

IB111 ÚVOD DO PROGRAMOVANÍ SKRZE PYTHON



Autor: Slavomír Krupa



N-TICE



TUPLE

- ★ N-tica (tuple) je dátová štruktúra podobná zoznamu
 - Je možné pristupovať cez indexy, rezy...
- ★ Má štruktúru nie poradie
- ★ Zložky často nemajú rovnaký dátový typ
- ★ Používa sa pre dočasné typy

```
tuple_example = (1, 2, "3") #n-tica
```

```
list_example = [1, 2, 3] #zoznam
```

```
list_example[1] = 5 #zoznam povoluje zmeny
```

```
#ntica je nemenna
```

```
tuple_example[1] = 5
```



DICTIONARY



DICTIONARY

- ★ Slovník (dictionary, map) je dátová štruktúra reprezentujúca asociatívne pole

```
dictionary = {} # vytvorenie slovníku  
in_dictionary = {'A': 6, 'Z': 4}  
var = in_dictionary['A'] # prístup k hodnote  
dictionary['A'] = 3 # vytvorenie hodnoty  
in_dictionary['A'] += 3 # zmena existujúcej  
hodnoty
```



DICTIONARY

★ Nie je zaručené poradie hodnôt

```
in_dictionary = {'A': 6, 'Z': 4}
for key in in_dictionary:
    print(key, ':', in_dictionary[key])
# Vypise:
# Z : 4
# A : 6
```




DICTIONARY

- ★ Klúčom môžu byť:
 - základné dátové typy (celé číslo, reťazec...)
 - N-tica (nie zoznam!)
 - Užívateľom definované dátové typy (* ak implementuju `__hash__` a `__eq__`)

```
#zakladne datove typy
```

```
basic = {6:2}
```

```
#tuple
```

```
t_dict = {(1, 5): "Black"}
```



DICTIONARY

★ Hodnotou môžu byť ľubovoľné dátové typy

```
basic = {6:2}
l_example = {1: [1, 2, 3]}
weird = {
  "11": 6, 8: [161], ("tuple", 2) : {1: "1"}
}
```

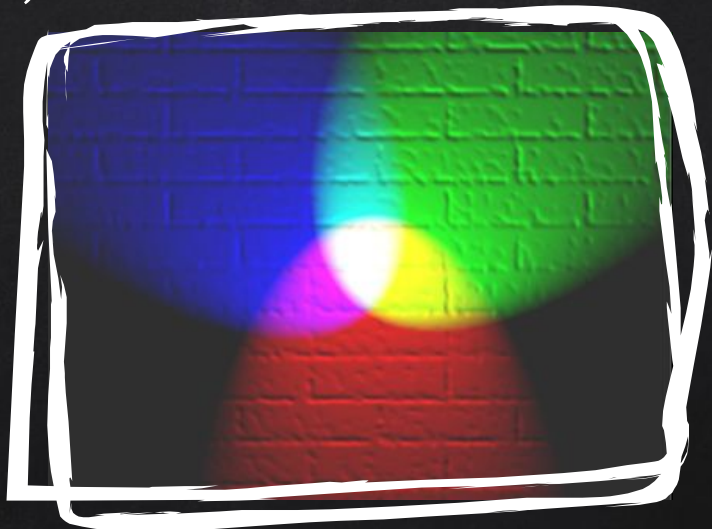



GRAFIKA



RGB

- ★ R(ed)G(reen)B(lue) je aditívny farebný model, pri ktorom svetlo želanej farby vzniká zmiešaním červeného, zeleného a modrého svetla vhodnej intenzity.





