B. Deepak Kumar

BL.EN.U4AIE21028

AIE-D

LAB - 6

In [10]:

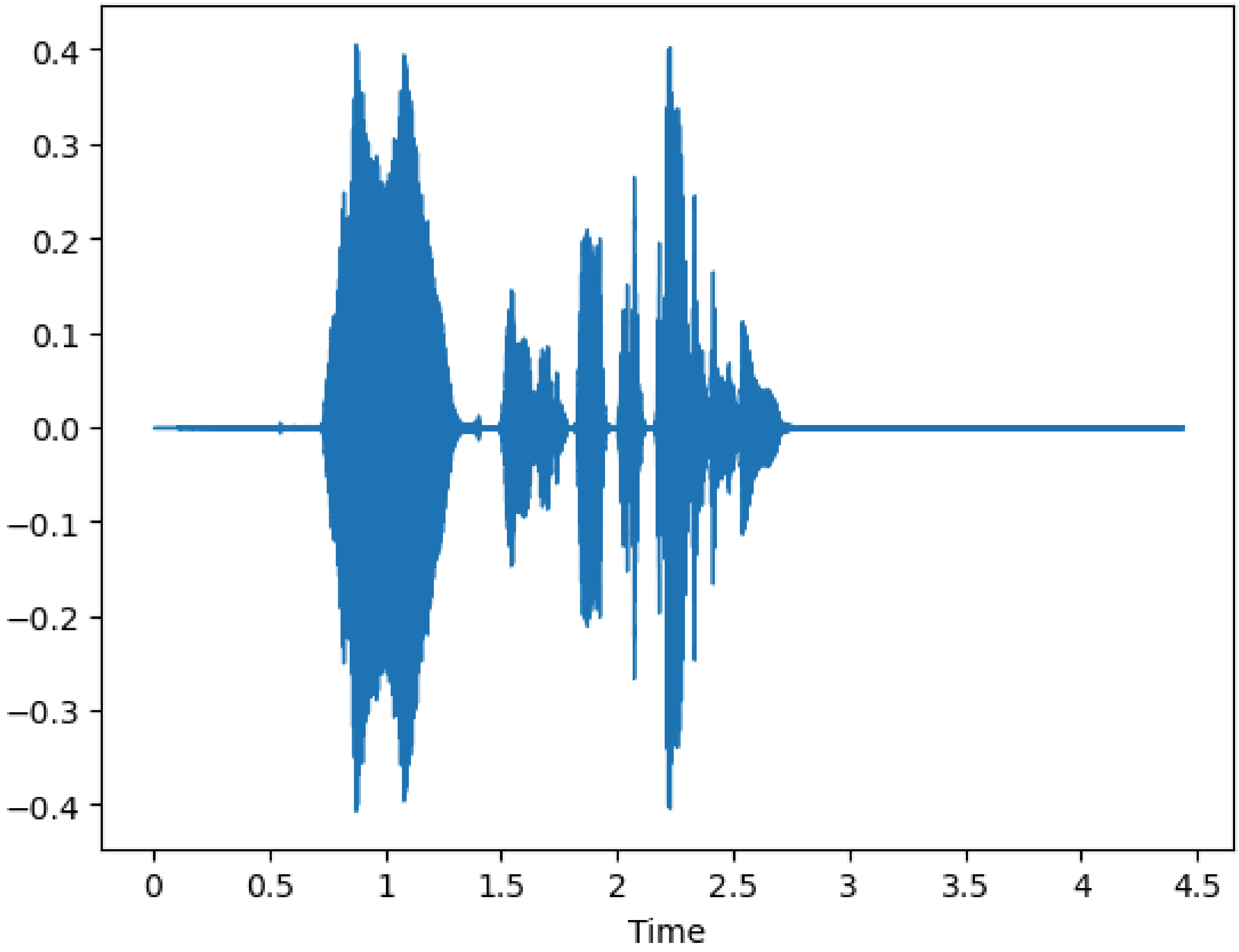
```python
import numpy as np
import librosa
import matplotlib.pyplot as plt
import IPython.display as ipd
import scipy.signal as signal
import scipy.io.wavfile as wavfile
from glob import glob
import seaborn as sns
from scipy.signal import spectrogram
```

