

Praktikum iz programiranja 3



2023/24



MathPlotLib

- **Matplotlib** je Pythonova biblioteka koja se koristi za vizuelizaciju podataka.

```
pip install matplotlib
```

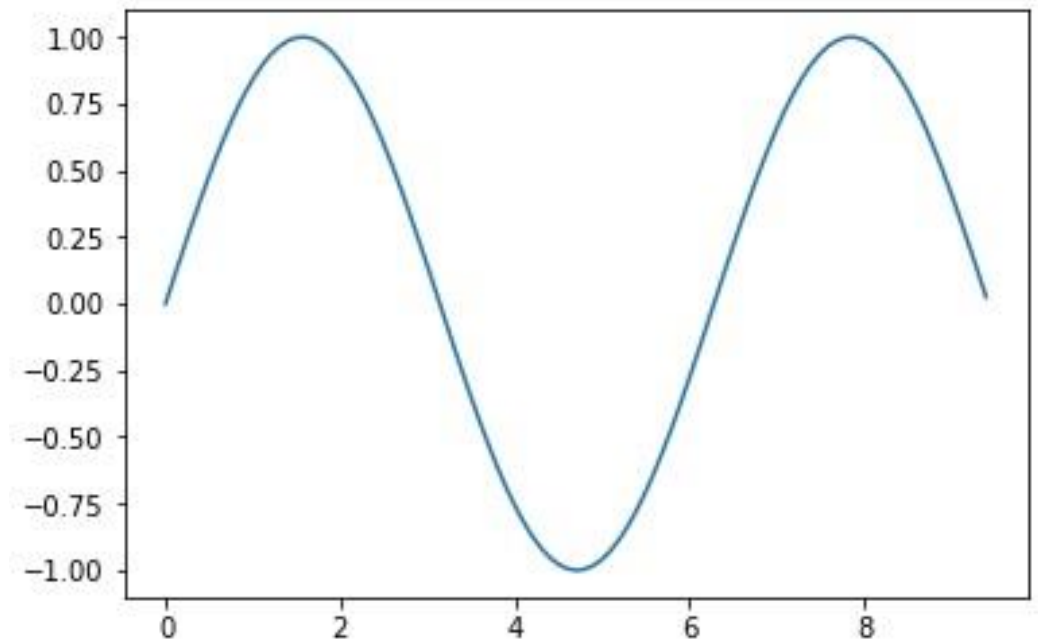
```
import matplotlib.pyplot as plt
```

Mathplotlib

```
import numpy as np
import matplotlib.pyplot as plt

x = np.arange(0, 3 * np.pi, 0.1)
y = np.sin(x)

plt.plot(x, y)
plt.show()
```



