

ŽELVÍ GRAFIKA VE VZDĚLÁVÁNÍ

LAVRINČÍK Jan – KLEMENT Milan, ČR

Resumé

Článek se zabývá významem, tříděním a využitím rekurzivních algoritmů ve vzdělávání, se zaměřením na želví grafiku. Vysvětluje, z jakých částí se rekurzivní algoritmus skládá a zamýšlí se nad jeho využitím ze současného pohledu vzdělávání. Zdůrazňuje výběr rekurzivních algoritmů z hlediska odborné náročnosti a rozvoje abstraktního myšlení u žáků nižšího sekundárního vzdělávání. Seznamuje čtenáře s klady a zápory jednotlivých algoritmů. V závěru polemizuje nad využitím programovacího jazyka Visual Basic 6 jako tématu výuky nižšího sekundárního vzdělávání a jeho možnostmi integrace do RVP.

Klíčová slova: algoritmus, rekurzivní algoritmus, Visual Basic, želví grafika.

TURTLE GEOMETRY IN EDUCATION

Abstract

The article deals with the significance creation, classification, and use recursion algorithms in education, especially turtle geometry. The origins of the parts are explained, notion recursion algorithms, and use in the contemporary time are remembered. The choice of recursion algorithms is emphasized from the field of expert demands and expansion of abstract thinking in pupils the lower secondary school. It pontificate readers with plus and minus common algorithms. At the end, it discusses about the future use of Microsoft Visual Basic 6 in the lower secondary education and potential usage in program RVP.

Key words: algorithm, recursion algorithm, Visual Basic, turtle graphic.

1 Úvod

V současné době se stala výuka moderních informačních a komunikačních technologií standardním nástrojem a trendem ve vyučovacím procesu na všech typech škol. Na druhém stupni základních škol je zaměřena především na zvládnutí práce s kancelářským balíkem Microsoft Office a základními grafickými nástroji. V souvislosti se zvýšením konkurenceschopnosti žáků základních škol se začal především jako volitelný předmět objevovat výuka programování pro žáky. Již v roce 1967 byl ve společnosti BBN navržen jazyk Logo na podporu a rozvoj logického myšlení (1). U novějších verzí se označení želví grafika začalo používat v souvislosti s pohybem želvy po pláži a kreslením. Svým spuštěným ocasem zanechává stopu (grafiku) v písku. Tento způsob myšlení byl přenesen do programovacích jazyků označovaných souhrnně termínem Logo. V českých a slovenských podmínkách i díky knize (1) se nejvíce ujal nástroj Imagine Logo (2).

Návrh programovacího jazyka speciálně pro žáky základních škol má svůj význam, který vyslovil psycholog Jean Piaget. Výklad spočíval v myšlence, že dětské myšlení nebo kognitivní procesy jsou podstatně rozdílné od těch dospělých. Ze čtyř etap dle Piageta (3), jsou pro nás zajímavé tři. Stádium konkrétních operací: od 7 do 12 let (dokáže logicky přemýšlet o konkrétních událostech, pochopení stálosti počtu, množství a hmotnosti a 4. stádium formálních operací: 12 let a více (dokáže logicky myslet o abstraktních pojmech). Díky výše popsaným skutečnostem jsme se pokusili spojit psychologické teorie J. Piageta

s využitím prvků jazyka Logo přeneseným do dnešního nejjednoduššího objektově orientovaného programovacího jazyka Microsoft Visual Basic. Díky tomuto spojení můžeme očekávat tvůrčí činnost žáků s akcentem na rozvoj logického myšlení, ale i solidní základ na kterém mohou žáci stavět v rámci vyššího sekundárního vzdělávání a terciárního vzdělávání.

Příspěvek se zabývá koncepcí výuky programování želví grafiky pomocí standardního programovacího jazyku Microsoft Visual Basic 6.

2 Rekurzivní algoritmy

Algoritmus je dle Ottova slovníku naučného definován jako přesný návod či postup, kterým lze vyřešit daný typ úlohy. Pojem algoritmus se nejčastěji objevuje při programování, kdy se jím myslí teoretický princip řešení problému (oproti přesnému zápisu v konkrétním programovacím jazyce). Obecně se ale algoritmus může objevit v jakémkoli jiném vědeckém odvětví. Jako jistý druh algoritmu se může chápat i např. kuchyňský recept. V užším smyslu se slovem algoritmus rozumí pouze takové postupy, které splňují některé silnější požadavky (4).

Pro účely informatiky, bychom mohli v užším smyslu algoritmus definovat jako soubor instrukcí, které v případě, že se provedou, vedou k řešení nějaké úlohy nebo problému. Rovnice nebo řada kroků musí být významově jednoznačná a mít jasný bod ukončení (4).

Klasifikace algoritmů souvisejících s programováním pomocí želví grafiky:

Algoritmy můžeme dále dělit dle různých kritérií, nejčastěji podle prováděné činnosti.

Rekurzivní algoritmy, typ algoritmů, které volají samy sebe v nezměněné podobě nebo částečně modifikované podobě (zmenšení, zvětšení parametrů). V programovacích jazycích jsou často realizovány pomocí cyklů (4).

