

RAČUNALNO RAZMIŠLJANJE I PROGRAMIRANJE



PROGRAMSKI JEZIK PYTHON

Verzija za instalaciju na računalo: <https://www.python.org/downloads/>

Online interaktivno sučelje: <https://www.python.org/shell/>

TIPOVI PODATAKA I VARIJABLE

- Osnovni tipovi podataka u Pythonu:

Cijeli brojevi – **int**

Decimalni brojevi – **float**

Logički – **bool**

Znakovni niz - **string**



Brojčani tip podataka

ZNAKOVNI NIZ - string

- Sastoji se od niza znakova poredanih zadanim redoslijedom
- Svaki znak unutar niza ima svoje mjesto u nizu (indeks)
- Prvi element niza u programskom jeziku Python ima uvijek indeks 0

ime= " Mate"

ime =	M	a	t	e
indeks	0	1	2	3

Pojedini element niza možemo dohvatiti na način da uz ime niza unutar uglatih zagrada upišemo njegov indeks:
npr. ime [2] dohvatit će znak 't'

Primjer

1. Napiši program koji ispisuje koliko ima znakova u riječi informatika , te koliko puta se znak 'a' ponavlja.

```
rijec= 'informatika'
brojac=0
for i in range (len(rijec)):
    if rijec[i]=='a':
        brojac=brojac+1 #ako je ispunjen uvjet uvećaj brojač za 1
print ('Broj znakova u riječi',rijec,'je:',len(rijec))
print ('Slovo a se ponavlja',brojac,'puta.')
```

Primjer

2. Napiši program koji ispisuje koliko ima riječi u rečenici.

```
recenica= 'informatika 7. razred'  
brojac=0  
for i in range (len(recenica)):  
    if recenica[i]==' ': #ispitujemo da li postoji razmak  
        brojac=brojac+1 #ako je ispunjen uvjet uvećaj brojač za 1  
print ('Rečenica:',recenica,'ima',brojac+1,'riječi.')
```

NIZOVI PODATAKA - LISTE

- Niz - jednodimenzionalna linearna struktura istih ili različitih tipova podataka
- Elementima liste pristupamo indeksima (isto kao i za znakovni niz)

niz = [2,-3, 4, 0]

niz=	2	-3	4	0
indeks	0	1	2	3

Pojedini element niza možemo dohvatiti na način da uz ime niza unutar uglatih zagrada upišemo njegov indeks:
npr. niz [1] dohvatit će broj -3

FUNKCIJE I METODE ZA RAD S LISTAMA

- **len** (ime niza) -> funkcija koja vraća veličinu (broj elemenata) niza
- **min** (ime niza) -> funkcija koja vraća najmanji element niza
- **max** (ime niza) -> funkcija koja vraća najveći element niza
- **sum** (ime niza) -> funkcija koja vraća zbroj svih elemenata liste
- **sorted**(ime niza) -> funkcija koja sortira elemente niza od manjih prema većim

```
>>> niz = [-3, 4, 0, 9, 7]
>>> len (niz)
5
>>> min (niz)
-3
>>> max (niz)
9
>>> sum (niz)
17
>>> sorted (niz)
[-3, 0, 4, 7, 9]
```


FUNKCIJE I METODE ZA RAD S LISTAMA

- **imeniza.append(x)** -> metoda koja dodaje element x na kraj postojeće liste

```
>>> niz=[1,2,3]
>>> x=-5
>>> niz.append(x)
>>> niz
[1, 2, 3, -5]
>>> prazan_niz=[]
>>> x=10
>>> prazan_niz.append(x)
>>> prazan_niz
[10]
```

Primjer

3. Napiši program koji ispisuje prosječnu ocjenu od ocjena : 5,4,5,3,3,4,5,4,5,4,5,2,5

```
ocjene = [5,4,5,3,3,4,5,4,5,4,5,2,5]
#1.Pristup
suma=0
for i in range (len(ocjene)):
    suma=suma+ocjene[i]

print ('Prosjek ocjena:', suma/len(ocjene))

#2.Pristup - korištenjem funkcije sum()
print ('Prosjek ocjena:', sum(ocjene)/len(ocjene))
```

Primjer

4. Napiši program (bez korištenja ugrađenih funkcija) koji ispisuje najmanji i najveći element niza.

```
niz=[7,5,0,10,-3,2]
#####Najmanji element niza
minimum =niz[0]
for i in range (6):
    if minimum > niz[i]:
        minimum=niz[i]
print ('Najmanji element niza:',minimum)
#####Najveći element niza
maksimum =niz[0]
for i in range (6):
    if maksimum < niz[i]:
        maksimum=niz[i]
print ('Najveći element niza:',maksimum)
```

Funkcije u Pythonu (potprogrami)

- **Potprogram (funkcija)** je zasebni dio programskog *koda* koji se može pozivati s različitih mjesta u programu.
- Svaki potprogram ima naziv i argumente
- Argumenti su vrijednosti koji se prosljeđuju potprogramu kao parametri, ulazne vrijednosti za izračun.

Definiranje funkcije

```
def naziv_funkcije (parametri)
    blok naredbi
    return vrijednost
```

Vrste



1. Bez ulaznih parametara i bez vraćanja vrijednosti

```
def pravokutnik ():  
    a=int(input('Unesi duljinu stanice a:'))  
    b=int(input('Unesi duljinu stanice b:'))  
    povrsina=a*b  
    print ('Površina pravokutnika je:',povrsina)  
  
for i in range (5):  
    pravokutnik()
```

2. Bez ulaznih parametara i s vraćanjem vrijednosti

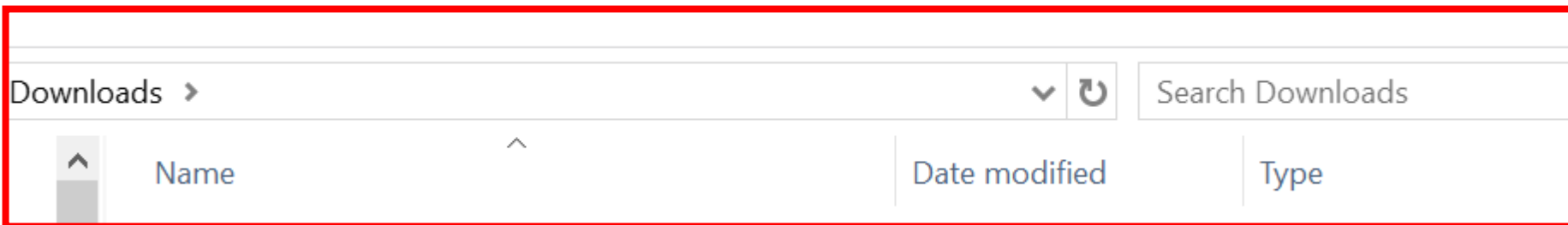
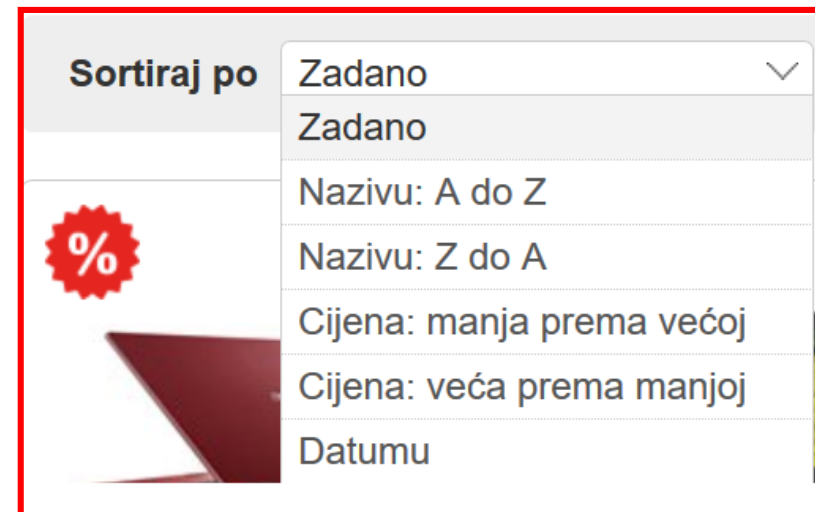
```
def pravokutnik ():  
    površina=a*b  
    return površina  
  
for i in range (5):  
    a=int(input('Unesi duljinu stranice a:'))  
    b=int(input('Unesi duljinu stranice b:'))  
    print ('Površina pravokutnika je:',pravokutnik())
```

SORTIRANJE PODATAKA

SORTIRANJE

- Jedan od važnijih koncepata računarstva
- Sortiranje podrazumijeva uređenje niza podataka, npr:
 - uređenje brojeva : silazno (veće prema manjem), uzlazno (manje prema većem)
 - uređenje niza znakova npr. abecedni redosljed

Primjer primjene sortiranja

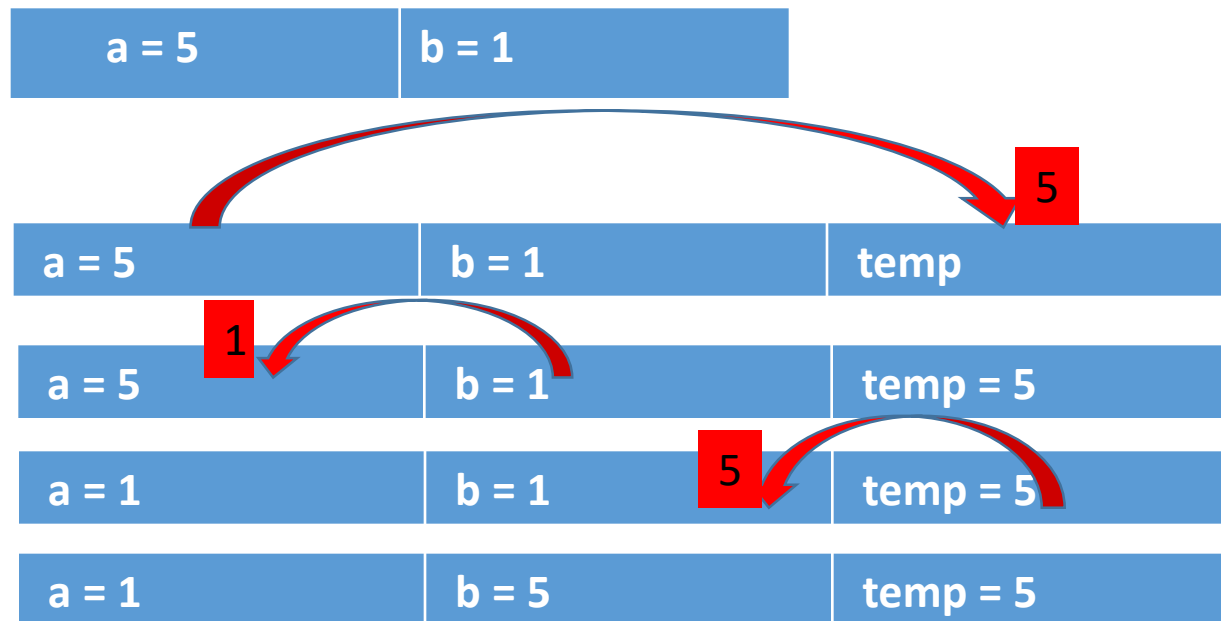


1. Primjer sortiranja (dva broja)

```
a=5  
b=1  
#Sortiraj uzlazno  
print (a,b)
```

RJEŠENJE

```
a=5  
b=1  
#Sortiraj uzlazno  
if a>b:  
    temp=a  
    a=b  
    b=temp  
print (a,b)
```



kraj

2. Primjer sortiranja (tri broja)

```
def sortiraj (a,b,c) :  
    if a>b:  
        temp=a  
        a=b  
        b=temp  
    if a>c:  
        temp=a  
        a=c  
        c=temp  
    if b>c:  
        temp=b  
        b=c  
        c=temp  
    print (a,b,c)  
  
a,b,c=map(int,input('Unesi 3 broja:').split())  
sortiraj (a,b,c)
```

SORTIRANJE ZAMJENOM (Bubble sort)

- Uspoređuju se susjedni elementi od početka niza do njegovog kraja
- SORTIRANJE SILAZNO:
 - ako je element na poziciji i manji od elementa na poziciji $i+1$ napravi zamjenu (*eng. swap position*)
- SORTIRANJE UZLAZNO:
 - ako je element na poziciji i veći od elementa na poziciji $i+1$, napravi zamjenu (*eng. swap position*)



Klikni na sliku

Bubble sort – primjer br.1

```
niz = [5, 4, 0, -3, 7]

print ('Nesortirani niz:', niz)

for i in range (len(niz)) :

    for j in range (len(niz)-1) :
        if niz[j]>niz[j+1]:
            temp=niz[j]
            niz[j]=niz[j+1]
            niz[j+1]=temp

print ('Sortirano uzlazno:', niz)
```

Bubble sort – primjer br.2

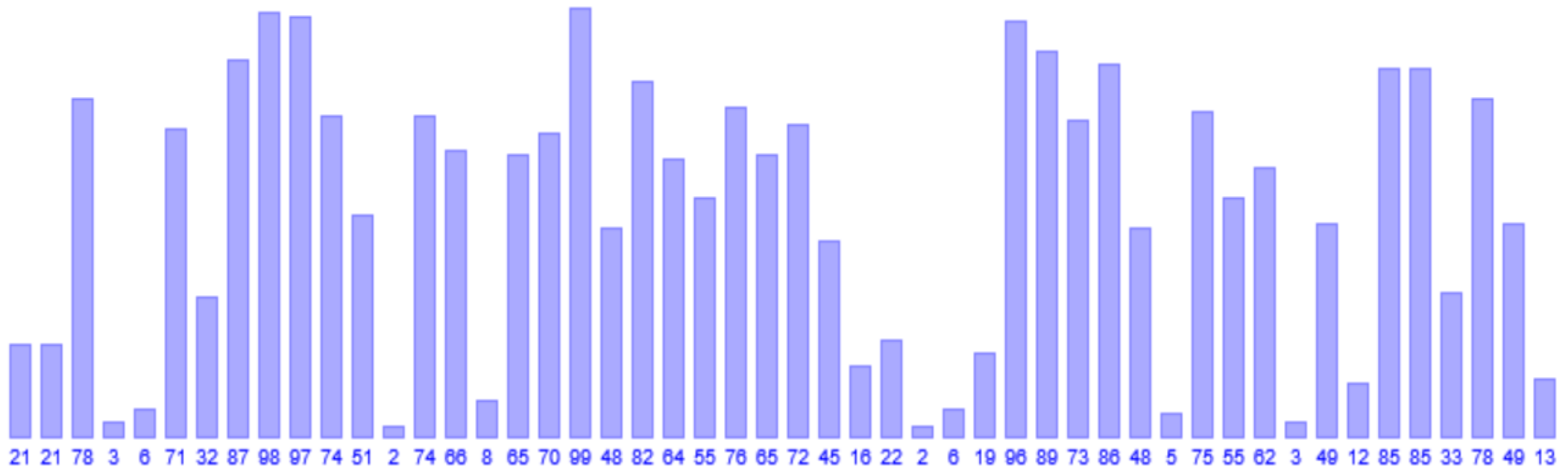
Stvori niz od **n** slučajnih parnih i neparnih brojeva u rasponu od broja **1** do broja **100**. Niz parnih brojeva sortiraj uzlazno, a neparnih silazno.

```
###Potprogrami###  
  
def uzlazno(): #funkcija za sortiranje uzlazno  
    for i in range (len(parni)):  
        for j in range (len(parni)-1-i):  
            if parni[j]>parni[j+1]:  
                temp=parni[j]  
                parni[j]=parni[j+1]  
                parni[j+1]=temp  
    print ('Parni uzlazno=',parni)  
  
def silazno(): #funkcija za sortiranje silazno  
    for i in range (len(neparni)):  
        for j in range (len(neparni)-1-i):  
            if neparni[j]<neparni[j+1]:  
                temp=neparni[j]  
                neparni[j]=neparni[j+1]  
                neparni[j+1]=temp  
    print ('Neparni silazno=',neparni)
```

```
#####GLAVNI PROGRAM#####  
from random import*  
n=int(input('Unesi broj elemenata niza:')) #unosimo broj elemenata niza  
parni=[] #deklaracija niza  
neparni=[] #deklaracija niza  
  
for i in range(n):  
    a=randint(1,100) #generiranje i spremanje slučajnog broja od 1 do 100  
    if a%2==0: #ispitujemo da li je slučajan broj 'a' paran  
        parni.append(a)  
    else:  
        neparni.append(a)  
silazno() #pozovi funkciju  
uzlazno() #pozovi funkciju
```

ALGORITMI SORTIRANJA

Randomize Array Insertion Sort Selection Sort Bubble Sort Quick Sort Merge Sort Shell Sort Change Size



Klikni na sliku