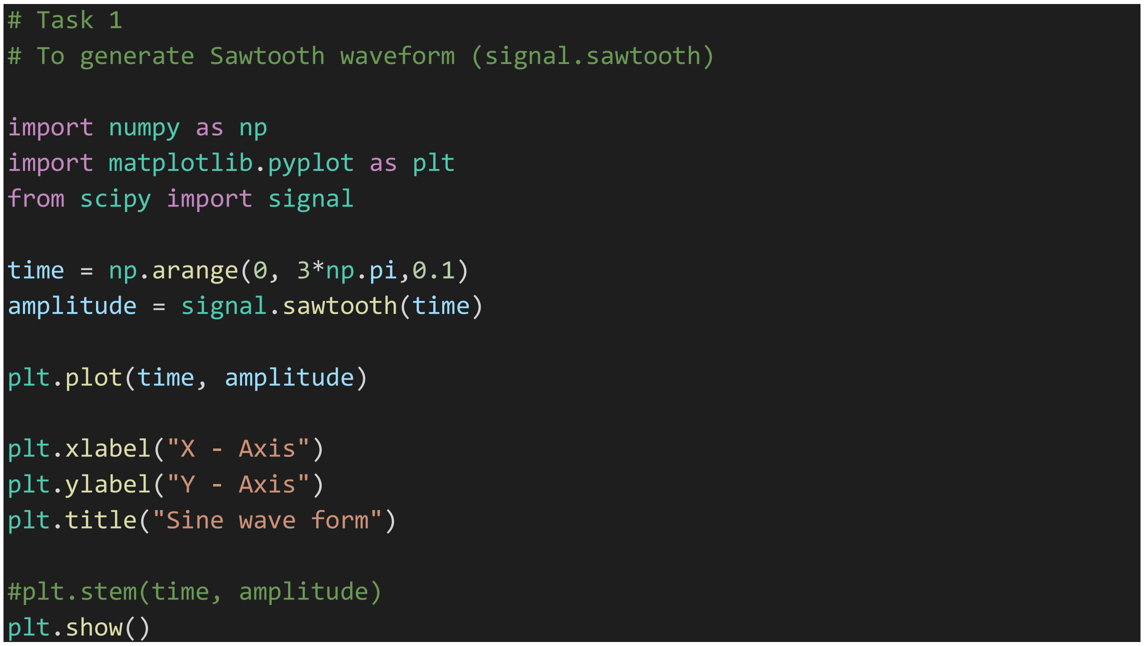
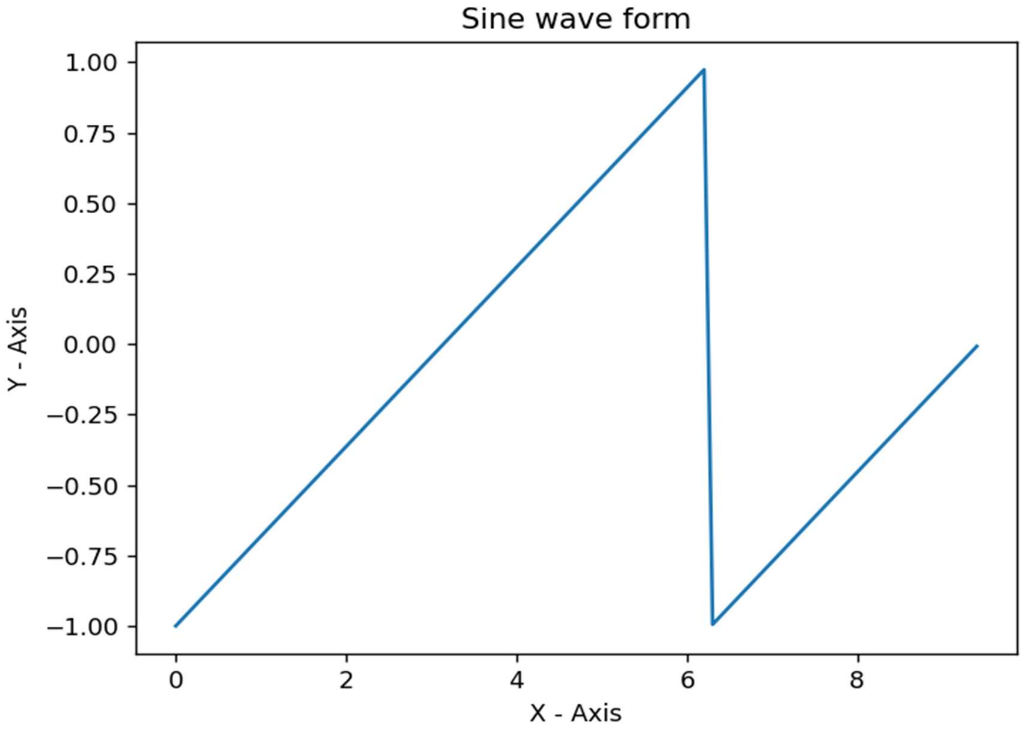
Lab 16

Name : Shashank Bagda **Date : 28 / 09 / 22**

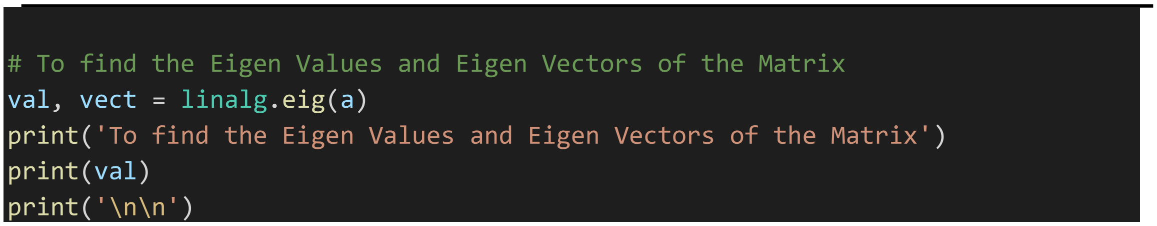
Enrollment No : 92100133020



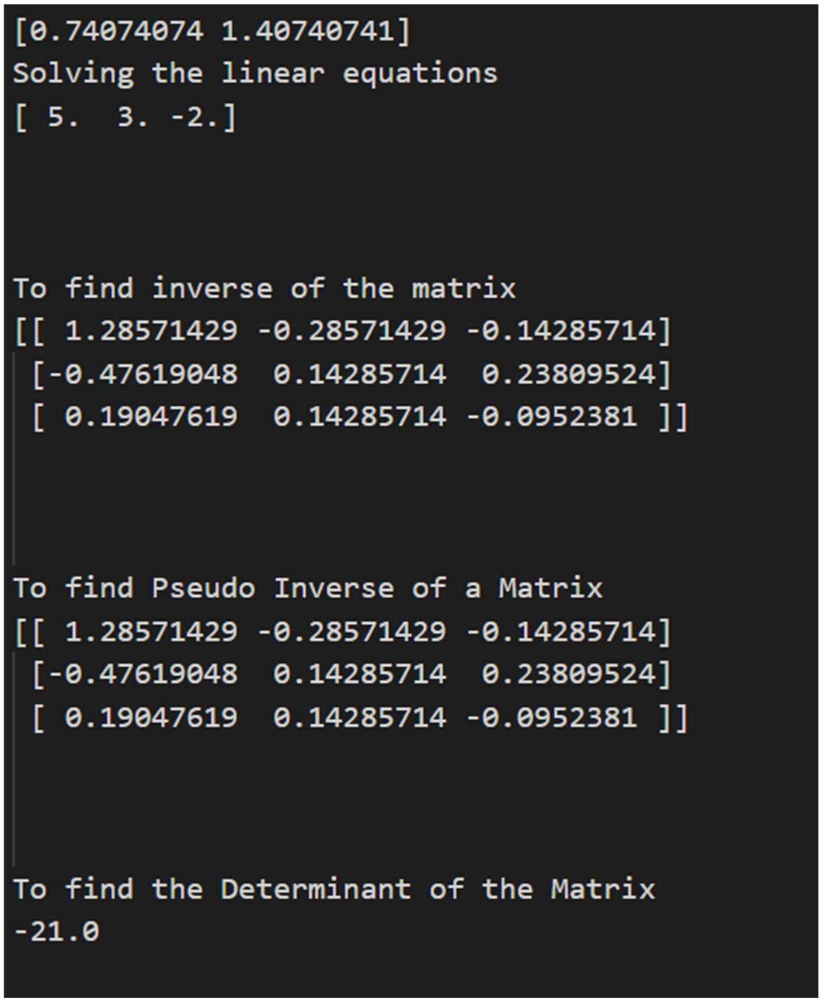
```
# Task 1
# To generate Sawtooth waveform (signal.sawtooth)

import numpy as np
import matplotlib.pyplot as plt
from scipy import signal

time = np.arange(0, 3*np.pi,0.1)
