

Algoritmické a numerické výpočty

5. a 6. cvičení

(vytvořeno 24. září 2020)

Základní aritmetické operace, vstup a výstup

- učebnice:
 - <http://howto.py.cz>
- online interpret jazyka Python:
 - <https://code.sololearn.com/#py>
- aritmetické operace:
 - proměnná, typy proměnných, přetypování
 - +, -, *
 - mocnění `a ** 2`
 - reálné dělení `a / b`
 - celočíselné dělení `a // b`
 - zbytek po dělení `a % b`
 - +=, -=, *=, ...
- výstup:
 - `print("retezec")`
 - `print(a, b, c)`
 - `print(a, b, c, sep=" ... ")`
 - `print(a, b, c, end="")`
- vstup:
 - `a = input("Zadej cislo: ")`
 - `a = int(input("Zadej cislo: "))`

Úloha 1. Napište program, který načte dvě celá čísla a vypíše jejich součet.

Úloha 2. Napište program, který načte koeficienty kvadratického polynomu $ax^2 + bx + c$ a pokusí se (bez ověřování, zda to jde) spočítat a vypsát oba reálné kořeny polynomu.

Úloha 3. Napište program, který načte poloměr koule a vypíše její objem a obsah jejího pláště.

Větvení programu

- syntaxe:

```
if <podminka>:
    <prikaz>
    <prikaz>
    <...>
elif <jina podminka>:
    <prikaz>
    <prikaz>
    <...>
else:
    <prikaz>
    <prikaz>
    <...>
```

- jednoduché podmínky:

- == ... rovnost
- != ... nerovnost
- > ... větší
- < ... menší
- >= ... větší nebo rovno
- <= ... menší nebo rovno
- a in [1,3,9] ... podmínka je splněna, pokud a je prvkem množiny {1, 3, 9}

- složené podmínky:

- and ... konjunkce
- or ... disjunkce
- not ... negace

Úloha 4. Napište program, který načte dvě celá čísla, zkontroluje, zda se nejedná o dělení nulou a vypíše podíl těchto čísel a jejich zbytek po dělení.

Úloha 5. Napište program, který načte celé číslo a vypíše jeho znaménko.

Úloha 6. Napište program, který načte reálné číslo, zkontroluje, zda není záporné a vypíše jeho odmocninu.

Úloha 7. Napište program, který načte koeficienty kvadratického polynomu $ax^2 + bx + c$. Pokud je diskriminant nezáporný, vypíše oba reálné kořeny polynomu.

Úloha 8. Napište program, který načte koeficienty kvadratického polynomu $ax^2 + bx + c$.

- Pokud je koeficient a nulový, vypíše jeden reálný kořen polynomu.

- Pokud je diskriminant kladný, vypíše oba reálné kořeny polynomu.
- Pokud je diskriminant nulový, vypíše jeden dvojnásobný reálný kořen polynomu.
- Pokud je diskriminant záporný, vypíše oba komplexně sdružené kořeny polynomu.

Cyklus while

- syntaxe:

```
while <podminka>:
    <prikaz>
    <prikaz>
    <...>
```

- speciální příkazy:
 - `break` ... ukončí cyklus a pokračuje prvním příkazem za cyklem
 - `continue` ... znovu zkontroluje podmínku cyklu
 - `pass` ... příkaz, který neudělá nic; vhodný, pokud chceme mít prázdné tělo cyklu

Úloha 9. Pomocí cyklu `while` naprogramujte celočíselné dělení.

Úloha 10. Napište program, který načítá nenulová čísla až do chvíle, než dostane první nulu. Poté program vypíše součet těchto čísel.

Úloha 11. Napište program, který načítá nenulová čísla až do chvíle, než dostane první nulu. Poté program vypíše:

- aritmetický průměr,
- minimum,
- maximum,
- počet minim,
- počet maxim.

Úloha 12. Program vyíše dělitele zadaného celého kladného čísla a napíše, zda se jedná o prvočíslo.

