

Skúšobné zadanie na druhé cvičenie

23. mája 2014

1 Príprava

Vaše riešenie bude obsahovať iba definície funkcií. Funkcie sa majú volať tak, ako je zadané. Nemajú nič vypisovať.

Zrejme budete potrebovať vaše riešenie skúšať. V Pythone sa to robí tak, že na koniec vášho modulu dáte konštrukciu tohto typu

```
if __name__=="__main__":  
    print testovaci_vypis_1  
    print testovaci_vypis_2  
    print testovaci_vypis_3
```

Tým dosiahnete, že váš modul bude po spustení na príkazovom riadku vypisovať testovacie výpisy. Zároveň keď ho ja budem importovať v testoch, nebudú mi tam srašiť vaše testovacie výpisy.

2 Upravte myeval (5 bodov)

Upravte funkciu myeval tak, aby fungovala aj pre súčty a súčiny viac ako dvoch operandov.

```
>>> myeval(['+', 'x', 10, 'y'], {'x': 10, 'y': 20})  
40  
>>> myeval(['*', 'x', 10, 'y'], {'x': 10, 'y': 20})  
2000
```

3 Preved'te na reťazec (5 bodov)

Napíšte funkciu, `myinfix(f)`, ktorá vráti reťazec, ktorý reprezentuje funkciu danú prefixovou reprezentáciou `f`. Vrátený reťazec je normálny infixový výraz, taký ako píšeme v C alebo v Pythone. Môže mať pokojne zbytočne veľa zátvoriek, ide o korektnosť, nie krátkosť. Uvedomte si, že máte vrátiť reťazec, *nie* ho vypisovať. Pozor! Má to fungovať aj pre viac ako dva operandy v súčte alebo v súčine.

```
>>> myinfix(1)
'1'
>>> myinfix(1.5)
'1.5'
>>> myinfix("x")
'x'
>>> myinfix(["+", "x", "y", 1])
'(x+y+1)'
>>> myinfix(["*", ["+", "x", 1.7], ["/", "y", "z"]])
'((x+1.7)*(y/z))'
```

4 Sploštite (5 bodov)

Toto zadanie vyžaduje zamyslenie sa, ale nie je to nič strašné.

Napíšte funkciu `flatten(f)`, ktorá prevedie funkciu v prefixovej reprezentácii na funkciu v prefixovej reprezentácii ktorá je ekvivalentná pôvodnej, ale všetky vnorené súčty sú v nej sploštené. To znamená, implementujte transformáciu typu

$$(x + 1) + y \mapsto x + 1 + y$$

$$(1 + y) + (2 + x + z) + (17 + (x + y)) \mapsto 1 + x + 2 + x + z + 17 + x + y$$

```
>>> flatten(['+', 10, ['+', 'x', 'y']])
['+', 10, 'x', 'y']
>>> flatten(['+', ['+', 10, 20, ['+', 'x', 'y', ['+', 'z', 30]]]])
['+', 10, 20, 'x', 'y', 'z', 30]
```