



Predmet: Signali i sistemi

Uputstvo za drugi domaći zadatak 2020/2021. Furijeova transformacija

1. Bibliotke koje treba instalirati:

- numpy
- matplotlib
- scipy

2. Potrebne funkcije

- # čitanje odbiraka iz .wav fajla

```
from scipy.io import wavfile  
fs, data = wavfile.read(file_name)
```

file_name – ime (putanja) .wav fajla (npr. 'audio_1.wav')
fs – učestanost odabiranja datog .wav fajla
data – odbirci audio signala

- # generisanje frekvencijske ose

```
from scipy.fftpack import fftshift, fftfreq  
freq_axis = fftshift(fftfreq(len(data), 1/fs))
```

fftfreq generiše učestanosti, a fftshift pomera promenljivu tako da komponenta na učestanosti 0 bude na sredini spektra

data – odbirci audio signala
fs – učestanost odabiranja datog .wav fajla
freq_axis – frekvencijska osa

- # projektovanje filtra i filtriranje signala

```
import scipy.signal as sci

[b, a] = sci.butter(n, f_gr/(fs/2), btype=filt_type) # projektovanje
filtra
```

n – red filtra (npr. n = 6)

f_gr – granice filtra u Hz (npr. za NF filter frekvencija do 2kHz: f_gr = 2000,

za PO filter opsega frekvencija 2kHz – 6 kHz: fgr = np.array((2000, 6000)))

filt_type – tip filtra (za NF filter filt_type = 'lowpass', za PO filter filt_type = 'bandpass')

b, a – parametri filtra

```
x_f = sci.filtfilt(b, a, x) # filtriranje signala
```

b, a – parametri filtra dobijeni prethodnom funkcijom

x – vremenski oblik signala (odbirci audio signala)

x_f – vremenski oblik filtriranog signala (odbirci filtriranog audio signala)

- # zapisivanje odbiraka u .wav fajl

```
(from scipy.io import wavfile) # ako nije ranije uvezeno
import numpy as np
```

```
x_scaled = np.int16(x/np.max(np.abs(x)) * 32767) # skaliranje odbiraka
na INT opseg
```

```
wavfile.write(file_name, fs, x_scaled) # čuvanje .wav fajla
```

x – odbirci audio signala

file_name – ime (putanja) fajla koji želimo da sačuvamo (npr. 'x1_filtrirano.wav')

fs – učestanost odabiranja signala koji čuvamo (ista kao fs datog signala)

x_scaled – skalirani celobrojni odbirci audio signala

- # Furijeova transformacija signala

```
from scipy.fftpack import fft
```

```
X_fft = fft(x)
Xa = np.abs(X_fft)
```

x – odbirci audio signala

X_fft – Furijeova transformacija signala

Xa – amplitudski spektar