Krátké příklady pro Python

Bedřich Said

17. října 2016

Jednořádkové příklady

0. Napište funkci, která jednoduše vydělí dvě čísla. Jak budete řešit dělení nulou? Co když na vstupu dostanete třeba text? Zkuste tuto funkci udělat opravdu robustní, tedy Python pro žádný vstup nevypíše chybovou hlášku.

1. Napište funkci `boolean expression(a, b, c)`, která přijímá tři proměnné typu boolean. Funkce vrací True, pokud alespoň dvě vstupní hodnoty jsou `True`. Naopak vrací `False`, pokud alespoň dvě vstupní hodnoty jsou `False`. (Inspirace: Triple Modular Redundand system, využívaný pro zvýšení spolehlivosti, využívá Hammingův kód Hamming-3,2)

2. Napište funkci `pixelDistance(x1, y1, x2, y2)`, která přijímá souřadnice dvou bodů typu `double`. Na výstup vrátí vzdálenost mazi těmito dvěma body v euklidovské rovině. Vzdálenost dvou bodů v euklidovské rovině je dána:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

3. Napište funkci `isEnd(ch)`, která na vstupu dostane právě jeden znak, a vrátí `True`, pokud se tento znak nachází uvnitř slova.

Funkce má rozpoznat konec slova na jednom řádku textu. Za každým slovem se nachází právě jedna mezera. Výjimku tvoří pouze poslední slovo na řádku, za kterým následuje pouze znak nového řádku. Funkce `isEnd` vrátí `False` právě když na vstupu dostane mezeru nebo znak nového řádku (False = znak není uvnitř slova), jinak vrátí `True`.

4. Napište funkci `roundNumber(x)`, která na vstupu dostane jedno desetinné číslo. Funkce tuto hodnotu zaokrouhlí na celé číslo a vrátí jako typ `int`. (Nepoužívejte prosím vestavěnou systémovou funkci k zaokrouhlování čísel.)

5. Napište funkci `printSpecialChars()`, která na obrazovku vypíše přesně tento text ([tabulator] nahraďte příslušným znakem):

```
~!@#$%^&*"()'[tabulator]_+:"?
<>{}[]-/\
```

Čtyřřádkové příklady

1. Napiště funkci `isPalindrom(text)`, která vrátí `True`, právě když zadaný vstupní text je palindrom, jinak vrátí `False`. Palindrom je text, který se čte zleva i zprava stejně. Známý příklad je třeba `kobylamamalybok` nebo `jelenovipivonelej`.

2. Napište funkci `correctBrackets(text)`, která na vstupu dostane řetězec složený pouze z kulatých závorek. Funkce vrátí `True`, právě když vstup je korektně uzávorkován, tedy v řetězci nechybí žádná levá ani pravá závorka.

3. Stojíme uprostřed parku na rozcestí. Všechny cesty od nás vedou k okraji parku a mohou se dále dělit. Cesty tvoří strom a já stojím v jeho kořeni (ať se vydám kamkoliv, tak na své místo se vrátím jedině po stejné cestě). Svou procházku parkem si chci co nejvíc užít a tak hledám nejdelší cestu (větev) ven z parku.

Mapu parku mám zadanou jako seznam čísel. Každá křižovatka má přířazené přirozené číslo. Já stojím na křižovatce "nula", která nemá přiřazené žádné číslo. V seznamu je vždy na indexu $2i$ index křižovatky, na kterou vede cesta vlevo. Na indexu $2i+1$ je index křižovatky, na kterou vede cesta vpravo. Mezi dvěma křižovatkami je vždy stejná vzdálenost. Všechny cesty, které vedou přímo ven z parku, odkazují na index 0.

Napište funkci `longestBranch(tree, i)`, která vrátí počet křižovatek na nejdelší cestě ven z parku.