amplitude = signal.sawtooth(time)

plt.plot(time, amplitude)

plt.xlabel("X - Axis")
plt.ylabel("Y - Axis")
plt.title("Sine wave form")

#plt.stem(time, amplitude)
plt.show()
```



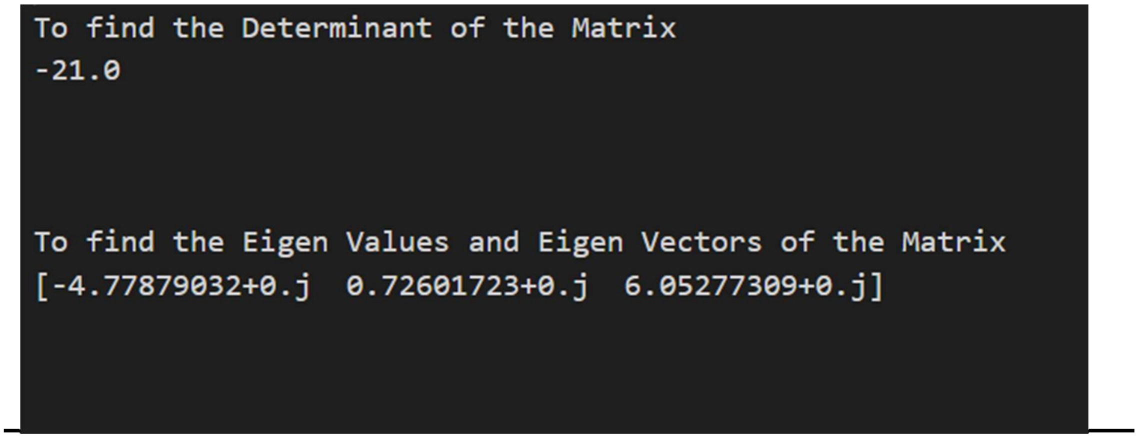
```
# Task 2 : Linear Algebra

# Example 1 : Solving the linear equations

from scipy import linalg
import numpy as np

a = np.array([[7, 2], [4, 5]])
b = np.array([8, 10])

res = linalg.solve(a,b)
print(res)


