

Praktikum iz programiranja 1



2024/25



Liste



Liste

- Lista je niz podataka proizvoljnog tipa
- Svaki podataka u listi se naziva **element**

[1, 4, 17, 256, 3, 59, 45] - lista celih brojeva

['april', 'jun', 'septembar', 'novembar'] - lista stringova

[] - prazna lista

['Marina', 28, 101, 'PMF'] - lista elemenata različitog tipa

[1, 4, [7, 6, [13]], [9, 5]] - lista čiji elementi su liste

Generisanje liste

```
zivotinje=['pas','macka','mis','majmun','slon']  
print (zivotinje)  
['pas', 'macka', 'mis', 'majmun', 'slon']
```

```
lista=list(range(1,10))  
print (lista)  
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
lista=list(range(1,10,2))  
print (lista)  
[1, 3, 5, 7, 9]
```

Elementi liste, indeksiranje

- Elementim liste se pristupa preko rednog broja pozicije – indeksa
- Indeks prvog elementa u listi je 0

```
lista = [10, 9, 8, 7, -1, -2, -3]
print (lista[2], lista[-1])
print (lista[1:5])
print (lista[1:5:2])
print (lista[12])
```

8 -3

[9, 8, 7, -1]

[9, 7]

IndexError: list index out of range

Promena vrednosti

- Elementima liste je moguće promeniti vrednost

```
lista = [2, 4, 6, 8]
```

```
lista[1] = 0
```

```
lista[-1] = 12
```

```
print (lista)
```

```
[2, 0, 6, 12]
```

Spajanje listi

- Dve liste se spajaju operatorom +

```
A = [1,2,3,4,5]
```

```
B = [6,7,8,9]
```

```
C = A + B
```

```
print (C)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
print (A + B[1:-1])
```

```
[1, 2, 3, 4, 5, 7, 8]
```

```
B = 'Jovana'
```

```
C = A + B
```

```
TypeError: can only concatenate list (not "str") to list
```

```
print (C)
```

```
B = ['Jovana']
```

```
C = A + B
```

```
print (C)
```

```
[1, 2, 3, 4, 5, 'Jovana']
```

Ponavljanje elemenata liste

- Elementi liste se mogu ponavljati određen broj puta operatorom *

```
C = [0]
```

```
A = [1, 2, 3]
```

```
B = 3 * A
```

```
D = 4 * C
```

```
print(B, D)
```

```
[1, 2, 3, 1, 2, 3, 1, 2, 3] [0, 0, 0, 0]
```


Funkcije za rad sa listama

- Dužina liste, odnosno broj elemenata liste – **len**

```
lista = ['jedan', 2, 3, 4, 5, [-2, -4, -5]]  
print (len(lista))  
print (len(lista[5]))  
print (len(lista[0]))  
print (len(lista[1]))
```

6

3

5

TypeError: object of type 'int' has no len()

Funkcije za rad sa listama

- Operator **in**

```
lista=[1,2,3,4,5]
print(5 in lista)
print(8 in lista)
if (not 10 in lista):
    print(10,"nije u listi")
else:
    print(10,"jeste u listi")
```

True

False

10 nije u listi

Funkcije za rad sa listama

- Dodavanje elemenata na kraj liste – **append**, **extend**

```
lista = [1,2,3]
```

```
lista.append([4,5])
```

```
print (lista)           [1, 2, 3, [4, 5]]
```

```
lista.append(6)
```

```
print (lista)           [1, 2, 3, [4, 5], 6]
```

```
lista.extend([7,8])
```

```
print (lista)           [1, 2, 3, [4, 5], 6, 7, 8]
```

```
lista.extend(9)          TypeError: 'int' object is not iterable
```

Funkcije za rad sa listama

- Dodavanje elemenata na željenu poziciju – **insert**

```
A = [1, 2, 3, 4, 5]
```

```
A.insert(2,0)
```

```
print (A)
```

```
[1, 2, 0, 3, 4, 5]
```

```
A.insert(4,['a','b','c'])
```

```
print (A)
```

```
[1, 2, 0, 3, ['a', 'b', 'c'], 4, 5]
```