In [11]:

```
y, sr = librosa.load('AISPS.wav')
librosa.display.waveshow(y)
```

Out[11]:

```
<librosa.display.AdaptiveWaveplot at 0x14e9234e910>
```



In [13]:

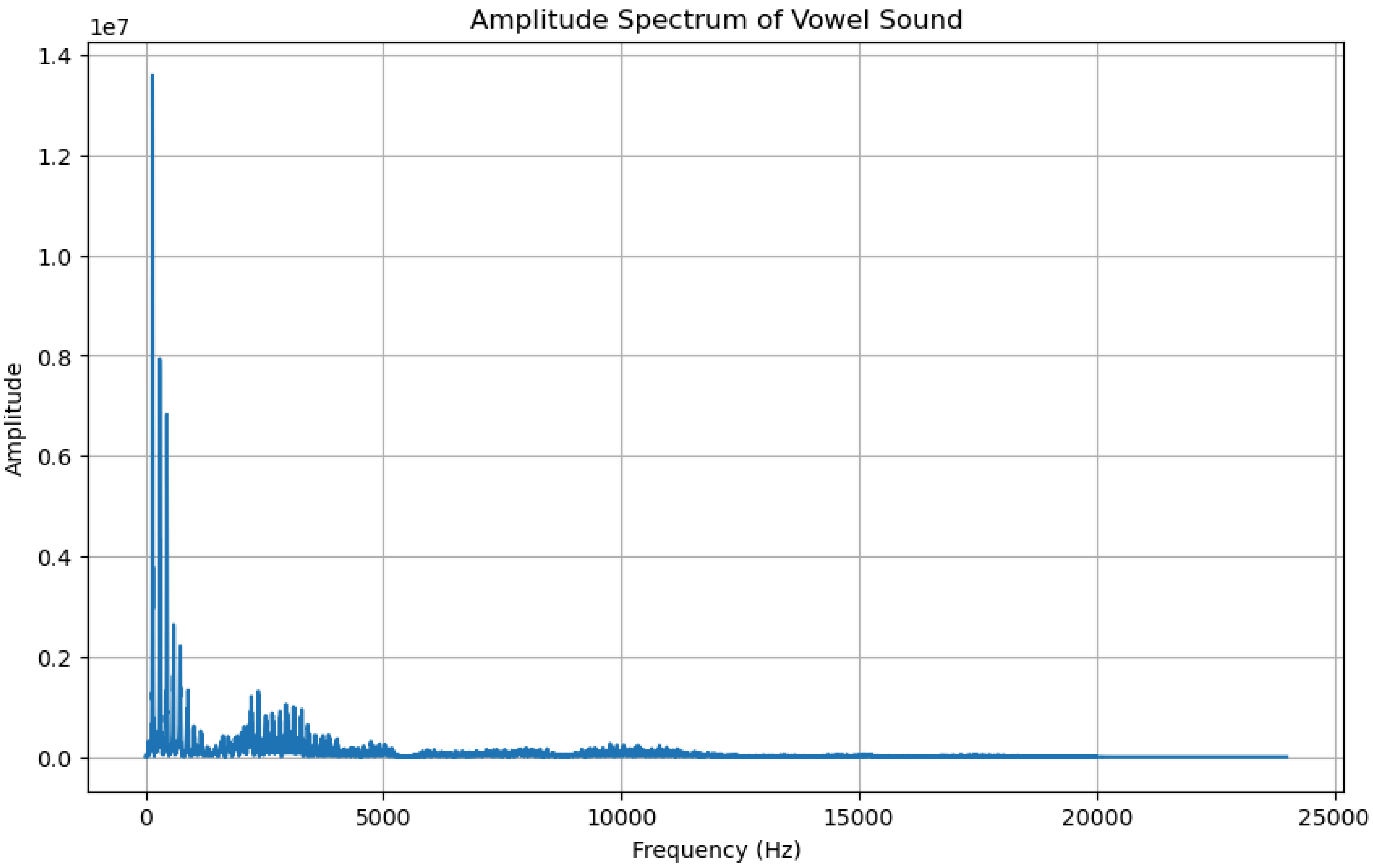
```
a = glob('AISPS.wav')
ipd.Audio(a[0])
```

