

Domaći zadatak broj 3

Ime i prezime studenta: Nastasija Stankovic Broj indeksa: 17955

Uputstvo: 1. Pre početka izrade promenite ime datoteke u Domaci3_ime_prezime. (ubacite svoje ime i prezime) 2. Popunite ćeliju ispod naslova odgovarajućim podacima. 3. Za rešavanje zadataka, ukoliko je potrebno, otvorite ispod teksta zadataka dodatne ćelije za upisivanje tekstualnog odgovora (Markdown) ili programskog koda (Code). 4. Nakon završetka izrade rešenja Notebook dokument sačuvati u pdf formatu i proslediti ga nastavniku. U možete da uradite ili kroz Teams ili na mail adresu jovana.dzunic@elfak.ni.ac.rs

Zadatak 1. Kog tipa podataka je skup vrednosti indeksa objekat for petlje u sledećem kodu, ukoliko ga for naredba prihvata?

a) (2 poena)

```
pythone
objekat = ["reć", "frazza", 8, [1,2,3]]
for indeks in objekat:
    print(indeks)
```

- [] string
- [X] lista
- [] lista
- [] niz
- [] broj
- [] ova naredba proizvodi poruku greške

b) (2 poena)

```
pythone
objekat = "zadatak i pitanje"
for indeks in objekat:
    print(indeks)
```

- [X] string
- [] lista
- [] niz
- [] broj
- [] ova naredba proizvodi poruku greške

c) (2 poena)

```
pythone
objekat = 18
for indeks in objekat:
    print(indeks)
```

- [] string
- [] lista
- [] niz
- [] broj
- [X] ova naredba proizvodi poruku greške

d) (2 poena)

```
pythone
objekat = numpy.arange(2,8,1.5)
for indeks in objekat:
    print(indeks)
```

- [] string
- [] lista
- [X] niz
- [] broj
- [] ova naredba proizvodi poruku greške

Zadatak 2. Kog tipa podataka je indeks u sledećem kodu?

a) (2 poena)

```
pythone
objekat = ["1", "2.3"]
for indeks in objekat:
    print(indeks)
```

- [X] string
- [] lista
- [] niz
- [] broj

b) (2 poena)

```
pythone
objekat = [[], [1,2],[ 'a', 'v',1]]
for indeks in objekat:
    print(indeks)
```

- [] string
- [X] lista
- [] niz
- [] broj

c) (2 poena)

```
pythone
objekat = numpy.arange(2,8,1.5)
for indeks in objekat:
    print(indeks)
```

- [] string
- [] lista
- [] niz
- [X] broj

Zadatak 3. a) Ispitati tačnost tvrdjenja: unija dva konveksna skupa je konveksan skup. Ako je odgovor TAČNO, dati dokaz. Ukoliko je odgovor NIJE TAČNO, dati primer (može slika) kada ovo tvrdjenje ne važi.

(3poena)

Ovo tvrdjene u nekim slucajevima vazia u nekim ne.Prvi primer dokazuje kada tvrdjene vazia, drugi primer kada ne vazi:

```
In [5]: import matplotlib as mpl
import matplotlib.pyplot as plt
s, =plt.subplots(1,2)
plt.subplot(1,2,1)
plt.title("Tvrdjenje važi:")
plt.fill([1,2,3],[1,1,2])
plt.fill([2,3,4],[1,2,2])
plt.plot([1,4],[1,2], '--g')

plt.subplot(1,2,2)
plt.title("Tvrdjenje ne važi")
plt.fill([1,2,3],[1,1,2])
plt.fill([4,5,6],[1,2,2])
plt.plot([1,6],[1,2], '--g')
```

Out [5]: <matplotlib.lines.Line2D at 0x1d6da4b7eb0>



b) Koristeći nejednakost trougla pokazati da je centralna jedinična sfera $S \subset \mathbb{R}^n$ konveksan skup.

$$S = \{v \in \mathbb{R}^n \mid \|v\| \leq 1\}.$$

(3poena)

