



## Osnove 2

## Osnove 2

Python urejevalnik se nahaja na strani:

<https://www.w3resource.com/python-exercises/python-basic-exercises.php#EDITOR>

Vhodne podatke preberete iz standardnega vhoda z ukazom `input`.

1. **Igra ugibanja** Uganite število, ki ga je izbral računalnik. Računalnik izbere naključno število med 1 in 100. In vpraša uporabnika, da vpiše svoje število. Ko uporabnik vpiše število, ki ga je izbral mu računalnik odgovori:
  - a) Previsoko, v primeru, če je bilo vpisano število večje od števila, ki ga izbral računalnik.
  - b) Prenizko, v primeru, če je bilo vpisano število manjše od števila, ki ga je izbral računalnik.
  - c) Uganil si, v primeru, ko je bilo vpisano število enako številu, ki ga je izbral računalnik.

Računalnik sprašuje uporabnika toliko časa, dokler uporabnik ne ugame števila, ki ga je izbral računalnik.

Napišite program, kjer bo računalnik ugibal število, ki ste si ga zamislili. Kakšna bi bila optimalna strategija.

2. **Vsota elementov gnezdenega seznama** Zapiši program, ki poišče vsoto vseh elementov v gnezdenem seznamu.  
Primer `l=[1,2,3, [4,5, [9,2, [4,4,2]]],3,4], [2,4,3],2]`  
Rezultat je 54
3. **Faktoriel** Zapišite funkcijo, ki za dano naravno število  $n$  izračuna produkt

$$n! = 1 \times 2 \times \dots \times n.$$

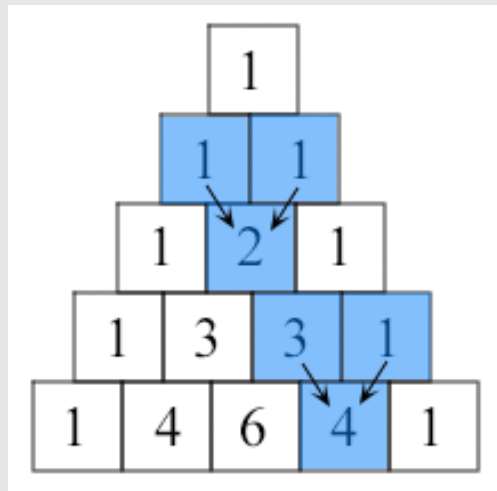
Funkcijo zapišite rekurzivno:

$$f(1) = 1, \quad f(n) = n f(n - 1)$$

4. **Fibonaccijevo zaporedje** Zapišite  $n$  členov Fibonaccijevega zaporedja. Definicija Fibonaccijevega zaporedja  $F_n$ :

$$F_0 = 0, \quad F_1 = 1, \quad F_{n+1} = F_n + F_{n-1}$$

5. **Paskalov trikotnik** Program naj izpiše prvih  $n$  vrstic Paskalovega trikotnika.



Slika 1: Pascalov trikotnik

## Rešitve 2

### Igra ugibanja

```
#!/usr/bin/env python3
import random

n = random.randint(1,101)
poskusi = 0

while True:
    poskusi += 1
    m = int(input('Vpiši število: '))
    if m < n:
        print('Prenizko')
    elif m > n:
        print('Previsoko')
    else:
        print('Uganil si število poskusov je bilo %d.' % poskusi)
        break
```

### Igra ugibanja 2

```
#!/usr/bin/env python3
import random

def bisekcija():
    minimum = 0
    maksimum = 101
    print('Zamisli si število med 1 in 100 in pritisni <enter>')
    input()
    while True:
        ugib = random.randint(minimum + 1, maksimum - 1);
        odg = input('Ali je to število %d ' % ugib)
        if odg == '>':
            minimum = ugib
        elif odg == '<':
            maksimum = ugib
        else:
            print(ugib)
            break

if __name__ == '__main__':
    bisekcija()
```

## Vsota elementov gnezdenega seznama

```
#!/usr/bin/env python3

def nestsum(s):
    suma = 0
    for i in s:
        if type(i) is int:
            suma += i
        else:
            suma += nestsum(i)
    return suma

if __name__ == '__main__':
    s = eval(input('Vpiši gnezden seznam -> '))
    print(nestsum(s))
```

## Faktoriel rekurzivno

```
#!/usr/bin/env python3

def faktoriel(n):
    if n < 0: return(0)
    elif n == 0: return(1)
    else: return n*faktoriel(n-1)

if __name__ == '__main__':
    s = input('Vpiši naravno število: ')
    try:
        n = int(s)
        print(faktoriel(n))
    except:
        print('Napaka')
```

## Faktoriel iterativno

```
def faktoriel(n):
    res=1;
    if n==0: return res
    if n>0:
        for i in range(n): res*=i+1
    else: res = 0
    return(res)
```

## Fibonacci rekurzivno

```
def bad_fibonacci(n):
    if n <= 1:
        return n
    else:
        return bad_fibonacci(n-2) + bad_fibonacci(n-1)

def good_fibonacci(n):
    if n <= 1:
        return (n,0)
    else:
        (a, b) = good_fibonacci(n-1)
    return (a+b, a)
```

## Fibonacci iterativno

```
def fibonacci(n):
    a = 0;
    b = 1;
    if n==0:
        return 0
    elif n==1:
        return 1
    else
        for i in range(n):
            a, b = b, a + b
    return(b)
```

## Paskalov trikotnik

```
def paskal(n)
    l=[1]
    print(l)
    for _ in range(n):
        l = [i + j for i, j in zip([0] + l, l + [0])]
        print(l)
```