

# Uvod v Python

Seznami (lists), matrike in  
terke (tuples)

# Seznami v Pythonu

- Podatkovna struktura
- Uporabljamo jih za shranjevanje različnih delčkov informacij pod imenom ene spremenljivke
- Seznane v Pythonu lahko shranjujemo tudi različne tipe podatkov

# Matrike v Pythonu

- Za namene naše vaje bomo pisali matrike na “osnoven” način
- Za matematično računanje z matrikami se navadno uporablja Python knjižnico Numpy, katero pa v okviru teh vaj ne bomo obravnavali

# Primer 2.1:

## Matrika – Način 1

```
A = [1, 2, 3, 4, 5,
      6, 7, 8, 9, 10,
      11, 12, 13, 14, 15,
      16, 17, 18, 19, 20,
      21, 22, 23, 24, 25]

print(A) #Zgornji zapis matrike je v bistvu seznam
#Dostopanje do elementov
print(len(A)) #Št. elementov v seznamu/matriki A
print(A[5]) #Izpis petega elementa seznama (prva vrstica,
prvi stolpec)
i = 5 #Št. vrstic
j = 5 #Št. stolpcev
print("A(vrstica, stolpec)")
m = int(input("Vstavi vrstico: "))
n = int(input("Vstavi stolpec: "))
print("A(", m, ", ", n, ") = ", A[(m-1)*j+(n-1)], sep='')
#-1 zato, ker se v pythonu indeksi začnejo z 0 in ne 1
```

# Primer 2.2:

## Seštevanje matrik

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

```
B = [2, 4, 6,  
     8, 10, 12,  
     14, 16, 18]
```

```
print("A = ", A, "\n", "B = ", B, sep='')  
print("A+B=", end='')
```

```
#A+B=C
```

```
C = [0]*len(A) #Definiramo novo matriko pravilne velikosti
```

```
for i in range(len(A)):  
    C[i] = A[i] + B[i]  
print(C)
```

# Naloga

Kvadrirajte matriko A (Matematični izraz:  $[A]^2$ )

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

# Rešitev

```
A = [1, 2, 3,
     4, 5, 6,
     7, 8, 9]
print("A = ", A, "\n", sep='')

A2 = [0]*len(A) #Definiramo novo matriko pravilne
              #velikosti
for i in range(len(A)):
    A2[i] = A[i]**2
print(A2)
```

# Terke (tuples) v Pythonu

- Terka je zaporedje nespremenljivih Python objektov (intiger, nizi znakov...)
- Glavna razlika med sezname in terkami je, da se terk ne da spreminjati medtem ko sezname se lahko spreminja



# Primer 3.1:

## Preprosti primeri terk

```
tuple_1 = ("Konstrukcijske", "Tehnike", 2015, 2016)
tuple_2 = (1, 2, 3, 4, 5 )
tuple_3 = "a", "b", "c", "d"
tuple_4 = (50,) #Za izpis enega elementa kot del tupla
               #je potrebno dodati vejico

tuple_prazen = ()

print(tuple_1)
print(tuple_2)
print(tuple_3)
print(tuple_4)
print(tuple_prazen)
```

# Primer 3.2:

Dostop do vrednosti v terkah (indeksi elementov)

```
tuple_1 = ("Konstrukcijske", "Tehnike", 2015, 2016)
tuple_2 = (1, 2, 3, 4, 5, 6, 7 )
#Izpis prvega elementa(indeks = 0!!):
print("tuple_1[0]: ", tuple_1[0])
#Izpis elementov z indeksom od 1 do 5:
print("tuple_2[1:5]: ", tuple_2[1:5])
#Izpis elementov z indeksom od 1 do 8 z razmakom 2:
print("tuple_2[1:8:2]: ", tuple_2[1:8:2])
#Izpis elementov z indeksom 0 (od začetka tupla ) do 6 z
razmakom 2:
print("tuple_2[:6:2]: ", tuple_2[:6:2])
#Izpis elementov z indeksom 2 (-5 z desne proti levi) do
8 (do konca tupla) z razmakom 3:
print("tuple_2[-5::3]: ", tuple_2[-5::3])
```

# Primer 3.3:

## Osnovne operacije s terkami

```
tuple_1 = (1, 2, "c")
tuple_2 = ("OK", )
dolžina = len(tuple_1) #Št. Elementov v terki
združevanje = tuple_1 + tuple_2 #Nastane nov tuple
ponavljanje = tuple_2 * 4
membership = 2 in tuple_1 #Ali je 2 element tuple_1?
print("Dolžina tuple_1: ", dolžina)
print("Združevanje tuple_1 + tuple_2: ", združevanje)
print("Ponavljanje OK*4: ", ponavljanje)
print("\n\"Članstvo\" 3 v tuple_1: ", membership,
      "\n-----")
for x in (1, 2, "c"): print (x) #Izpis vseh elementov
print("-----")
for x in združevanje: print (x)
```

# Primer 3.4:

## Brisanje terk

```
#Brisanje elementov iz terk ni mogoče

tuple_1 = ("Konstrukcijske", "Tehnike", 2015, 2016)
print (tuple_1)
del tuple_1;
print ("After deleting tuple_1 : ")
print (tuple_1) #Dobimo napako "name tuple_1 is not
                #defined", ker smo terko izbrisali
```

# Slovarji v Pythonu

- So podatkovna struktura podobna seznamom.
- Od seznamov se razlikujejo tako, da se do posameznih elementov slovarjev dostopa prek "ključa" oz. ključne besede in ne prek indeksa, kot je to pri seznamih.
- Ključ je lahko ali niz ali pa številka. Slovarje definiramo z zaviti oklepaji "{" in "}".