Seznamy a cyklus for

- seznamy:
 - `a=[1, -2, 7, 'x', "Eric Idle", -9.3]`
 - `c=a[3]`

- `a[1]="John Cleese"`
- `a.append(3.14)` ... přidá nový prvek na konec seznamu (seznam se tak zvětší o 1)
- `len(a)` ... vrátí délku seznamu
- `a=[]` ... vytvoření prázdného seznamu

- syntaxe cyklu `for`:

```
for <proměnná> in <seznam>:
    <prikaz>
    <prikaz>
    <...>
```

- místo seznamu lze použít příkaz `range()`, který iteruje mezi celými čísly

- `range(n)` ... čísla od 0 do $n - 1$
- `range(a, b)` ... čísla od a do $b - 1$
- `range(a, b, c)` ... čísla od a do $b - 1$ s tím, že vzdálenost mezi nimi je c

- **Příklad:** Program

```
for a in range(5):
    print(a)
```

vypíše:

```
0
1
2
3
4
```

- **Příklad:** Program

```
for a in range(3, 7):
    print(a)
```

vypíše:

```
3
4
5
6
```

- **Příklad:** Program

```
for a in range(2, 10, 2):  
    print(a)
```

vypíše:

```
2  
4  
6  
8
```

- **Příklad:** Program

```
for a in range(10, 2, -2):  
    print(a)
```

vypíše:

```
10  
8  
6  
4
```

Úloha 13. Napište program, který načte od uživatele seznam čísel (např. ukončený nulou) a vypíše ho.

Úloha 14. Pomocí algoritmu „Eratosthenovo síto” napište program, který vypíše všechna prvočísla, která jsou menší nebo rovna zadanému přirozenému číslu.

Funkce

- syntaxe:

```
def <název funkce>(<argumenty funkce>):  
    <prikaz>  
    <prikaz>  
    <...>
```

- speciální příkazy:

– `return <hodnota>` ... ukončí běh funkce a nastaví její návratovou hodnotu

Úloha 15. Napište funkci, která vrátí absolutní hodnotu čísla.

Úloha 16. Napište funkci, která vrátí aritmetický průměr čísel zadaných v seznamu.

Úloha 17. Napište funkci, která:

1. vrátí znaménko zadaného čísla,
2. vrátí součet dvou čísel,
3. vrátí průměr dvou čísel,
4. vrátí maximum ze dvou parametrů,
5. vrátí maximum ze tří parametrů (lze využít předchozí funkci),
6. vrátí n -tou mocninu zadaného čísla,
7. spočítá a vrátí faktoriál zadaného čísla, tento faktoriál vypočítá pomocí cyklu `for`,
8. vypíše prvky v části seznamu, která je vymezena nejnižším a nejvyšším indexem,
9. vrátí (součet, aritmetický průměr, nejmenší prvek, největší prvek) čísel v části seznamu, která je vymezena nejnižším a nejvyšším indexem,
10. převede číslo v binární soustavě do čísla v desítkové soustavě,
11. převede číslo v desítkové soustavě do čísla v binární soustavě.

Rekurze

Úloha 18. Napište funkci, která vrátí faktoriál zadaného čísla.

Úloha 19. Napište funkci, která vrátí n -tý člen Fibonacciho posloupnosti.

Úloha 20. Napište funkci, která vrátí n -tou mocninu zadaného čísla.

Úloha 21. Napište funkci, která vypíše návod pro řešení úlohy „Hanojské věže“.

Úloha 22. Napište funkci, která pomocí Euklidova algoritmu spočítá největší společný dělitel zadaných dvou čísel. Pro $a \geq b$ platí:

$$\text{nsd}(a,b) = \text{nsd}(b, a \% b)$$

Úloha 23. Napište funkci, která v setříděném seznamu nalezne zadanou hodnotu pomocí metody binárního půlení.

