

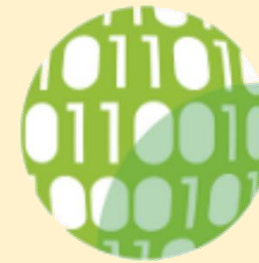


Bádateľský prístup vo vyučovaní programovania



Ľubomír Šnajder
Ústav informatiky PF UPJŠ v Košiciach

lubomir.snajder@upjs.sk



Otázky

1. Prečo neučiť programovanie po starom? Tradícia vs súčasnosť.
2. Môžu žiaci na hodine programovania skúmať podobne ako na hodinách fyziky či chémie?
3. Čo si o tom myslia učitelia informatiky?

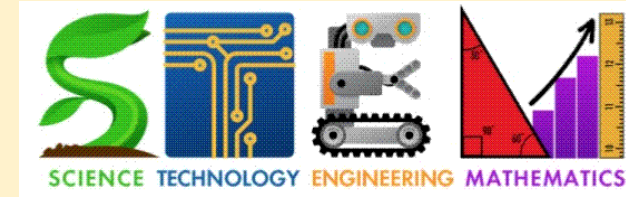
„Tradičná“ výučba programovania

- jazyky – BASIC, Pascal ...
- silné prepojenie na matematiku
- práca s číslami, textami ... grafikou
- zameranie na jazykové prostriedky a vybrané typy algoritmov
- ... vetvenie, cykly, polia, procedúry (rekurzia) ...
- tradičný prístup výučby (motivácia, expozícia, fixácia, diagnostika)

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

„Súčasná“ výučba programovania

- jazyky – Blockly (Scratch, Ai2), Python ...
- prepojenie na rôzne predmety
- udalosti, široké škála vstupov (senzorov), sieťová komunikácia, AI, AR ...
- zameranie na kreativitu a riešenie problémov
- ... funkcie, cykly 1, vetvenie, cykly 2, zoznamy ...
- cloud – virtuálne triedy, samoštúdium, komentovanie, remixovanie
- rôzne zariadenia – roboty, mobilné zariadenia, minidrony, mikroprocesorové dosky ...
- nové prístupy výučby – BOV, PBL, PV ...



Národný projekt IT Akadémia

– podpora výučby programovania

- 1.1a Inovatívne metodiky
 - ZŠ – Scratch (20), EV3 (5), BBC micro:bit (3)
 - SŠ – Python (27), Ai2 (6), RPi (3)
- 1.1b Nové informatické predmety
 - Riešenie problémov a programovanie (Python)
 - Programovanie mobilných zariadení (App Inventor)
 - Objektový prístup k riešeniu problémov (Java)
 - Programovanie webových stránok (Javascript, PHP, ...)
- 1.1c Motivačné predmety
 - Informatika v prírodných vedách a matematike (Python, Asymptote ...)

Národný projekt IT Akadémia

– podpora výučby programovania

- 1.2 Vzdelávanie učiteľov informatiky
 - kurzy venované inovatívnym metodikám, novým predmetom, programovaniu
 - workshopy
 - konferencie
 - otvorené hodiny ...
- 1.3 Motivačné mimotriedne a mimoškolské aktivity
 - krúžky
 - letné tábory
 - súťaže – PALMA, PALMA junior, IHRA, ...

Vývoj a nasadenie bádateľsky orientovanej metodiky

- Vývoj metodík
 - Výber tém
 - Návrh štruktúry metodík
 - Tvorba priebežnej verzie metodiky – overenie metodiky v praxi (dotazník po výučbe, vyplnené pracovné listy) – dve iterácie
 - Finálna verzia metodiky (skúsenosti a postrehy z výučby)
- Servis pre učiteľa:
 - Školenie k BOV a inovatívnym metodikám
 - Metodické materiály s prílohami (pracovné listy, pracovné súbory, zbierky úloh s riešeniami, nástroje formatívneho a sumatívneho hodnotenia)

Ukážka bádateľsky orientovanej metodiky

- Téma: Cyklus s pevným počtom opakovaní
 - Kognitívne ciele
 - Podkoncepty informatického myslenia
 - Pomôcky
 - Osnova hodiny
 - Spätná väzba od učiteľov

Cyklus s pevným počtom opakovaní

– kognitívne ciele

- Upraviť program s opakujúcimi sa príkazmi bez cyklu na program s cyklom.
- Vysvetliť význam príkazu `for i in range(<pocet>)` ako príkazu s pevným počtom opakovaní.
- Charakterizovať (odlíšiť) hlavičku cyklu (zatiaľ len počet opakovaní) a telo cyklu (ako skupinu opakujúcich sa príkazov, ktorá je medzerami odsadená vzhľadom k zápisu hlavičky cyklu).
- Vytvoriť program využívajúci cyklus `for`, vlastné funkcie a základné príkazy korytnačej grafiky.