Mathplotlib

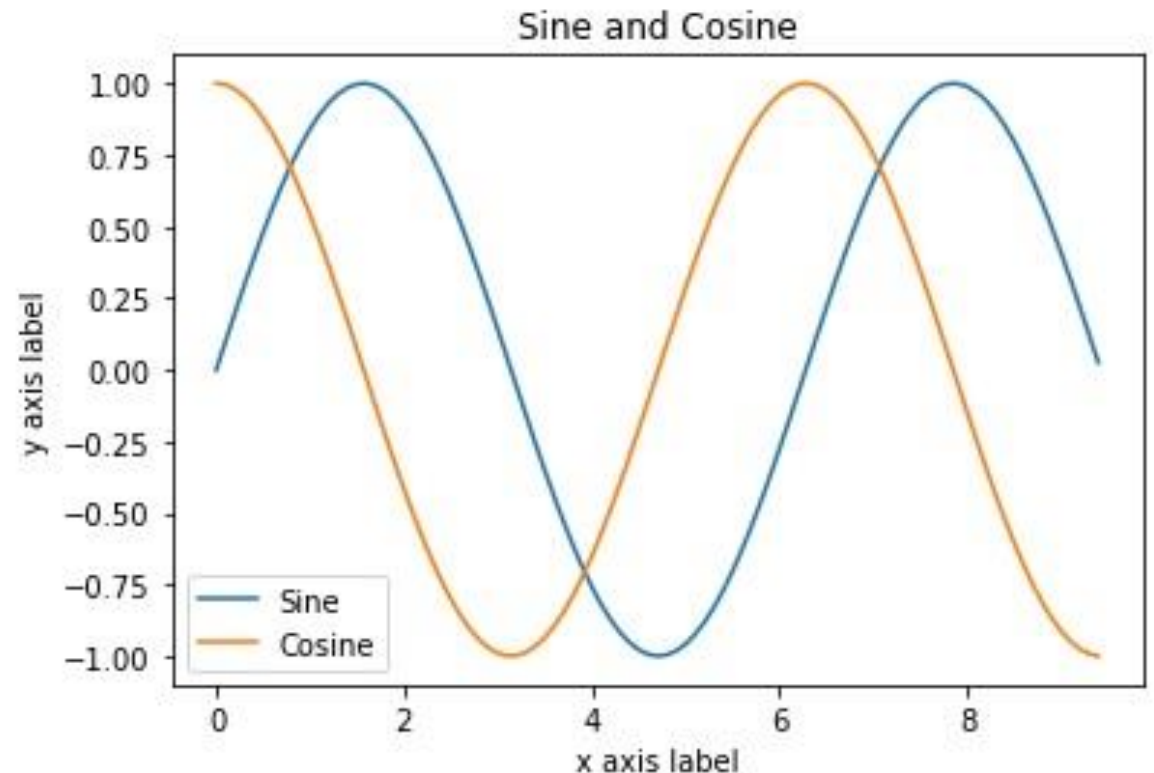
```
x = np.arange(0, 3 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)
```

```
plt.plot(x, y_sin)
plt.plot(x, y_cos)
```

```
plt.xlabel('x axis label')
plt.ylabel('y axis label')
```

```
plt.title('Sine and Cosine')
plt.legend(['Sine', 'Cosine'])
```

```
plt.show()
```



Mathplotlib

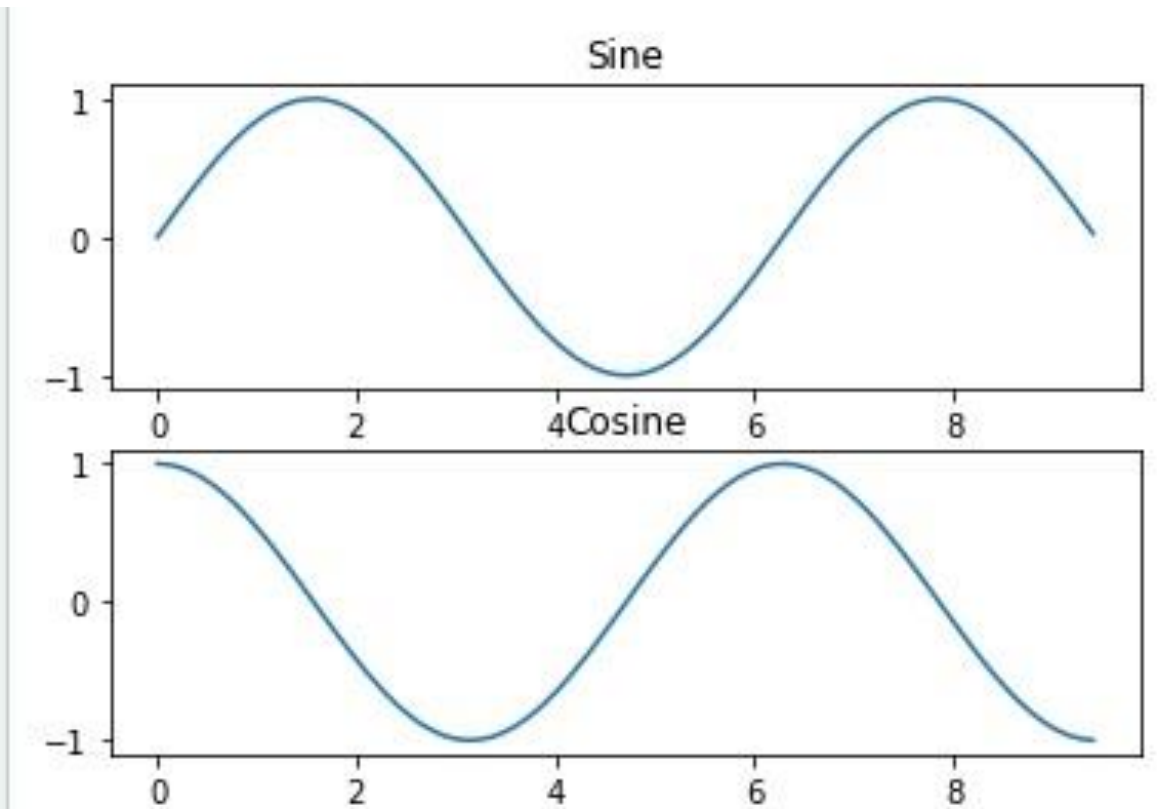
```
x = np.arange(0, 3 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)
```

```
plt.subplot(2, 1, 1)
```

```
plt.plot(x, y_sin)
plt.title('Sine')
```

```
plt.subplot(2, 1, 2)
plt.plot(x, y_cos)
plt.title('Cosine')
```