RGB

- ★ Definujete hodnoty jednotlivých zložiek a podľa toho namiešate farbu. (R,G,B)
- ★ Alebo použijete color picker:
 - http://www.rapidtables.com/web/color/RGB_Color.htm
- ★ Budeme používať knižnicu pillow a triedu Image.
- ★ Dokumentácia knižnice:
<https://pillow.readthedocs.io/en/3.4.x/reference/Image.html>



PILLOW

★ Nový obrázok

```
from PIL import Image
#kodovanie farieb,rozmary(sirka,vyska),
#farba pozadia
image = Image.new("RGB", (50, 50), (255,
255, 255))
image.putpixel((1, 1), (0, 0, 0))
image.save("graph.png")
image.show()
```



PILLOW

★ Existujúci obrázok

```
from PIL import Image
#nacitanie obrazku zo suboru a
konvertovanie do RGB
im = Image.open("file").convert("RGB")
#ziskanie rozmerov obrazku
width, height = im.size
#ziskanie farby pixelu
r, g, b = im.getpixel((0, 0))
```

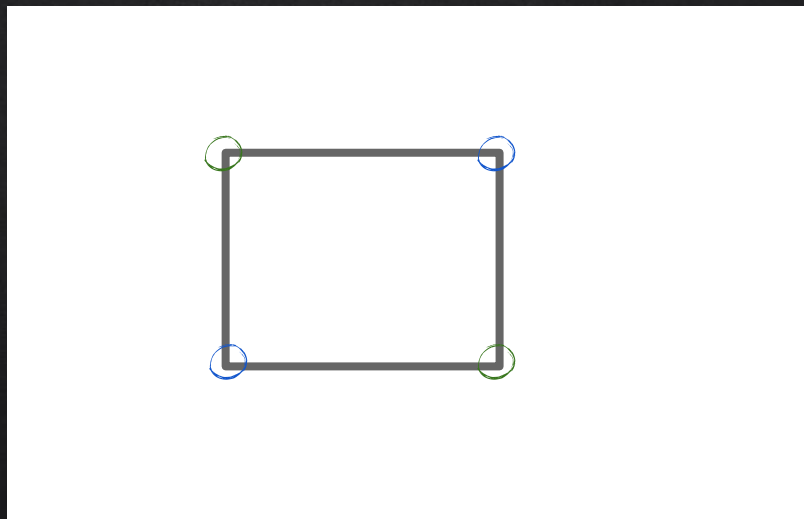


SÚRADNICE

X

$(0,0)$ \longrightarrow $(\text{WIDTH}-1,0)$

Y



$(0,\text{HEIGHT}-1)$

$(\text{WIDTH}-1,\text{HEIGHT}-1)$



ÚLOHA 1

- ★ Naprogramujte procedúru, ktorá vykreslí obdĺžnik zadaných rozmerov do obrázku s rozmermi 100x100 a zobrazí výsledok (nebude ho nikde ukladať).
- ★ `rectangle(10, 20, (0, 0, 0))`





ÚLOHA 2



Upravte procedúru `rectangle`.

- `color` = farba obdĺžniku
- `start` = súradnice bodu vľavo hore
- `end` = súradnice bodu vpravo dole
- `image` = obrázok kam bude vykresľovať



`rectangle((89, 89), (99, 99), (0, 0, 0), ima
ge)`

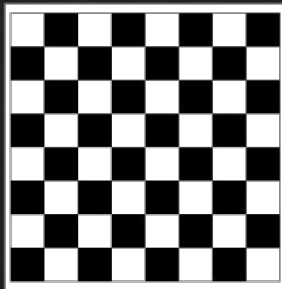




ÚLOHA 3

- ★ Vytvorte procedúru, ktorá vykreslí šachovnicu
 - Bude mať klasické rozmery 8x8
 - Štvorec bude mať rozmer 50px
- ★ `chessboard()`

Ilustračný obrázok





ÚLOHY



Vytvorte procedúry, ktoré budú robiť úpravy s existujúcim obrázkom. Ak nemáte žiadny dostupný môžete použiť tento.

- otočenie existujúceho obrázok o zadaný počet stupňov (dokumentácia rotate)
- Odstránenie červenej zložky farby
- Pripojí za existujúci obrázok zrkadlovú kópiu
- A dalšie...



MERRY XMAS

CREDITS

Special thanks to all the people who made and released these awesome resources for free:

- Presentation template by [SlidesCarnival](#)
- Photographs by [Unsplash](#)