Out[13]:

A1. Take a portion of your recorded signal which represents a vowel sound. Perform FFT on the signal snippet and observe the amplitude spectrum. Repeat the same for a few vowel sounds.

In [12]:

```

sample_rate, data = wavfile.read('AISPS.wav')

start_time = int(0.5 * sample_rate)
end_time = int(1.0 * sample_rate)

vowel_sound = data[start_time:end_time]
fft_result = np.fft.fft(vowel_sound)
frequency_bins = np.fft.fftfreq(len(fft_result), 1/sample_rate)
amplitude_spectrum = np.abs(fft_result)

plt.figure(figsize=(10, 6))
plt.plot(frequency_bins[:len(frequency_bins)//2], amplitude_spectrum[:len(freque
plt.title('Amplitude Spectrum of Vowel Sound')
plt.xlabel('Frequency (Hz)')
plt.ylabel('Amplitude')
plt.grid(True)
plt.show()

```

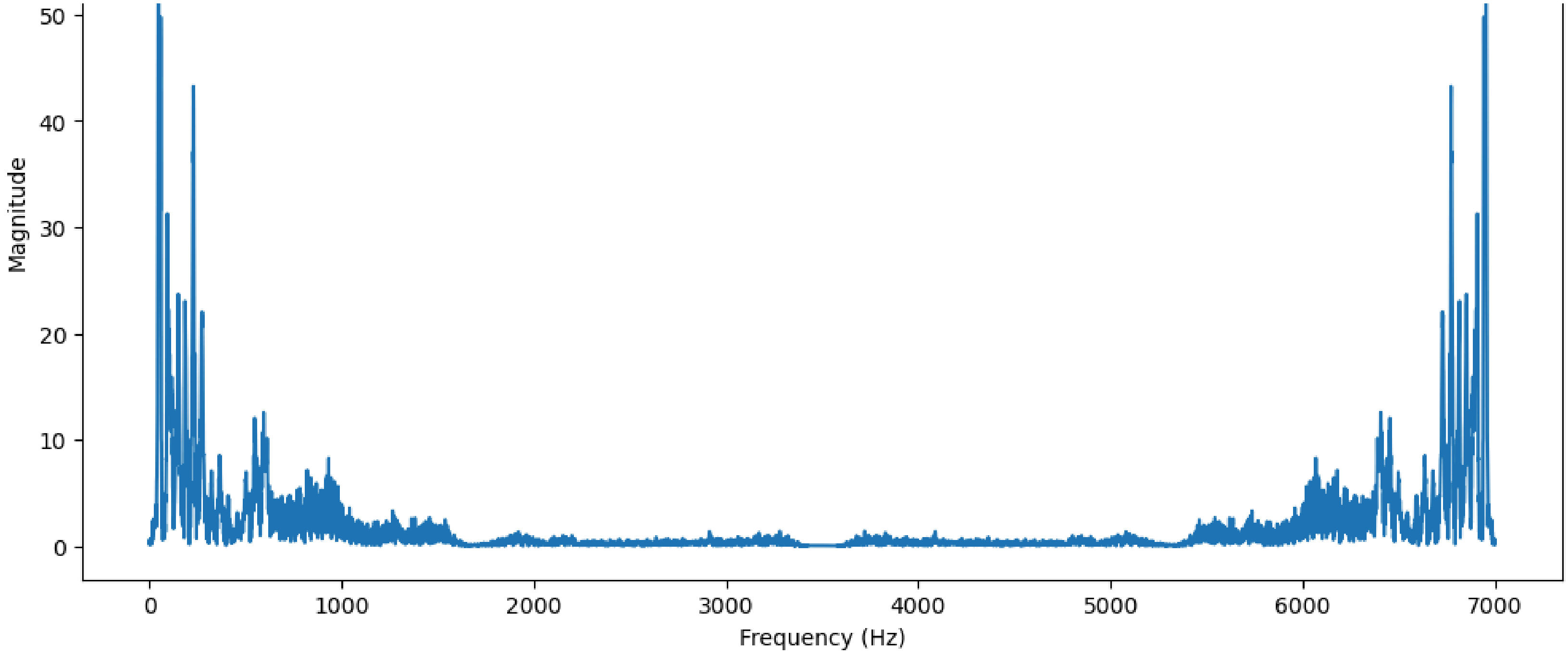

Amplitude Spectrum of Vowel Sound
1e7
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Amplitude
0
5000
10000
15000
20000
25000
Frequency (Hz)

In [15]:

```python
def plot_fft_and_play(signal, title):
    # Plot FFT
    fft_result = np.fft.fft(signal)
    plt.figure(figsize=(12, 6))
    plt.plot(np.abs(fft_result))
    plt.title(title)
    plt.xlabel('Frequency (Hz)')
    plt.ylabel('Magnitude')
    plt.show()

    # Play audio
    ipd.display(ipd.Audio(signal, rate=sr))
```

In [17]:

```
# Vowel sounds
vowel_a = y[10000:16000]
plot_fft_and_play(vowel_a, "FFT of the Vowel 'A'")

vowel_i = y[12700:19000]
plot_fft_and_play(vowel_i, "FFT of the Vowel 'I'")

vowel_e = y[19000:26000]
plot_fft_and_play(vowel_e, "FFT of the Vowel 'E'")

vowel_o = y[24000:31000]
plot_fft_and_play(vowel_o, "FFT of the Vowel 'O'")
```