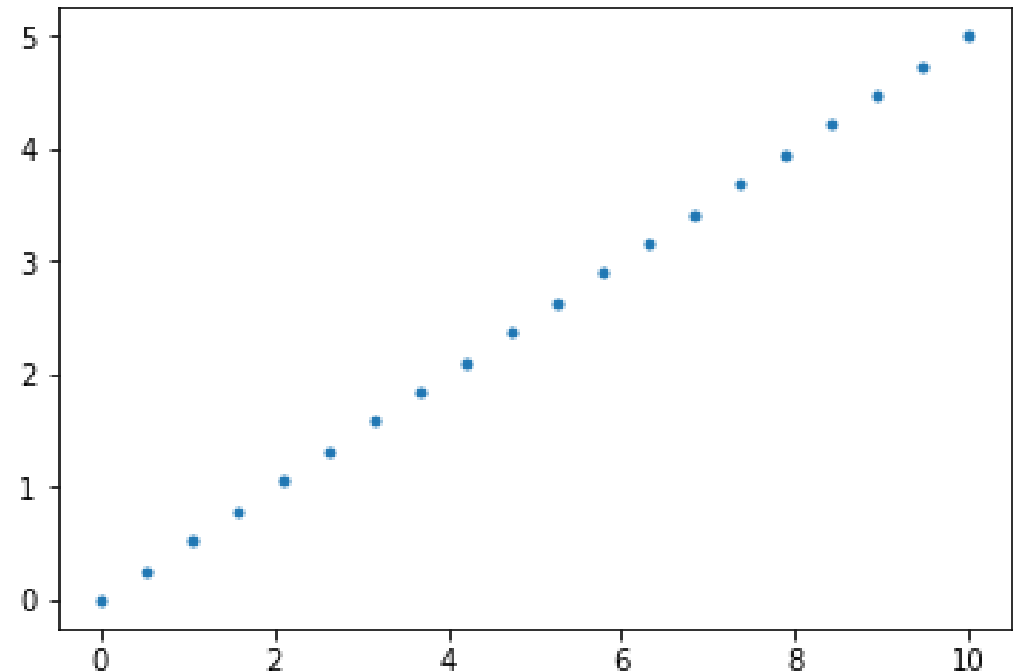
```
plt.show()
```



Mathplotlib

Napisati program kojim se iscrtavaju tačke (x_i, y_i) , $1 \leq i \leq 20$, pri čemu x_i predstavlja niz ekvidistantnih koordinata u intervalu $[0,10]$, a y_i niz ekvidistantnih koordinata u intervalu $[0,5]$. Prve i poslednje tačke oba niza se poklapaju sa krajevima intervala.