Cyklus s pevným počtom opakovaní

– podkoncepty informatického myslenia

- Logika
 - (LOG5) logicky **zdôvodniť rozdelenie** algoritmu/programu/problému/objektu na menšie časti
- Dekompozícia
 - (DEK2) hierarchická dekompozícia – **hierarchicky rozdeliť** objekty/problémy/procesy na menšie časti tak, aby sa dali využiť pre dosiahnutie cieľa
- Hľadanie vzorov
 - (VZO4) **zovšeobecniť** riešenie podobných problémov na celú triedu, zovšeobecniť na základe konkrétnych prípadov (namiesto viacerých podprogramov použijeme jeden s viacerými parametrami)
 - (VZO5) **preniesť/použiť vzory/myšlienky/riešenia** z jedného problému na druhý problému

Cyklus s pevným počtom opakovaní

– pomôcky

- priečink **I_SS_04_Python_ucitel** obsahujúci:
 - metodiku,
 - pracovný list s riešeniami úloh,
 - zbierku úloh s riešeniami,
 - zdrojové súbory s riešeniami úloh,
 - tabuľku pre zápis výsledkov žiackych riešení úloh z pracovného listu.
- priečink **I_SS_04_Python_ziak** obsahujúci:
 - pracovný list,
 - zbierku úloh,
 - pracovné súbory.

Cyklus s pevným počtom opakovaní

– osnova hodiny (podľa učebného cyklu 5E)

- **Zapojenie** (5 minút) – analýza obrázkov s opakujúcim (aj neopakujúcim) sa vzorom na pravidelných pozíciách a diskusia k spoločným črtám opisov obrázkov (úloha 1 z pracovného listu)
- **Skúmanie** (7 minút) – skúmanie hotových a upravovanie nekompletných príkazov obsahujúcich príkaz cyklu for, získanie prvej skúsenosti a predstave o príkaze cyklu for (úloha 2 z pracovného listu)
- **Vysvetlenie** (6 minút) – diskusia k ponímaniu príkazu cyklu for a naše zhrnutie učiva
- **Rozpracovanie** (14 minút) – programovanie obrázkov s opakujúcimi sa vzormi na pravidelne rozmiestnených pozíciách (úlohy 3 až 5 z pracovného listu)
- **Vyhodnotenie** (8 minút) – vyriešenie sebahodnotiaceho testu (úlohy 6 a 7 z pracovného listu) s našim vyhodnotením

ZAPOJENIE

Úloha 1 Opíšte **stručne** a čo **najpresnejšie** (pre kamaráta na telefóne), čo vidíte na uvedených obrázkoch **A, B, C, D**:

A



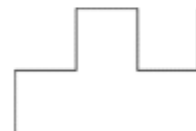
B



C



D



A

B

C

D

Uvedte, ktoré z obrázkov sa vám opísali ľahšie a ktoré ťažšie

Uvedte, čo majú spoločné vaše popisy ľahšie opísateľných obrázkov:

.....

Úloha 2 Preskúmajte obidva uvedené programy **A** a **B**. Najprv len na základe prečítania uvedeného programového kódu, potom po spustení ich kódov uložených v súboroch **04_02A.py** a **04_02B.py**.

A

```
import turtle

tabula = turtle.Screen()
pero = turtle.Turtle()
pero.penup()

pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)
pero.dot(40, 'orange')
pero.forward(40)

tabula.mainloop()
```

B

```
import turtle

tabula = turtle.Screen()
pero = turtle.Turtle()
pero.penup()

for i in range(5):
    pero.dot(40, 'orange')
    pero.forward(40)

tabula.mainloop()
```

- a) Opíšte obrázky, ktoré vykreslia uvedené programy **A** a **B**:
-
- b) Ktorý zo zápisov pokladáte za lepší a prečo:
- c) Ako by sa zmenil výsledok programu **B**, ak by sme v riadku 7 namiesto `range(5)` uviedli `range(10)`:
-

VYSVETLENIE

Prediskutujte a vysvetlite:

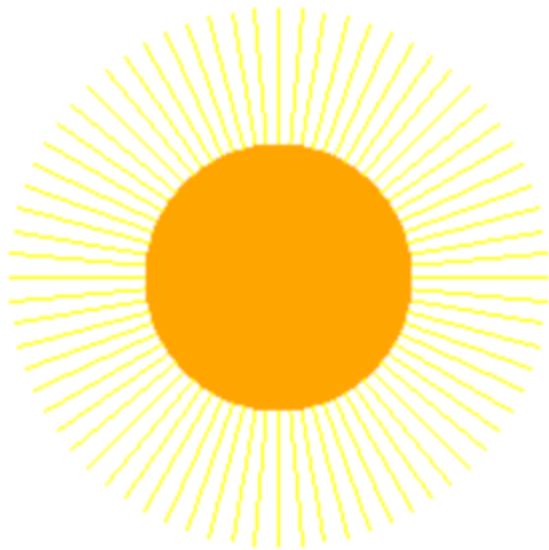
1. Načo je dobrý príkaz cyklu `for`?
2. Z akých časti sa skladá zápis príkazu cyklu `for`?
3. Ako funguje príkazu cyklu `for` , aký význam majú hlavička a telo cyklu?
4. Musíme použiť v cykle `for` vlastné funkcie?