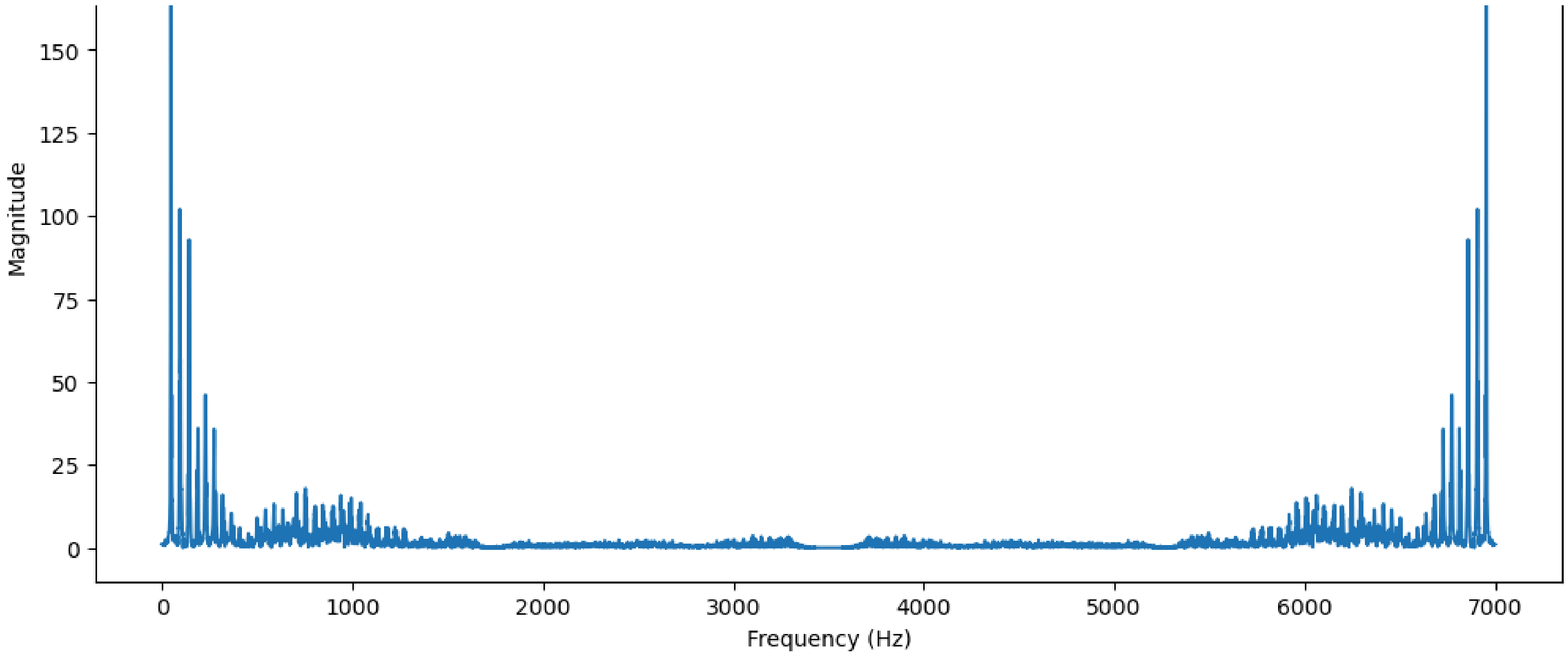

Magnitude
50
40
30
20
10
0
0
1000
2000
3000
4000
5000
6000
7000
Frequency (Hz)

0:00 / 0:00

A2. Repeat the A1 for a consonant sound. Perform the same for a few consonant sounds.

In [18]:

```python

# Consonant sounds
consonant_p = y[18000:26000]
plot_fft_and_play(consonant_p, "FFT of the Consonant 'P'")

consonant_s = y[16000:22000]
plot_fft_and_play(consonant_s, "FFT of the Consonant 'S'")

consonant_h = y[24000:31000]
plot_fft_and_play(consonant_h, "FFT of the Consonant 'H'")

consonant_c = y[18000:25000]
plot_fft_and_play(consonant_c, "FFT of the Consonant 'C'")

consonant_n = y[16000:23000]
plot_fft_and_play(consonant_n, "FFT of the Consonant 'N'")
```