Algoritmy typu rozděl a panuj, dělí problém na menší podproblémy, na něž se rekurzivně aplikují (až po triviální podproblémy, které lze vyřešit přímo), po čemž se dílčí řešení vhodným způsobem sloučí (4).

Rekurzivní algoritmy jsou zvláštní v tom, že současně patří do dvou skupin algoritmu, zároveň rekurzivní a typu rozděl a panuj. Model želva je právě vhodným nástrojem pro praktické pochopení a procvičení rekurzivních algoritmů.

3 Využití modelu želva pro výuku nižšího sekundárního vzdělávání

K návrhu řešení programování pomocí počítačové grafiky jsme vybírali programovací jazyk s ohledem na jednoduchost, srozumitelnost a standardizaci. Našemu řešení nejvíce vyhovoval objektově orientovaný programovací jazyk Microsoft Visual Basic. Zvolili jsme starší verzi 6, protože obsahuje jednodušší menu, méně ovládacích prvků a nižší náročnost na hardware počítače (5, s. 12).

Standardně námi vybraný jazyk není stavěný pro model želví grafiky. Můžeme jej však nadefinovat do samostatného Class Modulu a přiřazením ke každému prázdnému projektu (s menšími úpravami) je okamžitě připravený k používání. Řešení žákům nutí používat programovací jazyk Visual Basic, ovšem s výrazně usnadněnou prací.

V našem případě funguje objektově orientované programování následovně: za objekt je považována želva, která má nějaké metody a vlastnosti. V následujícím odstavci se podíváme na vlastnosti želvy.

Příklad:

ZELVA.ZOBRAZ

(ZELVA = objekt, ZOBRAZ = vlastnost)

Tabulka 1: Vlastnosti želvy (pohyb, rotace).

<u>Vlastnosti želvy (pohyb, rotace):</u>	
ZOBRAZ	,,, zobrazí aktuální polohu želvy,
POSUN 50	,,, posune želvu o danou hodnotu (parametr: délka),
OTOC 90	,,, otočí želvu o danou polohu (parametr: úhel).

Tabulka 2: Vlastnosti želvy (elementární kreslení).

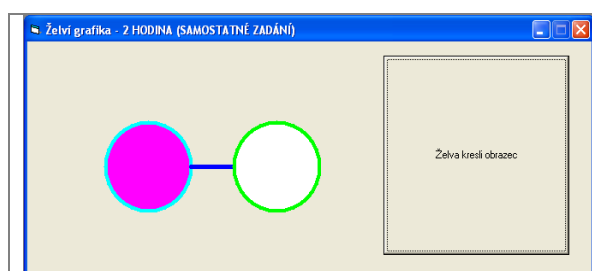
<u>Vlastnosti želvy (elementární kreslení):</u>	
USECKA	,,, nakreslí úsečku želvou (<i>parametry</i> : délka, šířka, barva),
KRUH	,,, nakreslí kružnici želvou (<i>parametry</i> : poloměr, excentricita, ...),
BOD	,,, nakreslí bod pomocí želvy (<i>parametry</i> : tloušťka, barva),
KRUHPLNY	,,, nakreslí plný kruh (<i>parametry</i> : výplň, barva výplně, ...).

Ukázka z metodiky pro žáky:

Samostatný příklad je zaměřený na kreslení dvou kruhů s různou výplní a úsečky. V příkladu má kruh poloměr 50 jednotek, šířku okraje 4 body, barvu ohraničení vbCyan a barvu výplně vbMagenta, excentricita je nastavena na 1 = kruh.

Náměty k samostatnému přemýšlení:

1. Vyzkoušejte změnit excentricitu kruhu na 0,6 a poté na 3, zdůvodněte rozdíly?
2. Vyzkoušejte si změnit typ výplně (1, 5, 7).



Obrázek 1: Samostatná práce.

```
zelva.kruhplny 50, 4, 0, vbCyan, vbMagenta, 1
zelva.posun 50
zelva.usecka 100, 5, vbBlue
zelva.kruhplny 50, 4, 0, vbGreen, vbWhite, 1
```

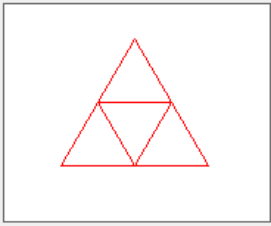
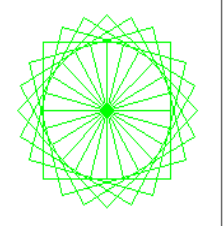
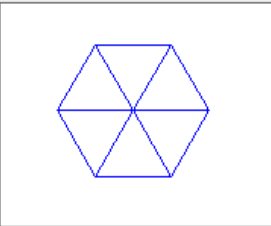
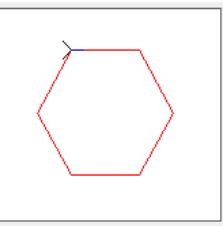
Tabulka 3: Správné řešení (zdrojové kódy).

