

Uvod u programiranje



2023/24



Ponavljanje

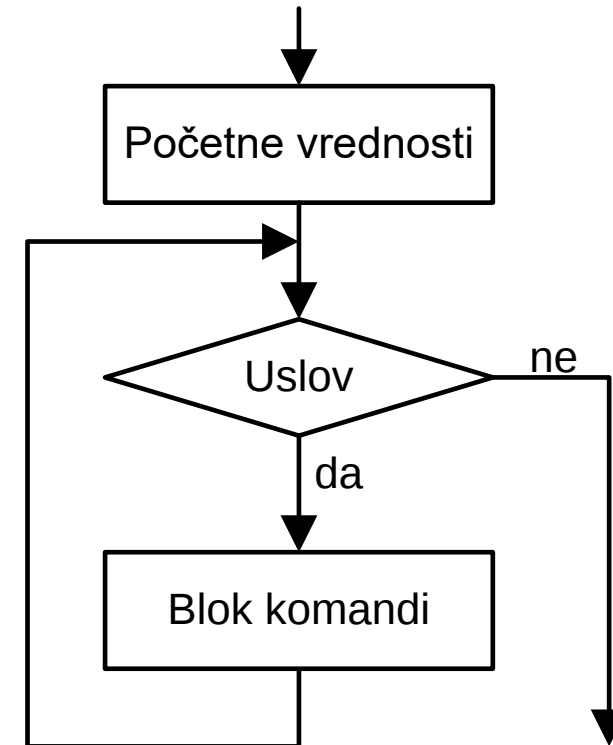


Ponavljanje naredbi - WHILE

- Python pruža mehanizam da se jedna ili više naredbi ponovi određen broj puta ili dok je neki uslov ispunjen

```
while (uslov):  
    naredba 1  
    naredba 2
```

- Naredbe se izvršavaju sve dok je uslov tačan.
- Uslov ne mora biti u zagradi.



Ponavljanje naredbi - WHILE

- Poređenje - želimo 5 puta da šampamo 'Zdravo!'

```
br = 0
if br < 5:
    print('Zdravo!')
    br = br + 1
```

Zdravo!

```
br = 0
while br < 5:
    print('Zdravo!')
    br = br + 1
```

Zdravo!
Zdravo!
Zdravo!
Zdravo!
Zdravo!

Ponavljanje naredbi - WHILE

- Šta radi dati deo koda?

```
n=5
while n > 0:
    print(n)
    n = n-1
print("Gotovo!")
```

šta se dešava bez ovog reda?

Ponavljanje naredbi - WHILE

- Odrediti koren svih celih brojeva od 1 do n ($n > 1$) i prikazati tabelarno.

```
import math
```

```
n=int(input())
```

```
br=1
```

```
while br <= n:
```

```
    print(br, '    ', math.sqrt(br))
```

```
    br=br+1
```

5	
1	1.0
2	1.4142135623730951
3	1.7320508075688772
4	2.0
5	2.23606797749979

Ponavljanje naredbi - WHILE

- Izračunati zbir unetih brojeva. Brojeve unositi dok se ne unese 0.

```
x = int(input("Unesi broj: "))
s = 0
while x != 0:
    s = s + x
    x = int(input("Unesi broj: "))
print("Zbir unetih brojeva je ", s)
```

```
Unesi broj: 3
Unesi broj: 5
Unesi broj: 4
Unesi broj: 7
Unesi broj: 0
Zbir unetih brojeva je 19
```

Primer

- Koliko puta će se ponoviti ciklus i šta će biti vrednosti promenljivih **a**, **b** i **s** posle izvršenja ovog niza naredbi?

```
a = 1
b = 1
while a+b<8:
    a = a+1
    b = b+2
s = a+b
```

Primer

- Šta je rezultat sledećeg koda?

```
i = 1
while i <= 6:
    print(5*i, '\t', end=' ')
    i = i + 1
print()
```

Ponavljanje naredbi - FOR

- Na osnovu prethodnog, **while** predstavlja najopštiji način za realizovanje ponavljanja algoritamskih koraka.
- Može se koristiti kako za situacije kada broj ponavljanja nije poznat, tako i za zadatke u kojima se on unapred zna (brojačke petlje).
- Međutim, za brojačke petlje pogodnije je koristiti drugi mehanizam – naredbu **for**

```
for brojač in opseg :  
    telo_petlje
```