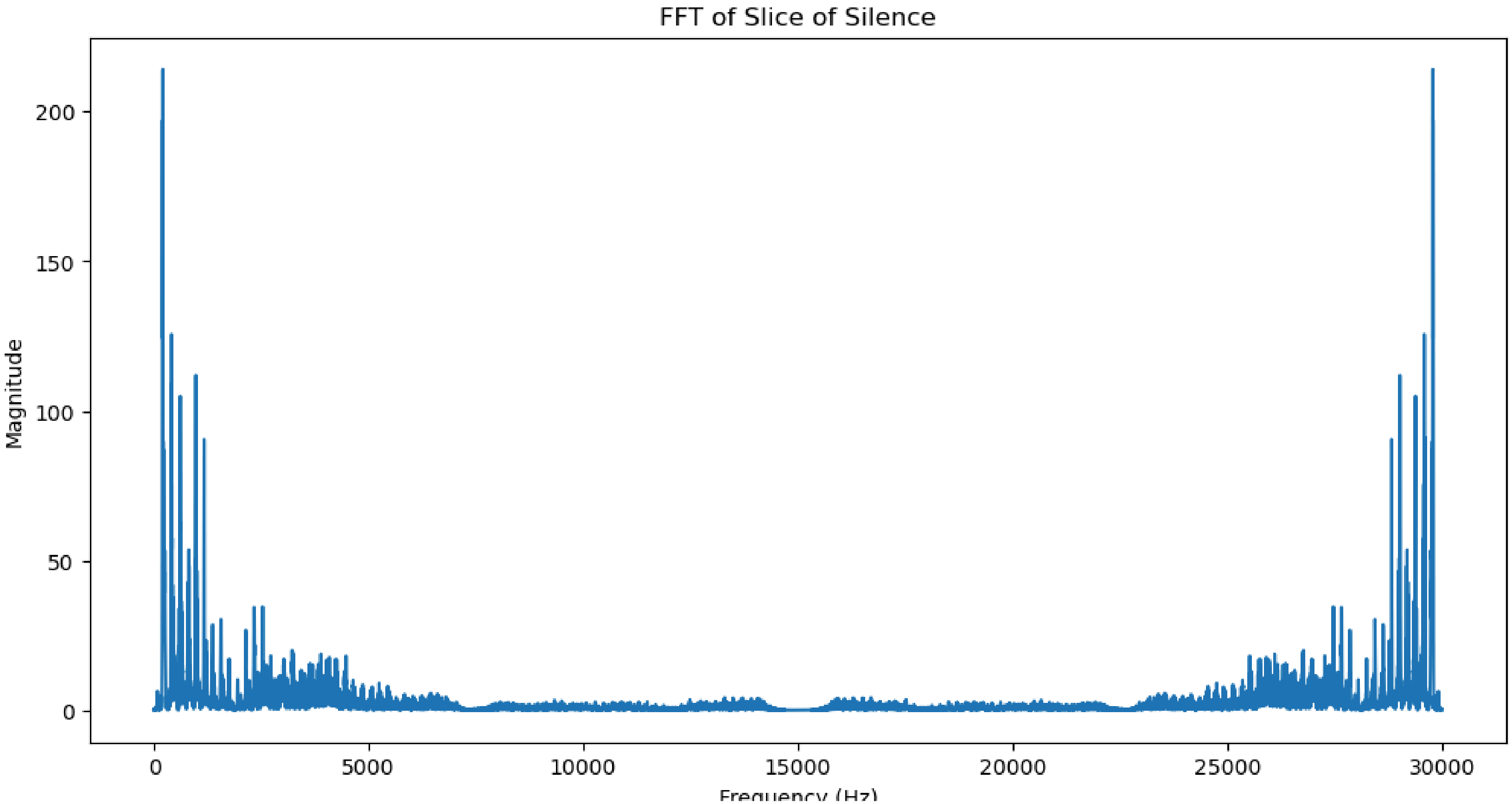

150
125
100
75
50
25
0
Magnitude
0
1000
2000
3000
4000
5000
6000
7000
Frequency (Hz)

0:00 / 0:00

A3. Repeat A2 for few slices of silence & non-voiced portions of the recorded speech signal.

In [19]:

```
# Silence and non-voiced portions
silence = y[0:30000]
plot_fft_and_play(silence, "FFT of Slice of Silence")

non_voiced = y[34000:]
plot_fft_and_play(non_voiced, "FFT of Non-voiced Portion")

non_voiced2 = y[30000:34000]
plot_fft_and_play(non_voiced2, "FFT of Non-voiced Portion")