Četně využívaným prvkem při tvorbě rekurzí jsou cykly. Cykly typu For...Next slouží v procedurách událostí k provádění určité skupiny příkazů pro pevný, předem daný počet opakování. Cyklus For...Next má proto smysl například při provádění několika spolu souvisejících výpočtů, při práci s elementy na obrazovce nebo při zpracování určité množiny vstupních dat. Cyklů můžeme používat vícenásobně, jak vidíme v příkladu rotace čtverce nebo kombinovat. Obecnou syntax práce s cykly bychom mohli zapsat jako:

For čítač = začátek **To** konec [**Step** krok]
[příkazy]
[Exit For]
[příkazy]
Next [čítač]

V praxi si můžeme všimnout, že rekurzivní grafika má úzkou souvislost s některými lineárními fraktály (např. obrázek 2: Sierpinského trojúhelník). Obrázky 3, 4, 5 jsou ukázkou práce s cykly v modelu želví grafiky.

Obrázky 2, 3, 4, 5.: Ukázky zdrojových kódů Visual Basic 6.

Dim i& zelva.otoc -360 / 3 For i = 1 To 3 trojuhelnik 50, vbRed zelva.posun 50 zelva.otoc 360 / 3 Next i		Dim i&, j& For j = 1 To 24 For i = 1 To 4 zelva.usecka 50, vbGreen zelva.otoc -90 Next i zelva.otoc 15 Next j	
Sierpinski		Čtverec rotace	
Dim i& For i = 1 To 6 trojuhelnik 50, vbBlue zelva.otoc 60 Next i		Dim i& For i = 1 To 6 Linka 50, vbRed Next i zelva.zobraz	
Šestiúhelník 1		Šestiúhelník 2	

Příklady představené v tomto příspěvku představují pouze zlomek potenciálu námi navrženého řešení. Model umožňuje i práci se složitějšími rekurzivními tvary a tvorbu učebních úloh pro nadané žáky.

4 Závěr

Příspěvek opírající se o problematiku želví grafiky v aplikaci Visual Basic 6 přispívá k neustálému cílenému vývoji, vnášení nových netradičních forem a metod ve vyučovacím procesu a podpoře abstraktního myšlení u žáků. Pozitivně může ovlivnit formování klíčových dovedností a postojů žáka v oblasti ICT a pomoci rozvoji objektového programování na 2. stupni ZŠ. Nemalou mírou může pomoci učitelům jako nástroj motivace.

V praxi může posloužit učitelů různých typů škol jako pomůcka k tvorbě příkladů z oblasti rekurzivních algoritmů. Z pohledu grafiky a designu k vizualizaci a zobrazení zajímavých řešení obrazců elementární geometrie Odborná vědecká stat' nastiňuje možnost propojení výuky ICT se zábavou a zvýšit tím interakci učitel-žák s podporou znalostí o algoritmizaci a základů programování v programovacím jazyce Microsoft Visual Basic 6.

5 Literatura

- [1] Wikipedia, otevřená encyklopedie. [on-line]. 2010. [cit. 2010-05-21]. URL: <[http://cs.wikipedia.org/wiki/Logo_\(programovac%C3%AD_jazyk\)](http://cs.wikipedia.org/wiki/Logo_(programovac%C3%AD_jazyk))>.
- [2] KALAŠ, I., BLAHO, A. *Imagine Logo : učebnice programování pro děti*. 1. vyd. Brno : Computer Press, 2006. 48 s. ISBN 80-251-1015-X.
- [3] Wikipedia, otevřená encyklopedie. [on-line]. 2010. [cit. 2010-05-21]. URL: <<http://cs.wikipedia.org/wiki/Piaget>>.
- [4] Wikipedia, otevřená encyklopedie. [on-line]. 2010. [cit. 2010-05-21]. URL: <<http://cs.wikipedia.org/wiki/Algoritmus>>.
- [5] KLEMET, M. *Základy programování v jazyce Visual Basic*. 1. vyd. Olomouc : VUP, 2002. 336 s. ISBN 80-244-0472-9.
- [6] ABELSON, H., diSESSA, A. *Turtle Geometry : The Computer as a Medium for Exploring Mathematics*. 1st edition. Massachusetts : MIT Press, 1980. 478 s. ISBN 0-262-51037-5.

Lektoroval: PaedDr. PhDr. Jiří Dostál, Ph.D. (Moravská vysoká škola v Olomouci)

Kontaktní adresy

Jan Lavrinčík, Mgr., DiS.
Pedagogická fakulta Univerzity Palackého v Olomouci,
Žižkovo nám. 5, 771 40,
+420 585 635 813,
Email: nobilis.felis@seznam.cz,
Www pracoviště: www.kteiv.upol.cz

Milan Klement, PhDr., Ph.D.
Pedagogická fakulta Univerzity Palackého v Olomouci,
Žižkovo nám. 5, 771 40,
+420 585 635 813,
Email: nobilis.felis@seznam.cz,
Www pracoviště: www.kteiv.upol.cz