Určování výstupu

Úloha 24. Určete, jaký bude výstup programu:

```

def pow(x, n):
    if n == 0:
        return 1;
    else:
        return x * pow(x, n - 1)
print(pow(4, 0))
print(pow(4, 1))
print(pow(4, 2))
print(pow(7, 2))
print(pow(2, 3))
print(pow(2, 5))

```

Úloha 25. Určete, jaký bude výstup programu:

```

def div(a, b):
    if a < b:
        return 0;
    else:
        return 1 + div(a - b, b)
print(div(4, 1))
print(div(4, 2))
print(div(7, 2))
print(div(2, 3))

```

Úloha 26. Určete, jaký bude výstup programu:

```

def abc(n):
    if n < 100:
        print(n)
        abc(n + 20)
abc(51)
abc(40)
abc(500)

```

Úloha 27. Určete, jaký bude výstup programu:

```

def zzz(a, b):
    if a < b:
        return a
    else:
        return zzz(a - b, b)
print(zzz(3, 3))
print(zzz(9, 4))
print(zzz(123, 10))
print(zzz(1234, 100))

```

Úloha 28. Určete, jaký bude výstup programu:

```

def aaa(n):
    if n > 0:
        print(n)
        aaa(n // 3)
aaa(3)

```

```
aaa(10)
aaa(22)
```

Úloha 29. Určete, jaký bude výstup programu:

```
def bbb(n):
    if n == 1:
        return "1"
    else:
        return bbb(n - 1) + str(n) + bbb(n - 1)
print(bbb(1))
print(bbb(2))
print(bbb(3))
print(bbb(4))
```

Úloha 30. Určete, jaký bude výstup programu:

```
def ccc(n):
    if n == 1:
        return 1
    else:
        return 3 * ccc(n - 1) + 1
print(ccc(1))
print(ccc(2))
print(ccc(3))
print(ccc(4))
```

Úloha 31. Určete, jaký bude výstup programu:

```
def ddd(n):
    if n in [0, 1]:
        return n
    else:
        return 2 * ddd(n - 1) + 3 * ddd(n - 2)
print(ddd(2))
print(ddd(3))
print(ddd(4))
print(ddd(5))
```

Úloha 32. Určete, jaký bude výstup programu:

```
def eee(n):
    if n <= 1:
        return str(n)
    else:
        return eee(n // 2) + eee(n % 2)
print(eee(2))
print(eee(3))
print(eee(4))
print(eee(5))
```


Poznámky

- Odmocnina:

```
from math import *  
a = sqrt(2)
```

- Generátor náhodných čísel:

```
from random import *  
a = randint(1, 10)
```

- Rozdělení řetězce:

```
a = "a b c"  
b = a.split()
```

Řešení (některých) úloh

Řešení 1.

```
prvniCislo = int(input("Zadej prvni cislo: "))  
druheCislo = int(input("Zadej druhe cislo: "))  
soucet = prvniCislo + druheCislo  
print("Soucet techto cisel je:", soucet)
```

Řešení 2.

```
from math import *  
a = int(input("Zadej koeficient kvadratickeho clenu"))  
b = int(input("Zadej koeficient linearniho clenu"))  
c = int(input("Zadej koeficient absolutniho clenu"))  
D = b**2 - 4 * a * c  
x1 = (-b + sqrt(D)) / (2 * a)  
x2 = (-b - sqrt(D)) / (2 * a)  
print("Prvni koren:", x1)  
print("Druhy koren:", x2)
```

Řešení 4.

```
prvniCislo = int(input("zadej prvni cislo: "))  
druheCislo = int(input("zadej druhe cislo: "))  
if druheCislo == 0:  
    print("Delit nulou muze jenom Chuck Norris!")  
else:  
    podil = prvniCislo // druheCislo  
    zbytek = prvniCislo % druheCislo  
    print("Celociselný podíl techto čísel je:", podil)  
    print("Zbytek po dělení techto čísel je:", zbytek)
```

Řešení 5.

```
cislo = int(input("zadej cele cislo: "))
if cislo > 0:
    print("Toto cislo je kladne.")
elif cislo < 0:
    print("Toto cislo je zaporne.")
else:
    print("Toto cislo je nula.")
```

Řešení 8.

```
from math import *
a = int(input())
b = int(input())
c = int(input())
if a == 0:
    if b == 0:
        print("Tohle ti počítat nebudu.")
    else:
        print("x =", -c / b)
else:
    D = b**2 - 4 * a * c
    if D >= 0:
        x1 = (-b + sqrt(D)) / (2 * a)
        x2 = (-b - sqrt(D)) / (2 * a)
        print("x1 =", x1)
        print("x2 =", x2)
    else:
        real = -b / (2 * a)
        imag = sqrt(-D) / (2 * a)
        print("x1 = ", real, " + ", imag, "i", sep="")
        print("x2 = ", real, " - ", imag, "i", sep="")
```