Ponavljanje naredbi - FOR

```
for i in range(5):  
    print("Zdravo")
```

Zdravo
Zdravo
Zdravo
Zdravo
Zdravo

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

0
1
2
3
4

Ponavljanje naredbi - FOR

- Funkcija **range** generiše sekvencu celih brojeva

Poziv funkcije	Generisana sekvenca
<code>range(n)</code>	za $n > 0$: $0, 1, 2, 3, \dots, n-1$
<code>range(start, stop)</code>	za $stop > start$: $start, start+1, start+2, \dots, stop-1$
<code>range(start, stop, korak)</code>	za $stop > start$ i $korak > 0$: $start, start+korak, start+2*korak, \dots, n$ važi $stop - korak \leq n < stop$
<code>range(start, stop, korak)</code>	za $stop < start$ i $korak < 0$: $start, start+korak, start+2*korak, \dots, n$ važi $stop - korak \geq n > stop$

Ponavljanje naredbi - FOR

```
a = int(input("Unesi ceo broj "))
b = int(input("Unesi ceo broj "))
for i in range(a,b+1):
    print (i)
print()
```

```
for i in range(5,25,5):
    print (i)
print()
```

```
for i in range(25,4,-5):
    print (i)
```

```
Unesi ceo broj 3
Unesi ceo broj 9
3
4
5
6
7
8
9

5
10
15
20

25
20
15
10
5
```

Ugnježdene petlje

- Ispis tablice množenja

```
for x in range(1, 11):  
    for y in range(1, 11):  
        print(x*y)
```

```
for x in range(1, 11):  
    for y in range(1, 11):  
        print(x*y, ' ', end='')  
    print()
```

```
1  2  3  4  5  6  7  8  9 10  
2  4  6  8 10 12 14 16 18 20  
3  6  9 12 15 18 21 24 27 30  
4  8 12 16 20 24 28 32 36 40  
5 10 15 20 25 30 35 40 45 50  
6 12 18 24 30 36 42 48 54 60  
7 14 21 28 35 42 49 56 63 70  
8 16 24 32 40 48 56 64 72 80  
9 18 27 36 45 54 63 72 81 90  
10 20 30 40 50 60 70 80 90 100
```

Komande BREAK i CONTINUE

- Za prekid izvršavanje petlje, bez obzira na ispunjenost uslova koristi se komanda **BREAK**

```
for i in range(1,10):  
    if i == 3:  
        break  
    print (i)
```

1
2

- Komandom **CONTINUE** se prekida izvršavanje trenutne iteracije (preskače se deo koda) i prelazi se na sledeću iteraciju

```
for broj in range(1,10):  
    if broj == 3:  
        continue  
    print (broj)
```

Primer

- Napisati algoritam i program kojim se za uneti prirodan broj n ($n \geq 1$) i realan broj x , izračunava broj S na sledeći način:

$$\sum_{i=1}^n \frac{x^i}{i!}$$

```
n=int(input("Unesi n: "))
x=float(input("Unesi x: "))
s=0
f=1
for i in range(1,n+1):
    f*=x/i
    s+=f

import math as m
for i in range(1,n+1):
    s+=x**i/m.factorial(i)

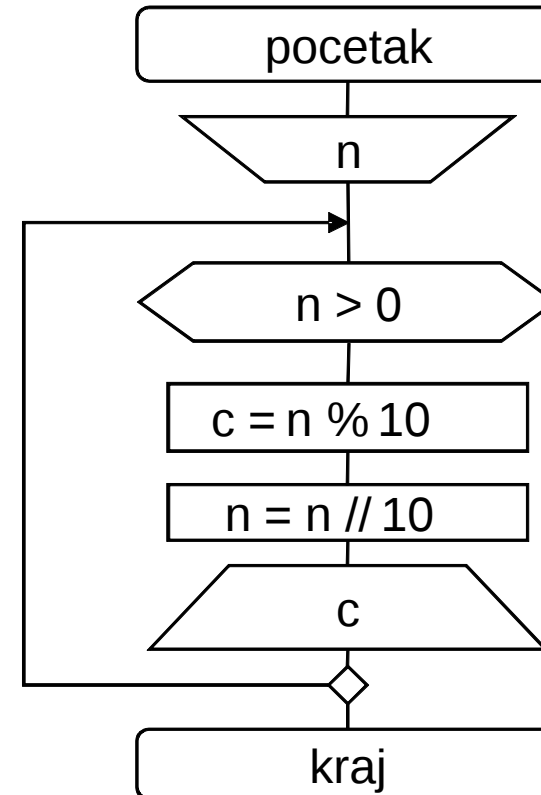
print(s)
```

Primer

- Napisati algoritam i program koji za uneti ceo broj **n** ispisuje njegove cifre.

```
n = int(input("Unesi broj: "))
```

```
while n > 0:  
    c = n % 10  
    n = n // 10  
    print(c)
```



Primer

- Šta je rezultat rada sledećeg programa?

```
def printMultiples(n):  
    i = 1  
    while i <= 6:  
        print (n*i, '\t', end='')  
        i = i + 1  
    print()
```

```
i = 1  
while i <= 6:  
    printMultiples(i)  
    i = i + 1
```