3. KURELOVÁ, M. Pedagogika II. 1.vyd. Ostrava : PdF OU, 1993. ISBN 80-7042-068-5.
4. PRŮCHA, J., WALTEROVÁ, E., MAREŠ, J. Pedagogický slovník. 4. aktualizované vyd. Praha : Portál, 2003. ISBN 80-7178-772-8.
5. ČÁP, J., MAREŠ, J. Psychologie pro učitele. 2. vyd. Praha : Portál, 2007. ISBN 978-80-7367-273-7.
6. KUBRICKÝ, J. Objektově orientované programování ve výuce. Journal of Technology and Information Education. 2009, Olomouc - EU, Univerzita Palackého, Ročník 1, Číslo 3, s. 136 - 1388. ISSN 1803-537X (print). ISSN 1803-6805 (on-line).
7. LAVIN, P. PHP objektově orientované – koncepty techniky a kód. 1. vyd. Praha : Grada, 2009. ISBN 978-80-247-2137-8.

Lektoroval: Mgr. Jan Lavrinčík, DiS.

Kontaktní adresa:

Jan Kubrický, Mgr.,
Katedra technické a informační výchovy,
Pedagogická fakulta UP, Žižkovo nám. 5, 771 40
Olomouc, ČR, tel. 00420 585 635 806,
e-mail: jan.kubricky@upol.cz