Za dva vektora $u, v \in \mathbb{S}$ i znamo da za konveksnost vazia $\lambda v + (1 - \lambda)u \in \mathbb{S}$.Uslov za pripadnost skupa S je $S = \{v \in \mathbb{R}^n \mid \|v\| \leq 1\}$ treba dokazati da je

$$\|\lambda v + (1 - \lambda)u\| \leq 1$$

Koristeći nejednakost trougla

$$\|\lambda v + (1 - \lambda)u\| \leq \|\lambda v\| + \|(1 - \lambda)u\|$$

Znamo da vazia $\|\lambda v\| = \lambda \|v\|$ i $\|(1 - \lambda)u\| = (1 - \lambda)\|u\|$, zamenimo ove izraze u nejednakosti trougla

Dobijamo:

$$\|\lambda v + (1 - \lambda)u\| \leq \lambda \|v\| + (1 - \lambda)\|u\|$$

Iz uslova pripadnosti znamo da je $\|v\| \leq 1$ sto znaci:

$$\|\lambda v + (1 - \lambda)u\| \leq \lambda + (1 - \lambda) = 1,$$

$\lambda \in [0, 1]$ time smo dokazali da je centralna jedinicna sfera koveksna.

c) Koristeći nejednakost trougla pokazati da je sfera $S \subset \mathbb{R}^n$ sa centrom $c \in \mathbb{R}^n$ i poluprečnika r konveksan skup.

$$S = \{v \in \mathbb{R}^n \mid \|v - c\| \leq r\}.$$

(5poena)

Razlika u odnosu na dokaz u prethodnom primeru je u tome sto je sfera izmeštena, tako da moramo da oduzmemo položaj centra sfere od vektora tačaka. Dokažimo da za $\lambda \in [0, 1]$ važi:

$$\|\lambda(v - c) + (1 - \lambda)(u - c)\| \leq r$$

Na osnovu nejednakosti trougla dobijamo:

$$\|\lambda(v - c) + (1 - \lambda)(u - c)\| \leq \|\lambda(v - c)\| + \|(1 - \lambda)(u - c)\| \leq \lambda \|v - c\| + (1 - \lambda)\|u - c\|$$

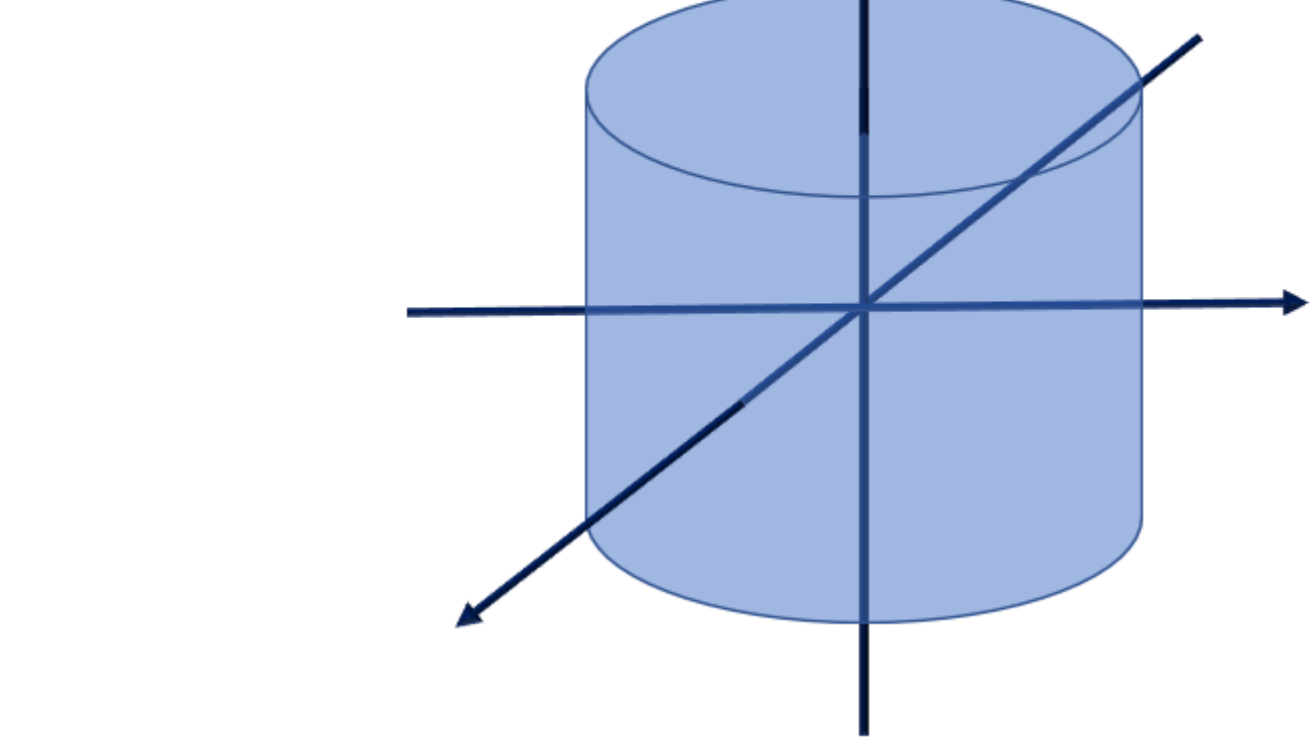
Ukoliko izvršimo izmenu $\|(v - c)\|$ i $\|(u - c)\|$ sa r:

