

Velký domácí úkol č. 6

Deadline: neděle 19. 12. 23:59

1) [10 bodů] Ze studijních materiálů si stáhněte soubor `book.txt`. Napište funkci `findNames :: string → [string]`, která na vstupu vezme cestu k souboru a na výstup vrátí seznam jmen hlavních postav. Každé jméno začíná velkým písmenem a objevuje se v textu alespoň 5 krát. Že s tímto přístupem vrátíte i často se vyskytující první slova ve větě ignorujte.

2) [15 bodů] Ze studijních materiálů si stáhněte soubor `hidden_image.png`. Na první pohled sice vypadá nevinně, je v něm ale ukryt jiný obrázek a vaším úkolem bude jej najít. Napište funkci `decipher :: string → string → void`, která na vstupu vezme cestu k obrázku k dešifrování a jméno, pod jakým se má uložit obrázek dešifrovaný. Každý pixel obrázku se skládá ze tří barevných složek nabývajících hodnot mezi 0 a 255; toto číslo je uloženo na osmi bitech a druhý obrázek byl ukryt tak, že byly jeho 4 významnější bity uloženy na pozici 4 méně významných bitů v obrázku prvním. Při řešení si vystačíte s násobením a dělením, práci vám ale mohou usnadnit tyto bitové operátory: `<< >> & |`

3) [15 bodů] Na každém řádku v souboru `file` je uložený kruh definovaný svým středem a poloměrem. Tyto dva údaje jsou od sebe odděleny čárkou, střed se skládá ze souřadnic `x` a `y` v závorce oddělené čárkou. Závorka může být buď kulatá, nebo hranatá a místo čárky v zápise smí být středník. Některé řádky jsou zapsány správně a na jiných nějaká část formátu chybí. Napište funkci `write :: string → void`, která na vstupu dostane cestu k souboru `file` a správně zadané kruhy nakreslí. Jestli je formát v pořádku zjistěte pomocí regulárních výrazů. Při vykreslování začněte s černým čtvercem o velikosti `550x550` a uvnitř každého kruhu invertujte barvu. Svůj výsledek si můžete porovnat se vzorovým řešením `circles.png`. Pro parsování jednotlivých řádků se vám bude hodit metoda `groupdict()` a formátovací řetězec `"(?P<name>[...])"`, kde [...] zastupuje libovolný regulární výraz, viz příklad.

```
string = "jedna dvě tři"  
result = re.match(r"\w+ (?P<middle>\w+) \w+", string)  
dictionary = result.groupdict()  
print(dictionary["middle"])
```