Řešení 6.

```
from math import *
a = int(input("Zadej realne cislo: "))
if a >= 0:
    print("Odmocnina tohoto cisla je:", sqrt(a))
else:
    print("Cislo je zaporne, to si odmocnuj sam!")
```

Řešení 9.

```
prvniCislo = int(input("zadej prvni cislo: "))
druheCislo = int(input("zadej druhe cislo: "))
if druheCislo == 0:
    print("Delit nulou muze jenom Chuck Norris!")
else:
    podil = 0
    while prvniCislo >= druheCislo:
```

```

    prvniCislo -= druheCislo
    podil += 1
zbytek = prvniCislo
print("Celociselny podil techto cisel je:", podil)
print("Zbytek po deleni techto cisel je:", zbytek)

```

Řešení 10.

```

soucet = 0
while True:
    nacteneCislo = int(input("Zadej cislo (nula to ukonci, bacha!): "))
    if nacteneCislo == 0:
        break
    soucet += nacteneCislo
print("Soucet vsech tebou zadanych cisel je:", soucet)

```

Řešení 13.

```

seznam = []
while True:
    nacteneCislo = int(input("Zadej cislo (nula to ukonci, bacha!): "))
    if nacteneCislo == 0:
        break
    seznam.append(nacteneCislo)
print("Vypisu seznam najednou:", seznam)
print("Vypisu seznam postupne:")
for prvek in seznam:
    print(prvek)
print("Vypisu seznam postupne, a to i s indexama:")
for i in range(len(seznam)):
    print(i, "->", seznam[i])

```

Řešení 14.

```

limit = int(input("Do jakého čísla chceš ty prvočísla vypsat?: "))
jePrvocislo = limit * [True]
for cislo in range(2, limit):
    if jePrvocislo[cislo]:
        print(cislo)
        for i in range(2*cislo, limit, cislo):
            jePrvocislo[i] = False

```

Řešení 15.

```

def absHod(x):
    if x < 0:
        return -x
    else:
        return x

```

Řešení 16.

```

def vratPrumer(seznam):
    soucet = 0
    for prvek in seznam:
        soucet += prvek
    return soucet / len(seznam)

```

Řešení 18.

```

def faktorial(x):
    if x == 1:
        return 1
    else:
        return x * faktorial(x - 1)

```

Řešení 19.

```

def fib(n):
    if n == 1:
        return 1
    elif n == 2:
        return 1
    else:
        return fib(n - 2) + fib(n - 1)

```

Řešení 20.

```

def mocnina(x, n):
    if n == 0:
        return 1
    else:
        return x * mocnina(x, n-1)

```

Řešení 21.

```

def hanoj(odkud, kam, velikost):
    if velikost == 1:
        print(odkud, "->", kam)
    else:
        tretí = 6 - odkud - kam
        hanoj(odkud, tretí, velikost - 1)
        hanoj(odkud, kam, 1)
        hanoj(tretí, kam, velikost - 1)

```

Řešení 22.

```

def nsd(a, b):
    z = a % b
    if z == 0:
        return b
    else:
        return nsd(b, a % b)

```

Řešení 23.

```
def hledej(seznam, od, do, hledany):
    if od > do:
        return None
    uprostred = (od + do) // 2
    if seznam[uprostred] == hledany:
        return uprostred
    elif seznam[uprostred] > hledany:
        return hledej(seznam, od, uprostred - 1, hledany)
    else:
        return hledej(seznam, uprostred + 1, do, hledany)
```

Řešení 24.

1
4
16
49
8
32

Řešení 25.

4
2
3
0

Řešení 26.

51
71
91
40
60
80

Řešení 27.

0
1
3
34

Řešení 28.

3
1
10
3
1
22
7
2

Řešení 29.

1
121
1213121
121312141213121

Řešení 30.

1
4
13
40

Řešení 31.

2
7
20
61

Řešení 32.

10
11
100
101