$$\|\lambda(v - c) + (1 - \lambda)(u - c)\| \leq \lambda r + (1 - \lambda)r = r$$

čime je dokazan uslov konveksnosti.

Zadatak 4. Programskim kodom napraviti grafički prikaz cilindrične površi zadate parametarskim jednačinama

$$\begin{cases} x = x(\theta) = r \cos(\theta), \\ y = y(\theta) = r \sin(\theta), \quad \theta \in [0, 2\pi], r = 1.5. \\ z \in [-2, 2], \end{cases}$$

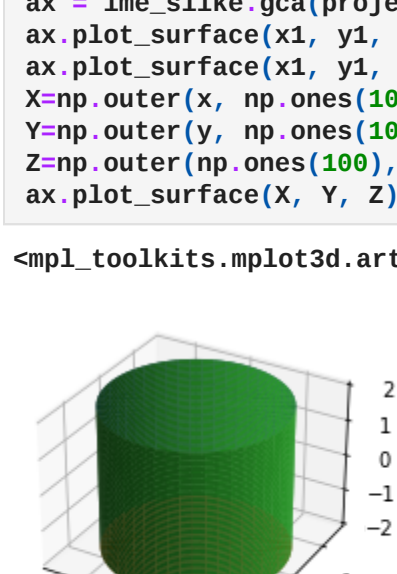


Učitati potrebne module i dati im skraćena imena. Te skraćence koristiti i u narednom zadatku.

(15poena)

```
In [7]: %matplotlib inline
import matplotlib as mpl
from mpl_toolkits.mplot3d import Axes3D
import numpy as np
import matplotlib.pyplot as plt
r=1.5
teta=np.linspace(0, 2*np.pi, 100)
z=np.linspace(-2, 2, 100)
x=r*np.cos(teta)
y=r*np.sin(teta)
x1=np.outer(np.linspace(0, 1, 100), x)
y1=np.outer(np.linspace(0, 1, 100), y)
z1=z*(1+x1*y1)/(1+x1*y1)
z2=-2*(1+x1*y1)/(1+x1*y1)
ime_slike = plt.figure(figsize=plt.figaspect(1)*0.8)
ax = ime_slike.gca(projection='3d')
ax.plot_surface(x1, y1, z1)
ax.plot_surface(x1, y1, z2)
X=np.outer(x, np.ones(100))
Y=np.outer(y, np.ones(100))
Z=np.outer(np.ones(100), z)
ax.plot_surface(X, Y, Z)
```