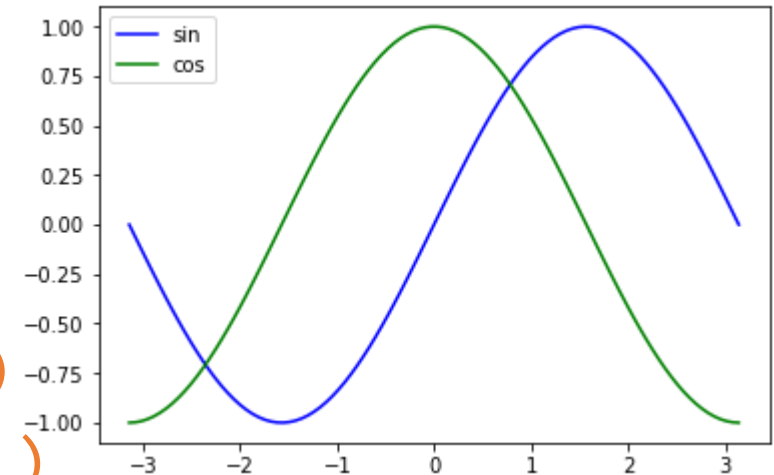
```
x = np.linspace(0, 10, 20)
y = np.linspace(0, 5, 20)
plt.plot(x, y, '.')
plt.show()
```



Mathplotlib

Napisati program kojim se iscrtavaju grafici funkcija $\sin x$ i $\cos x$ za $x \in [-\pi, \pi]$. Funkcija $\sin x$ treba da bude iscrtana plavom, a funkcija $\cos x$ zelenom bojom. Legenda treba da se nalazi u gornjem levom uglu.

```
x = np.linspace(-np.pi, np.pi, 100)
sin_x = np.sin(x)
cos_x = np.cos(x)
plt.plot(x, sin_x, label='sin', color='blue')
plt.plot(x, cos_x, label='cos', color='green')
plt.legend(loc='upper left')
plt.show()
```



Mathplotlib

Napisati program kojim se iscrtava grafik funkcije $\sin x$ za $x \in [-2\pi, 2\pi]$. Linija grafika treba da bude isprekidana i debljine 2.

Naslov grafika je Grafik sinusne funkcije, a na x -osi i y -osi treba redom staviti obeležja Domen funkcije i Vrednost funkcije.

Na x -osi treba da bude ucrtano 15 ekvidistantnih vrednosti u intervalu $[-7, 7]$, a na y -osi 9 ekvidistantnih vrednosti u intervalu $[-1, 1]$.

Veličina figure treba da bude 7×4 kvadratna inča, a sliku treba sačuvati pod imenom sinus.jpg. Rezolucija prikaza i sačuvane slike iznosi dpi = 80.

Mathplotlib

```
plt.figure(figsize = (7, 4), dpi = 80)
```

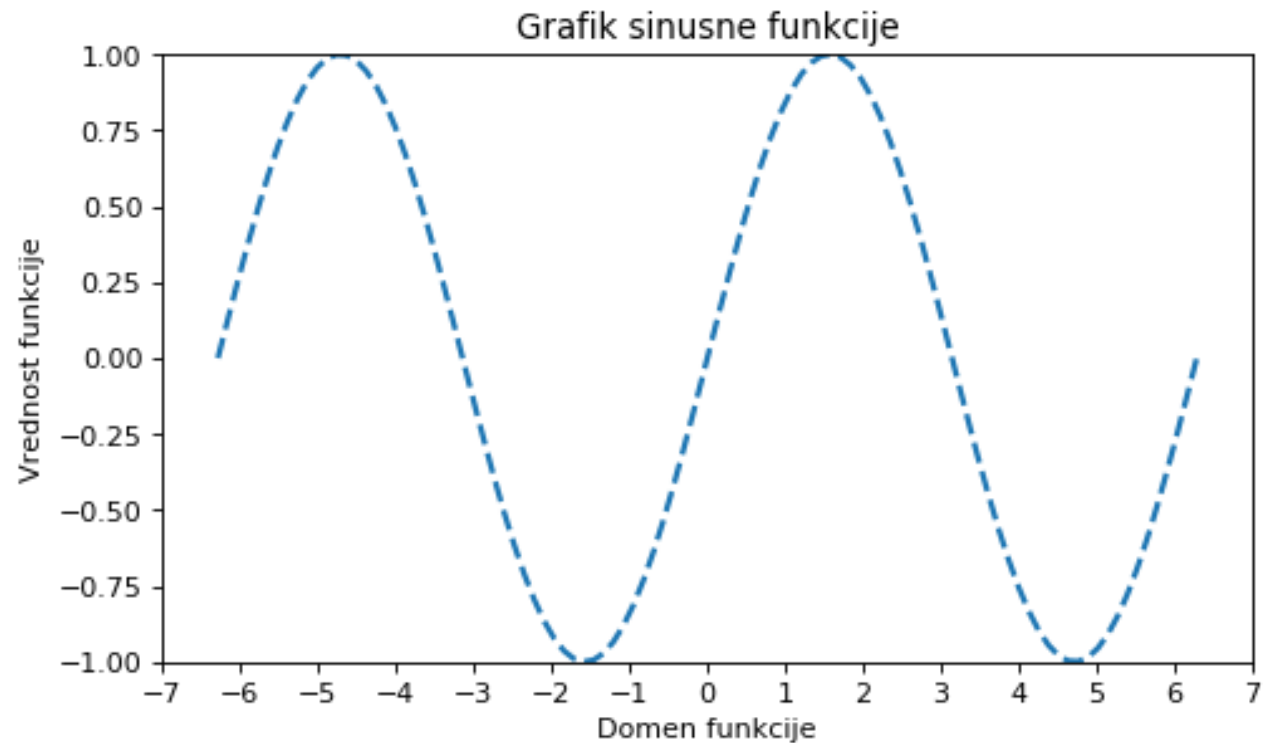
```
plt.xlabel("Domen funkcije")  
plt.ylabel("Vrednost funkcije")  
plt.title("Grafik sinusne funkcije")
```

```
# krajnje tacke intervala x-ose i y-ose  
plt.xlim(-7, 7)  
plt.ylim(-1, 1)
```

```
# vrednost podeoka na x-osi i y-osi  
plt.xticks(np.linspace(-7, 7, 15))  
plt.yticks(np.linspace(-1, 1, 9))
```

```
x = np.linspace(-2*np.pi, 2*np.pi, 100)  
sin_x = np.sin(x)  
plt.plot(x, sin_x, linewidth='2', linestyle='--')
```

```
plt.savefig("sinus.jpg", dpi = 80)  
plt.show()
```



Mathplotlib

Kako dodati ose?