ROZPRACOVANIE

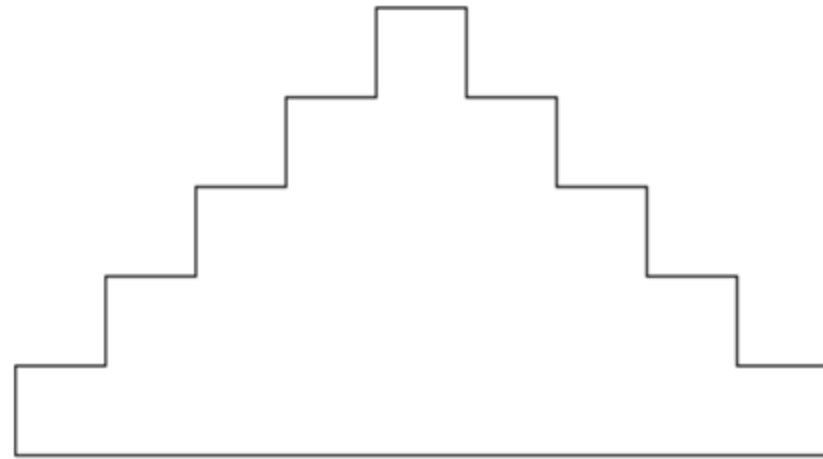
Úloha 4 Vytvorte programy `04_04A_slnko.py` a `04_04B_schodiky.py` vykresľujúce uvedené obrázky **A** a **B** tak, aby bolo grafické pero na konci vykreslenia v počiatočnej pozícii a počiatočnom natočení.

Riešte
podľa
pokynov
učiteľa

A



B



HODNOTENIE

Sebahodnotiaci test

Úloha 6 Uvedte koľkokrát sa vykoná vlastná funkcia prikaz() v programoch **A**, **B** a **C**.

A

```
for i in range(3):  
    prikaz()  
    prikaz()  
prikaz()
```

B

```
for i in range(3):  
    prikaz()  
  
prikaz()
```

C

```
for i in range(3):  
    prikaz()  
prikaz()  
prikaz()
```

Spätná väzba od učiteľov overujúcich metodiku v praxi

- „Odporúčam pripravenú metodiku realizovať na **dvoch vyučovacích hodinách** a prejsť všetky úlohy aj nepovinné (8-12), aby žiaci mali možnosť **precvičiť si učivo.**“
- „V pracovnom liste navrhujem **vymeniť poradie úloh č. 3 a č. 4**, aby žiaci najprv použili cyklus úpravou programov a až potom vytvárali program vlastný“
- „Navrhujem rovno vykresľovať tieto obrázky v pythone **Tkinter**, nie v turtle, je to pre mňa návrat k imaginu :D“

Spätná väzba od učiteľov overujúcich metodiku v praxi

- „Úlohy sú dostatočne zamerané na pochopenie cyklov. Možno by som zdôraznil aj **cyklus v cykle**.“
- „Žiaci majú väčší problém s návrhom riešenia na papier, s uvedením počiatočného a koncového bodu vykresľovania a natočenia pera, s naprogramovaním úlohy už problém nemajú.“
- „Musím kresliť, pretože niektorým žiakom chýba predstavivosť“
- „Úlohy na bádanie a objavovanie cyklu sú pekné, v prvej úlohe mali žiaci problém opísať obrázky. Väčšina vychádzala z bežného života a opisovala analógiu obrázku zo života, nie algoritmicky. Úloha 5 nie je až taká triviálna, ako sa na prvý pohľad zdá, a možno na cyklus by bolo vhodné vymyslieť menej náročnú (aj časovo) úlohu.“

Diskusia

- BOV a učítelia
 - vedomosti
 - postoje (pružnosť meniť svoje pg. prístupy, otvorenosť k zmene)
 - skúsenosti
- BOV a žiaci
 - zvyk na výklad učiteľa
 - nezvyk pracovať počas celej hodiny
 - nezvyk analyzovať problém, kresliť, ale rovno písať kód ... používať pracovný list