Out [7]: <mpl_toolkits.mplot3d.art3d.Poly3DCollection at 0x1d6da7f6910>



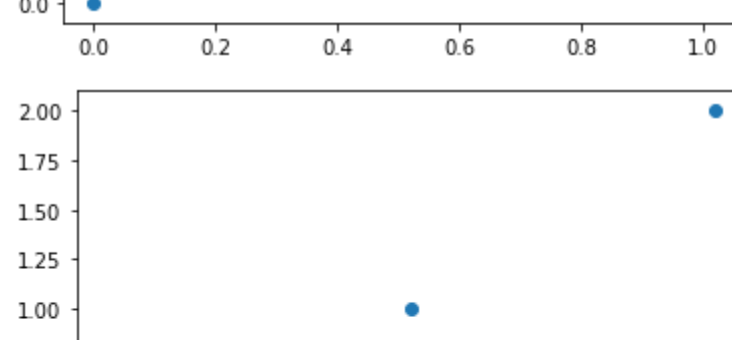
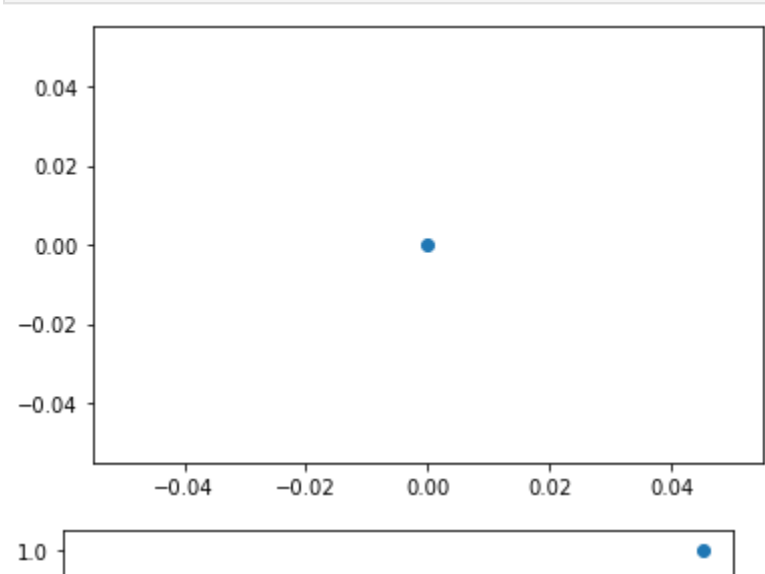
Zadatak 5. Za kreiranje niza grafičkih prikaza koji se menjaju sa svakom iteracijom može se koristiti naredba pause.

matplotlib.pyplot.pause(t)

Njom se postize i odlaganje izvršenja naredne instrukcije u vremenskom periodu t (meri se u sekundama) koji se navede između zagrada. Videti jednostavan primer u nastavku.

```
In [14]: import matplotlib.pyplot as plt

for i in range(3):
    plt.plot(range(i+1),range(i+1),'o')
    plt.pause(1)
```



Primenom naredbe pause kreirati iterativni prikaz izgradnje konveksnog zatvorenja na sledeći način:

- Kreirati niz points od 7 random tačaka u 3D sa opsegom indeksa (-1, 1).
 - Za indeks iteracije i = 4 : 7 raditi sledeće:
 - Za tačke points[i] odrediti konveksno zatvorenje.
 - Napraviti 3D grafički prikaz dobijenog poliedra sa opcijama za strane edgecolors='k', facecolors='w', linewidths=1, alpha=0.5.
 - Staviti naslov niza "konveksno zatvorenje i tačaka" tako da i uzme vrednost indeksa iteracije.
 - Pozvati naredbu pause u željenom trajanju.

Za realizaciju koda učitati potrebne dodatne module ili pojedine naredbe iz njih. (10poena)

```
In [9]: points=np.random.rand(7,3)
from mpl_toolkits.mplot3d.art3d import Poly3DCollection
from scipy.spatial import ConvexHull
%matplotlib notebook
for i in range(4,7):
    t=plt.figure(figsize=plt.figaspect(1))
    ax=t.gca(projection='3d')
    teme1=[]
    for j in range(0, i-1):
        teme1.append(points[j,:])
    teme2=np.array(teme1)
    Z=ConvexHull(teme2)
    plt.plot(teme2[:,0], teme2[:,1], teme2[:,2], "ko")
    for s in Z.simplices:
        ax.add_collection3d(Poly3DCollection(points[s], edgecolors='k', facecolors='w', linewidths=1, alpha=0.5))
    plt.pause(1)
```