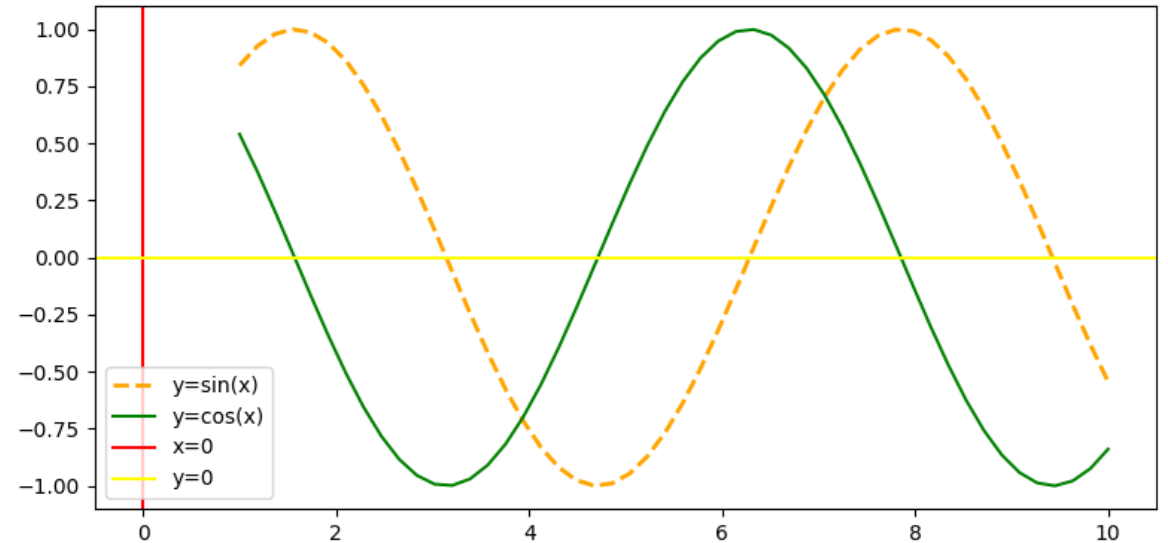
```
import numpy as np
from matplotlib import pyplot as plt
```

```
x = np.linspace(1, 3*np.pi, 50)
y1 = np.sin(x)
y2 = np.cos(x)
```

```
plt.plot(x, y1, c="orange", label="y=sin(x)", linewidth='2', linestyle='--')
plt.plot(x, y2, c="green", label="y=cos(x)")
```

```
plt.axvline(x=0, c="red", label="x=0")
plt.axhline(y=0, c="yellow", label="y=0")
```

```
plt.legend()
plt.show()
```



Mathplotlib

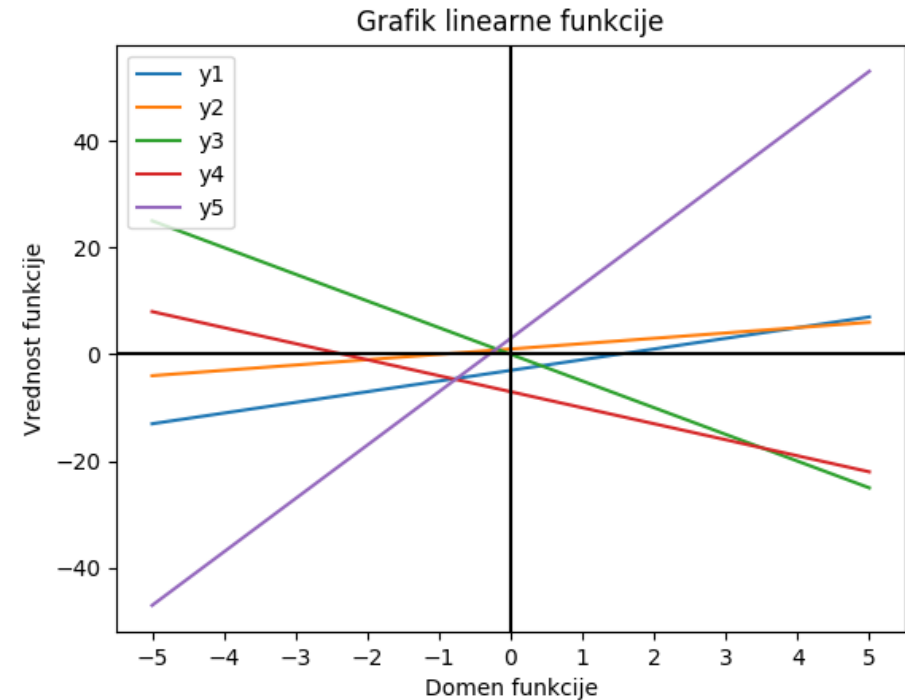
Prikazati različite linearne funkcije:

- $y = 2x - 3$;
- $y = x + 1$;
- $y = -5x$;
- $y = -3x - 7$;
- $y = 10x + 3$.

Opšti oblik linearne funkcije je $y = kx + n$, gde su k i n konstante koje mogu imati različite vrednosti. Ovaj zadatak ćemo podeliti u dva dela, prvi je generisanje vrednosti (x, y) tačaka za svaku od navedenih funkcija. U tu svrhu biće korišćena NumPy biblioteka. Nakon toga, korišćenjem Matplotlib biblioteke biće iscrtane sve funkcije u okviru istog grafikona, kako bi se lakše uočile međusobne razlike u zavisnosti od vrednosti k i n .

Mathplotlib

```
import numpy as np
import matplotlib.pyplot as plt
#prvo formiramo vrednosti za x:
x = np.linspace(-5, 5, 100)
k = np.array([2,1,-5,-3,10])
n = np.array([-3,1,0,-7,3])
#ima 5 razlicitih vrednosti za y, zbog razlicitih funkcija
y = np.empty((5,100));
for i in range(5):
    yi=k[i]*x+n[i]
    y[i]=np.array(yi);
#crtanje
plt.xlabel("Domen funkcije")
plt.ylabel("Vrednost funkcije")
plt.title("Grafik linearne funkcije")
# vrednost podeoka na x-osi
plt.xticks(np.linspace(-5, 5, 11))
for i in range(5):
    plt.plot(x, y[i], label='y'+str(i+1))
#plt.grid(True, which='both')
plt.axhline(y=0, color='black')
plt.axvline(x=0, color='black')
plt.legend(loc='upper left')
plt.show()
```



Mathplotlib

Na osnovu rezultata jedne ankete, utvrđeno je da se 45% ispitanika glasalo za, 20% protiv, a 35% je bilo uzdržano. Napisati program kojim se iscrtava odgovarajući kružni grafik. Svaku vrednost obojiti različitom bojom, i dodati odgovarajuća obeležja.

```
plt.title('Rezultati ankete')
labels = ['Za', 'Protiv', 'Uzdržani']
sizes = [45, 20, 35]
colors = ['red', 'yellow', 'blue']
plt.pie(sizes, colors=colors, labels=labels)
plt.show()
```



Mathplotlib

Napisati program koji grafički predstavlja rezultate pismenog zadatka po ocenama. Kako je poznat ukupan broj učenika koji je radio pismeni zadatak i poznat je broj učenika koji su dobili svaku od ocena, jednostavno je odrediti procenat zastupljenosti neke ocene u odnosu na ukupan broj ocena.

```
import numpy as np
from matplotlib import pyplot as plt

plt.title('Rezultati pismenog zadatka')
labels = ['5', '4', '3', '2', '1']
broj_ucenika = 35
broj_ucenika_po_ocenama = np.array([5,8,17,3,2])

procenat_po_ocenama = np.empty((5,))

for i in range(5):
    procenat_po_ocenama[i] = 100/broj_ucenika*broj_ucenika_po_ocenama[i]

colors = ['red', 'yellow', 'blue', 'green', 'pink']
plt.pie(procenat_po_ocenama, colors=colors, labels=labels)
plt.show()
```



Mathplotlib

```
import numpy as np
from matplotlib import pyplot as plt
```

```
plt.title('Rezultati pismenog zadatka')
labels = ['5', '4', '3', '2', '1']
izmestanje = [0.1, 0, 0, 0, 0]
broj_ucenika = 35
broj_ucenika_po_ocenama = np.array([5,8,17,3,2])
```

```
procenat_po_ocenama = np.empty((5,))
```

```
for i in range(5):
    procenat_po_ocenama[i] = 100/broj_ucenika*broj_ucenika_po_ocenama[i]
```

```
colors = ['red', 'yellow', 'blue', 'green', 'pink']
plt.pie(procenat_po_ocenama, colors=colors, labels=labels, explode=izmestanje)
plt.show()
```

