

Operátory

Jiří Zacpal



KATEDRA INFORMATIKY
UNIVERZITA PALACKÉHO V OLOMOUCI

KMI/ZPP1 Základy programování v Pythonu 1

Řešení úkolu z minulé hodiny



```
cislo=1234
print(f"Číslo je: {cislo}")
print(f"První číslice je: {cislo//1000}")
print(f"Druhá číslice je: {(cislo//100)%10}")
print(f"Třetí číslice je: {(cislo//10)%10}")
print(f"Čtvrtá číslice je: {cislo%10}")
```

Operátory - základní pojmy

- **Priorita**
 - udává pořadí, ve kterém se operace provádějí
 - pořadí lze pozměnit použitím kulatých závorek
(závorky je ovšem dobré používat i tam, kde nejsou nutné, protože zvyšují přehlednost kódu)
- **Asociativita**
 - udává „směr“, ve kterém se vyhodnocují operace se stejnou prioritou (zleva nebo zprava)
 - lze opět ovlivnit použitím závorek
- **Arita**

Operátor umocňování

Operátor umocnění



$L^{**}P$

- L - levý operátor je mocněnec
- P – pravý operátor je mocnitel
- operátor je asociativní zprava

Příklad



- Napište výraz:

```
>>> 5**2
```

```
25
```

```
>>>
```

- Napište výraz:

```
>>> -1**2
```

```
-1
```

- `**` má vyšší prioritu než `-`, proto se výraz vyhodnotí jako tento: `-1(1**2)`. Pokud chceme umocnit `-1`, musíme použít operátor závorek:

```
>>> (-1)**2
```

```
1
```

```
>>>
```

- Napište výraz:

```
>>> 2**2**3
```

```
256
```

- `**` je asociativní zprava, takže se první vyhodnotí pravý operátor `**`. Pokud chceme první vyhodnotit levý operátor, musíme použít závorky:

```
>>> (2**2)**3
```

```
64
```

```
>>>
```

- Napište výraz, který pro zadané proměnné a, b a c vypočítá tuto hodnotu:

$$\frac{a^{b^{2+c}}}{4 \cdot c - 2^a}$$

- Řešení:

a=2

b=3

c=4

```
print ( ((a**b)**(2+c))/(4*c-2**a) )
```

Operátory porovnání a logické operátory

Logické hodnoty

- Součástí jazyka Python je Booleovský datový typ, který reprezentuje
- logickou pravdu (True) a logickou nepravdu (False).
- Příklad:

```
a = True  
b = False
```
- Ve své podstatě se jedná o celočíselný datový typ. Hodnota True je v jazyce Python chápána jako hodnota **1** a hodnota False jako hodnota **0**.

Logické operátory



- logický součin, logický součet, negace

a and b , a or b , not a

a	b	a and b
0	0	0
1	0	0
0	1	0
1	1	1

a	b	a or b
0	0	0
1	0	1
0	1	1
1	1	1

a	not a
0	1
1	0

Příklad



- Napište výrazy:

```
>>> True or False
```

```
True
```

```
>>> True and False
```

```
False
```

```
>>> not True
```

```
False
```

- Nejvyšší prioritu má not, poté and a nakonec or. Proto platí:

```
>>> not False and not True or True
```

```
True
```

```
>>> ((not False) and (not True)) or True
```

```
True
```

Operátory porovnání

- (je menší, je větší, je menší nebo rovno, je větší nebo rovno, rovná se, nerovná se)

$a < 1$, $2 > b$, $2 \leq 3$, $a \geq b$, $a == 1$, $b \neq 2$

- výsledkem logická hodnota True nebo False.

Příklad



- Napište tyto výrazy:

```
>>> 1 < 0
```

```
False
```

```
>>> 2 < 2
```

```
False
```

```
>>> 2 <= 2
```

```
True
```

```
>>> 3 > 2
```

```
True
```

```
>>> 4 >= 5
```

```
False
```

- Pozor na operátor ==. Neplést s operací =.

```
>>> 10==10
```

```
True
```

```
>>> 10=10
```

```
File "<stdin>", line 1
```

```
SyntaxError: can't assign to literal
```

```
>>>
```

Líné vyhodnocování



- V logických výrazech se vyhodnocuje pouze část výrazu (podle asociativity) nutná pro získání výsledku.
- Příklad:

```
>>> a=0
```

```
>>> a == 0 or 20 % a == 0
```

```
True
```

```
>>> 20 % a == 0 or a == 0
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
ZeroDivisionError: integer division or modulo by zero
```

```
>>>
```

Rozhodovací příkaz

- Máme rozhodnout, zda číslo a je menší než číslo b:

```
# vstup
```

```
a = 1
```

```
b = 2
```

```
# porovnání
```

```
result = a < b
```

```
# výstup
```

```
print(result)
```

- Rozhodněte zda čtyři zadaná čísla tvoří aritmetickou posloupnost. Program vytiskne True, jestliže ano, False, jestliže ne.
- Řešení:

```
c1=2
```

```
c2=4
```

```
c3=6
```

```
c4=8
```

```
d=c2-c1
```

```
result=(c3==c2+d and c4==c3+d)
```

```
print (result)
```


Standardní vstup

Vstup z klávesnice

- Pro čtení z klávesnice slouží funkce

`input(text)`

- text – text, který se zobrazí před čekáním na čtení z klávesnice.
- Příklad:

```
jmeno=input("Zadej své jméno:")  
print(f"Tvoje jméno je {jmeno}.")
```

- Z klávesnice se vždy načítá řetězec!
- Příklad:

```
cislo=input("Zadej číslo:")  
print(f"Druhá mocnina čísla je {cislo**2}.")
```

- chyba -> Musíme vstup převést na jiný datový typ!

Bodovaný úkol



- Pro zadaná přirozená čísla a , b a c . Rozhodněte, zda platí $a^2 + b^2 = c^2$.
- Jsou dány tři celá čísla a , b a c . Rozhodněte, zda a náleží do otevřeného intervalu (b, c) .
- Jsou zadány tři délky (nezáporná čísla). Rozhodněte, zda je možné sestrojit trojúhelník, jehož strany budou mít zadané délky.

Povoleno je použít jen ty příkazy z Pythonu, které byly dosud představeny.