Funkcije za rad sa listama

- Minimalni i maksimalni element liste – **min, max**

```
A = [1,13,-2,6,101,28,44,17,-27]
```

```
print (min(A), max(A))
```

```
A = ['Danijela', 'Anabela', 'Marko', 'Katarina', 'Nikola']
```

```
print (min(A), max(A))
```

```
A = [2, 3, 4, [1, 6], 5]
```

```
print (min(A), max(A))
```

-27 101

Anabela Nikola

TypeError: '<' not supported between instances of 'list' and 'int'

Funkcije za rad sa listama

- Broj ponavljanja elementa – **count**

```
A = [2, 3, 4, 1, 7, 2, 3, 1, 1, 0, 9, 5]
```

```
print (A.count(1))
```

```
odeljenje = ['Ana', 'Marko', 'Jovana', 'Darko', 'Jovana']
```

```
print (odeljenje.count('Jovana'))
```

```
A = [2, [3, 4], 1, [3, 4], 3, 1, 1, 0, 9, 5]
```

```
print (A.count(3), A. count([3,4]))
```

3

2

1 2

Funkcije za rad sa listama

- Brisanje elementa iz liste – **del, pop, remove**

```
A=[1, 2, 3, 4, 5, 6, 7, 4, 8, 9]
```

```
del(A[3])
```

```
print (A)
```

```
[1, 2, 3, 5, 6, 7, 4, 8, 9]
```

```
A.pop(1)
```

```
print (A)
```

```
[1, 3, 5, 6, 7, 4, 8, 9]
```

```
A.pop()
```

```
print (A)
```

```
[1, 3, 5, 6, 7, 4, 8]
```

```
A.remove(5)
```

```
print (A)
```

```
[1, 3, 6, 7, 4, 8]
```

Funkcije za rad sa listama

- Sortiranje i okretanje elemenata liste – **sort, reverse**

```
A = [2,6,1,9,3,5,4]
```

```
A.sort()
```

```
print (A)                [1, 2, 3, 4, 5, 6, 9]
```

```
A.reverse()
```

```
print (A)                [9, 6, 5, 4, 3, 2, 1]
```

```
A = ['Pera', 'Laza', 'Mika', 'Aca']
```

```
A.sort()
```

```
print (A)                ['Aca', 'Laza', 'Mika', 'Pera']
```


Liste i petlje

```
for voce in ["banana", "jabuka", "kruska"]:  
    print ("Volim da jedem " + voce)
```

Volim da jedem banana

Volim da jedem jabuka

Volim da jedem kruska

Liste i stringovi

- Koliko data rečenica ima reči?

```
recenica = "Danas je divan dan"  
lista_reci = list(recenica.split())  
print(lista_reci)  
print(len(lista_reci))
```

`recenica.split()` – od date rečenice pravi listu čiji su elementi dobijeni razvajanjem rečenice po jednom (ili više) blanko karakteru

Funkcija `split()` se može primeniti direktno na ulaz:

```
lista_reci = list(input().split())
```

Liste i stringovi

- Metoda `join()` poziva se nad listom `stringova`, formirajući jedan string od elemenata liste

```
L = ['Tata', 'kaže', 'U', 'laži', 'su', 'kratke' , 'noge']  
recenica=' '.join(L)  
print(recenica)
```

Tata kaže U laži su kratke noge

Formiranje liste brojeva

- Kako formirati listu **brojeva** na prethodno opisan način?

```
brojevi= list(input().split())
```

```
print(brojevi)      # kog tipa su elementi liste?
```

```
1 2 3.3 5.6
```

```
['1', '2', '3.3', '5.6']
```

Ako želimo brojeve potrebno je svaki element liste pretvoriti u broj. Na koji način?

```
brojevi= list(map(int, input().split()))
```

ili

```
for i in range(len(brojevi)):
    brojevi[i] = float(brojevi[i])
```

```
[1, 2, 3.3, 5.6]
```

Formiranje liste brojeva

- Formirati listu koja za dat ceo broj **n** sadrži kvadrate brojeva od 0 do **n**, ne računajući **n**?

```
n = int(input())  
L1 = [ i**2 for i in range(n) ]  
print (L1)
```

5

```
[0, 1, 4, 9, 16]
```

```
>>>
```

Filtriranje

- U mnogim zadacima je potrebno od date liste formirati novu koja sadrži elemente koji zadovoljavaju neki dati uslov.

```
lista = [1, 2, 3, 4, 5, 6, 7, 8]  
nova_lista = [x for x in lista if x % 2==0]  
print(nova_lista)
```

```
[2, 4, 6, 8]  
>>>
```