```

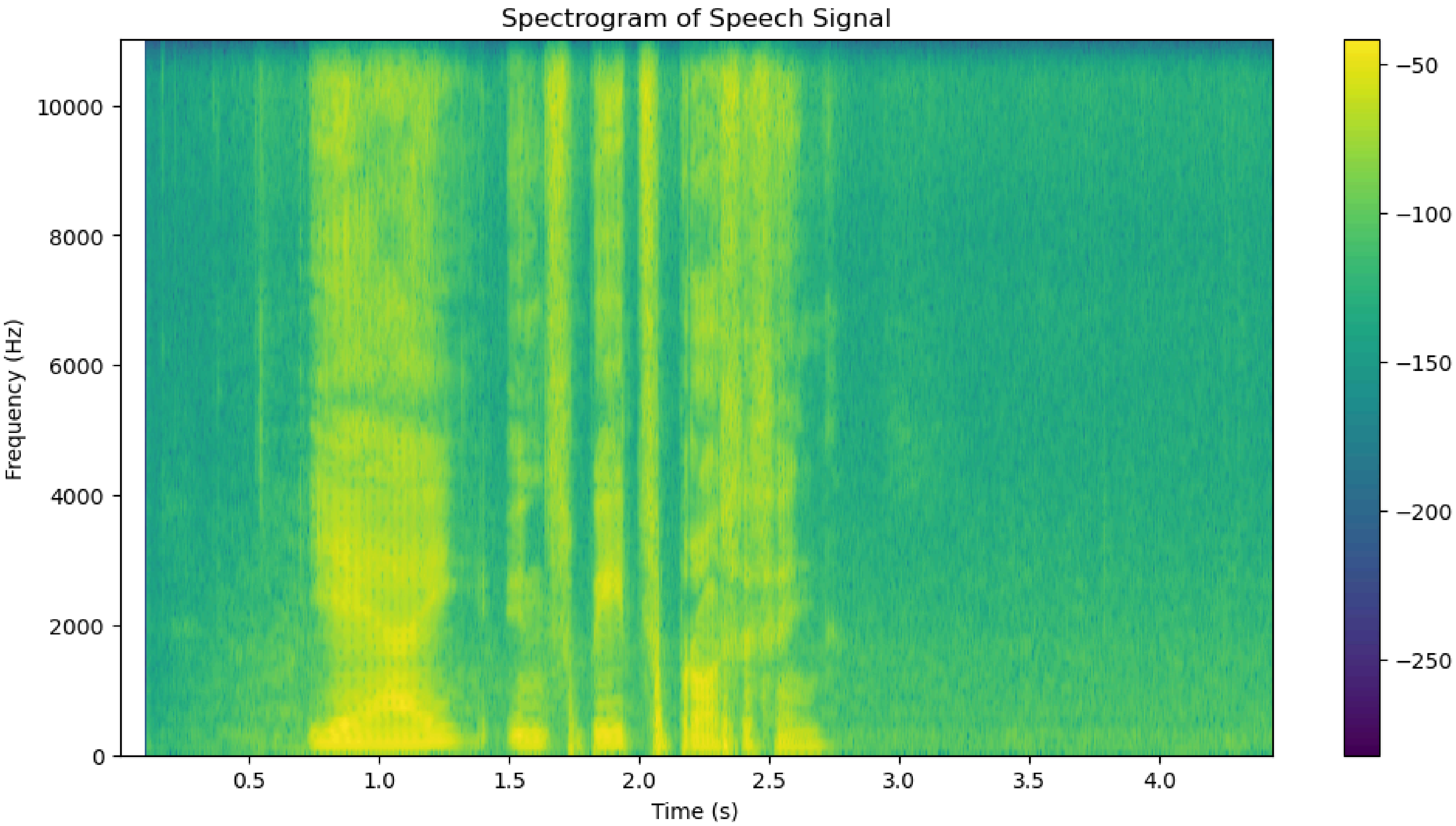

FFT of Slice of Silence
Magnitude
200
150
100
50
0
0
5000
10000
15000
20000
25000
30000
Frequency (Hz)

A4. Now you have acquainted yourself with spectral amplitudes of various consonants and vowelbased phonemes. Generate the spectrogram of the signal and observe the change points of the signals with associated speech segments. Observe to identify the consonants and vowels from the spectrogram.

In [20]:

```python
plt.figure(figsize=(12, 6))
plt.specgram(y, Fs=sr)
plt.title("Spectrogram of Speech Signal")
plt.xlabel("Time (s)")
plt.ylabel("Frequency (Hz)")
plt.colorbar()
plt.show()

# Play entire audio
ipd.display(ipd.Audio(y, rate=sr))
```

```
C:\Users\saide\anaconda3\lib\site-packages\matplotlib\axes\_axes.py:7622: RuntimeWar
ning: divide by zero encountered in log10
  Z = 10. * np.log10(spec)
```


Spectrogram of Speech Signal
Frequency (Hz)
0
2000
4000
6000
8000
10000
Time (s)
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
−50
−100
−150
−200
−250

0:04 / 0:04

In [ ]: