

Praktikum iz programiranja 1



2024/25



Lista listi



Lista listi

- Svaki podataka u listi je lista

`[[1, 2], [3]]` - liste u okviru liste nemaju jednak broj elemenata

`[[1, 2], [2, 5]]` - lista ima dve liste i svaka od njih po dva elementa

- Pristup elementu unutar liste:

```
A = [[1, 2], [2, 5]]
```

```
print(A)
```

```
B = A[0]
```

```
print(B)
```

```
k = A[0][1]
```

```
print(k)
```

```
[[1, 2], [2, 5]]
```

```
[1, 2]
```

```
2
```

Matrica

- Lista listi se koristi za reprezentaciju matrica u Python-u
- **Matrica** dimenzije $m \times n$ je šema brojeva koja se zapisuje kao:

$$A = \begin{bmatrix} a_{00} & a_{01} & \dots & a_{0n-1} \\ a_{10} & a_{11} & \dots & a_{1n-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m-10} & a_{m-11} & \dots & a_{m-1n-1} \end{bmatrix}$$

vrste

kolone

The diagram shows a matrix A enclosed in large square brackets. Inside the brackets, the elements are arranged in rows and columns. The first row contains a_{00} , a_{01} , an ellipsis, and a_{0n-1} . The second row contains a_{10} , a_{11} , an ellipsis, and a_{1n-1} . The third row contains a vertical ellipsis, another vertical ellipsis, a diagonal ellipsis, and a vertical ellipsis. The last row contains a_{m-10} , a_{m-11} , an ellipsis, and a_{m-1n-1} . To the right of the matrix, the word 'vrste' (rows) is written, with three arrows pointing to the first, second, and last rows of the matrix. Below the matrix, the word 'kolone' (columns) is written, with three arrows pointing to the first, second, and last columns of the matrix.

- Matrica kod koje je $m = n$ zove se **kvadratna matrica**.

Primer

- Napisati algoritam i program kojim se za unete prirodne brojeve m i n koji predstavljaju dimenzije matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama.
- Napisati funkciju **ZbirPoObodu(mat,m,n)** koja računa i vraća zbir elemenata te matrice koji su raspoređeni po obodu.

Primer

```
m=int(input("Unesi broj vrsta"))
n=int(input("Unesi broj kolona"))

matrix = [[0 for i in range(n)] for i in range(m)] #matrica
dimenzije nxm sa nulama

for i in range(m):
    for j in range(n):
        matrix[i][j]=int(input())

print(matrix)
print(matrix[:,0])
print(ZbirPoObodu(matrix,m,n))
```

Primer

```
def ZbirPoObodu(e,m,n):  
    s=sum(e[0][:])+sum(e[m-1][:])#+sum(e[1:m-1][0])+sum(e[1:m-1][n-1])  
  
    for i in range(1,m-1):  
        s+=e[i][0]+e[i][n-1]  
    return s
```

Primer

- Napisati algoritam i program kojim se za uneti prirodan broj n , unose celi brojevi koji predstavljaju elemente matrice dimenzije $n \times n$ po vrstama.
- Napisati funkciju **ZbirParnihVrsta(mat,n)** koja vraća niz koji predstavlja zbir elemenata matrice koji pripadaju parnim vrstama

Primer

```
def ZbirParnihVrsta(e,n):  
    b=[sum(e[i][:]) for i in range(0,n,2) ]  
    return b
```

```
n=int(input("Unesi dimenziju"))
```

```
matrix = [[0 for i in range(n)] for i in range(n)] #matrica dimenzije nxn  
sa nulama
```

```
for i in range(n):  
    for j in range(n):  
        matrix[i][j]=int(input())
```

```
print(ZbirParnihVrsta(matrix,n))
```

Razni zadaci



Zadatak 1

Katarina je odlučila da svojoj drugarici pošalje šifrovanu poruku, koja sadrži samo slova engleske abecede, cifre i interpunkcijske znake. Svako slovo će šifrovati posebno na osnovu narednih pravila. Mala slova se šifruju velikim slovima tako što se slovo a šifruje slovom Z, slovo b šifruje slovom Y, c slovom X itd., sve do slova y koje se šifruje slovom B i z koje se šifruje slovom A. Velika slova se šifruju potpuno analogno - od A koje se šifruje sa z do Z koje se šifruje sa a. Ostali karakteri se ne menjaju.

Sa standardnog ulaza unosi **se jedna linija teksta**.

Na standardni izlaz ispisati **šifrovani tekst**.

Ulaz:

Zdravo svima.

Izlaz:

aWIZEL HERNZ.

Zadatak 1

```
def pretvoriKarakter(c):  
    noviKar = ord(c) # dobijanje ASCII vrednosti od karaktera  
    if (c >= 'a' and c <= 'z'):  
        inkrement = 25 - 2 * (ord(c) - ord('a'))  
        noviKar = ord(c) + inkrement - 32  
        # -32 da bismo prebacili u veliko slovo  
    elif (c >= "A" and c <= "Z"):  
        inkrement = 25 - 2 * (ord(c) - ord('A'))  
        noviKar = ord(c) + inkrement + 32  
        # +32 da bismo prebacili u malo slovo  
  
    return chr(noviKar) # dobijanje karaktera od ASCII vrednosti  
  
s = input()  
for slovo in s:  
    novoSlovo = pretvoriKarakter(slovo)  
    print(novoSlovo, end='')
```

Zadatak 2

- Data su dva niza iste dužine $\mathbf{v}$ i $\mathbf{t}$. Niz $\mathbf{t}$ sadrži vremenske trenutke [s], a niz $\mathbf{v}$ brzine [m/s] tela izmerene u tim vremenskim trenucima. Napisati funkciju koda koji računa ukupni pređeni put i prosečnu brzinu tela na tom putu.

Zadatak 2

```
def PredjeniPut(v,t):  
    s=0  
    t1=0  
    for i in range(0,len(v)):  
        s+=v[i]*(t[i]-t1)  
        t1=t[i]  
    vp=s/t[-1]  
    print(vp)  
    print(s)
```

```
PredjeniPut([20,18,22,17],[1200,1680,2900,3700])
```

```
19.751351351351353
```

```
73080
```

Zadatak 3

- Napisati funkciju **Polinom(c, x)** koja računa vrednost polinoma $c[0] + c[1] \cdot x + c[2] \cdot x^2 + \dots + c[n-1] \cdot x^{n-1}$.
- Može li se zadatak rešiti bez petlji?

```
def Polinom(c,x):  
    p = 0  
    for i in range(len(c)):  
        p+=(c[i]*x**i)  
    return p
```

```
def Polinom(c,x):  
    a1=[x**i for i in range(len(c))]  
    d=[b[0]*b[1] for b in list(zip(c, a1))]  
    return sum(d)
```

```
print(Polinom([1,-2,3],2))
```

9

Zadatak 4

- Data je kvadratna jednačina kao string oblika " $ax^2+bx+c=0$ ", recimo " $-3x^2+2x-1=0$ ". Napisati funkciju koja kao argument uzima ovakav string, a vraća rešenja kvadratne jednačine.

```
def KvJed(s):  
    i=s.find("x")  
    if i==0:  
        a=1  
    else:  
        a=int(s[0:i])  
    s=s[i+3:len(s)]  
    i=s.find("x")  
    if i==0:  
        b=1  
    else:  
        b=int(s[0:i])
```


Zadatak 4

```
s=s[i+1:len(s)]
i=s.find("=")
if i==0:
    c=0
else:
    c=int(s[0:i])
print(a,b,c)
d=b**2-4*a*c
if d<0:
    print("Resenja su kompleksni brojevi")
elif d==0:
    print(-b/(2*a))
else:
    x1=(-b+m.sqrt(d))/(2*a)
    x2=(-b-m.sqrt(d))/(2*a)
    print(x1,x2)
```

KvJed("-3x^2+2x-1=0")

KvJed("x^2-2x-2=0")

Zadatak 5

- Napisati algoritam i program kojim se za unete prirodne brojeve m i n ($1 \leq m, n \leq 20$) koji predstavljaju dimenzije matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama i prirodni brojevi l i k ($1 \leq l, k \leq m$), a zatim razmenjuje vrednosti elementima u vrstama l i k .

Zadatak 5

```
n=int(input("Unesi broj vrsta"))
m=int(input("Unesi broj kolona"))
l=int(input("Unesi vrstu l"))
k=int(input("Unesi vrstu k"))
```

```
matrix = [[0 for i in range(m)] for i in range(n)]
#matrica dimenzije nxm sa nulama
```

```
for i in range(n):
    for j in range(m):
        matrix[i][j]=int(input())
```

```
for j in range(m):
    matrix[k][j], matrix[l][j] = matrix[l][j], matrix[k][j]
```

```
print(matrix)
```

```
Unesi broj vrsta3
Unesi broj kolona4
Unesi vrstu l1
Unesi vrstu k2
3
4
5
6
7
8
9
10
11
12
1
2
[[3, 4, 5, 6], [11, 12, 1,
2], [7, 8, 9, 10]]
>>>
```