Liste kao argumenti funkcija

- Lista je promenljive prirode i prilikom prenošenja u funkciju može doći do (ne)željenog sporednog efekta

```
def f(lista):  
    lista.append(100) # sporedni efekat  
    lista = [] # lista se redefiniše kao lok. prom.  
    return lista
```

```
lista = [1, 2]  
print('lista pre poziva', lista)  
print('rezultat f(lista):', f(lista))  
print('lista posle poziva', lista)  
lista pre poziva [1, 2]  
rezultat f(lista): []  
lista posle poziva [1, 2, 100]
```

Liste kao argumenti funkcija

- Ukoliko ne želimo da promenimo listu:
- lista se, pri navodjenju stvarnih parametara u pozivu funkcije, može kopirati u novu listu

```
lista = [1, 2]
print('lista pre poziva', lista)
print('rezultat f(lista):', f(lista[:]))
print('lista posle poziva', lista)
lista pre poziva [1, 2]
rezultat f(lista): []
lista posle poziva [1, 2]
```


Primer

Napisati funkciju koja datu listu sortira u nerastućem poretku. U glavnom delu programa uneti listu od n celih brojeva a zatim korišćenjem napisane funkcije sortirati elemente liste.

```
def sortiranje(lista):  
    lista.sort()  
  
A=[]  
n=int(input())  
for i in range(n):  
    A.insert(i,int(input()))  
  
print(A)  
sortiranje(A)  
print(A)
```

Primer

```
def sortiranje(lista):  
    for i in range(len(lista)-1):  
        for j in range(i+1, len(lista)):  
            if (lista[i] >= lista[j]):  
                lista[i], lista[j] = lista[j], lista[i]  
  
A=[]  
n=int(input())  
for i in range(n):  
    A.insert(i,int(input()))  
  
print(A)  
sortiranje(A)  
print(A)
```

Primer

Napisati algoritam i program kojim za unetu listu od n celih brojeva.

- a) Štampa elemente liste u obrnutom redosledu.
- b) Rotira elemente liste za:
 - Jedno mesto ulevo
 - Jedno mesto udesno

Primer

a)

```
A=[]  
n=int(input())  
for i in range(n):  
    A.insert(i,int(input()))
```

```
for i in range(n-1,-1,-1):  
    print(A[i])
```

Primer

b)

```
b=A[0]
```

```
for i in range(0,n-1):  
    A[i]=A[i+1]
```

```
A[n-1]=b  
print(A)
```

```
b=A.pop(0)  
A.insert(n-1,b)  
print(A)
```

Primer

b)

```
b=A[n-1]
```

```
for i in range(n-1,0,-1):  
    A[i]=A[i-1]
```

```
A[0]=b  
print(A)
```

```
b=A.pop()  
A.insert(0,b)  
print(A)
```

Primer

Za unet prvi član aritmetičkog niza, i razliku aritmetičkog niza, napraviti prvih 20 elemenata aritmetičkog niza i odštampati ih.

Primer

```
print('Unesite prvi clan aritmetickog niza')
a1 = int(input())
print('Unesite razliku aritmetickog niza')
razlika = int(input())
niz = [a1]
for i in range(19):
    a1 = a1 + razlika
    niz = niz + [a1]
print('Aritmeticki niz je:')
print(niz)
```


Primer

Napisati funkciju koja vraća listu prvih 100 Fibonačijevih brojeva. Nakon toga filtrirati listu tako da sadrži samo parne elemente.

Izlaz:

[0, 2, 8, 34, 144, 610, ..., 51680708854858323072,
218922995834555169026]

Primer

```
def fibonacci():  
    l = list() #pravi praznu listu  
  
    l.append(0)  
    l.append(1)  
  
    for i in range(2, 100):  
        l.append(l[i - 2] + l[i - 1])  
    return l  
  
def paranBroj(x):  
    if(x % 2 == 0):  
        return True  
    return False  
  
lista = fibonacci()  
#Filtriranje - prvi nacin  
novaLista = [x for x in lista if x % 2 == 0]  
#Filtriranje - drugi nacin  
novaLista = list(filter(paranBroj, lista))  
  
print(novaLista)
```