# Example 2 : Solving the linear equations

from scipy import linalg
import numpy as np

a = np.array([[1, 1, 1], [0, 2, 5], [2, 5, -1]])
b = np.array([6, -4, 27])

res = linalg.solve(a,b)
print('Solving the linear equations')
print(res)
print('\n\n')

# To find inverse of the matrix
inv = linalg.inv(a)
print('To find inverse of the matrix')
print(inv)
print('\n\n')

# To find Pseudo Inverse of a Matrix
psu = linalg.pinv(a)
print('To find Pseudo Inverse of a Matrix')
print(psu)
print('\n\n')

# To find the Determinant of the Matrix
det = linalg.det(a)
print('To find the Determinant of the Matrix')
print(det)
print('\n\n')
```



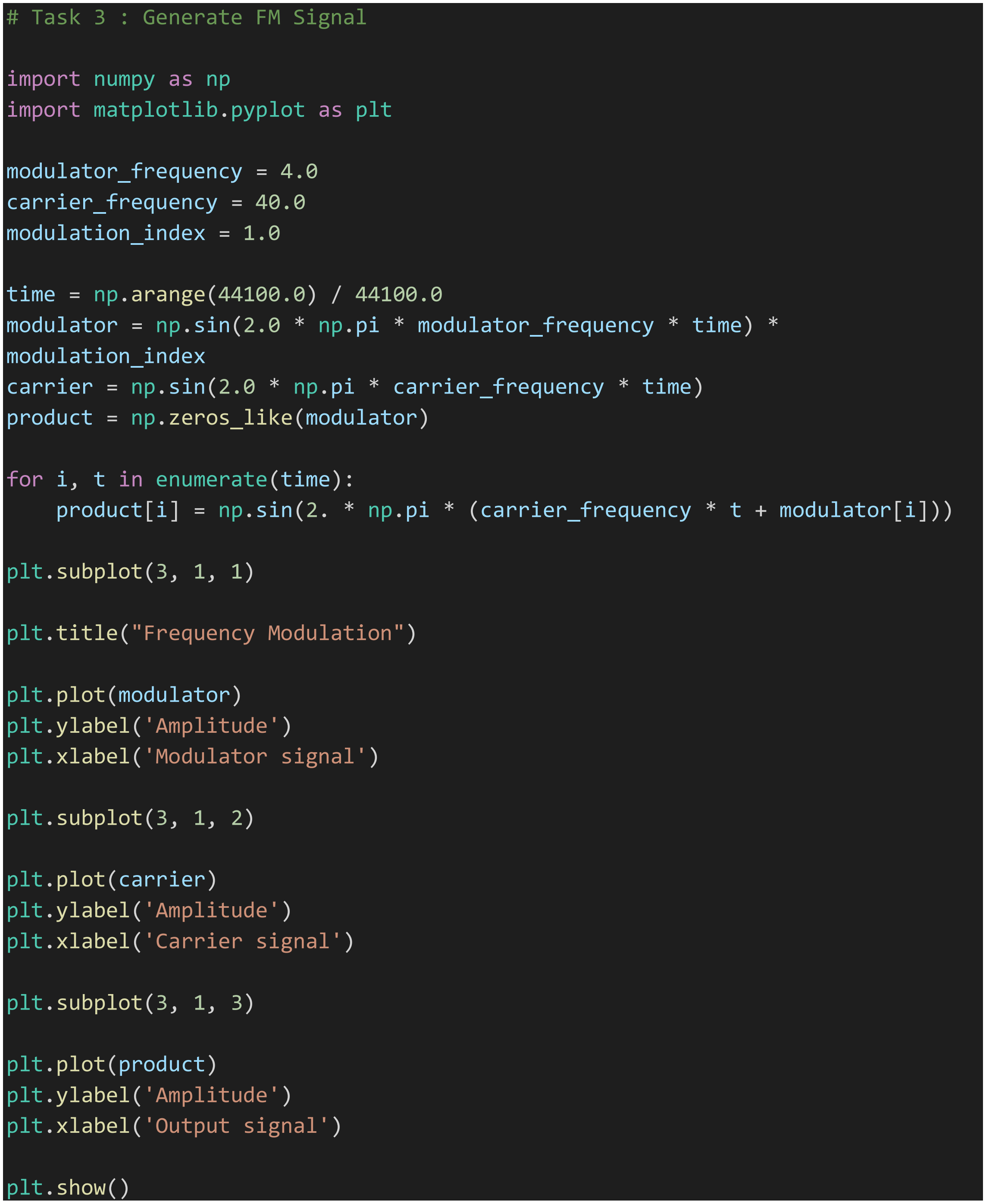
```
# To find the Eigen Values and Eigen Vectors of the Matrix
val, vect = linalg.eig(a)
print('To find the Eigen Values and Eigen Vectors of the Matrix')
print(val)
print('\n\n')
```



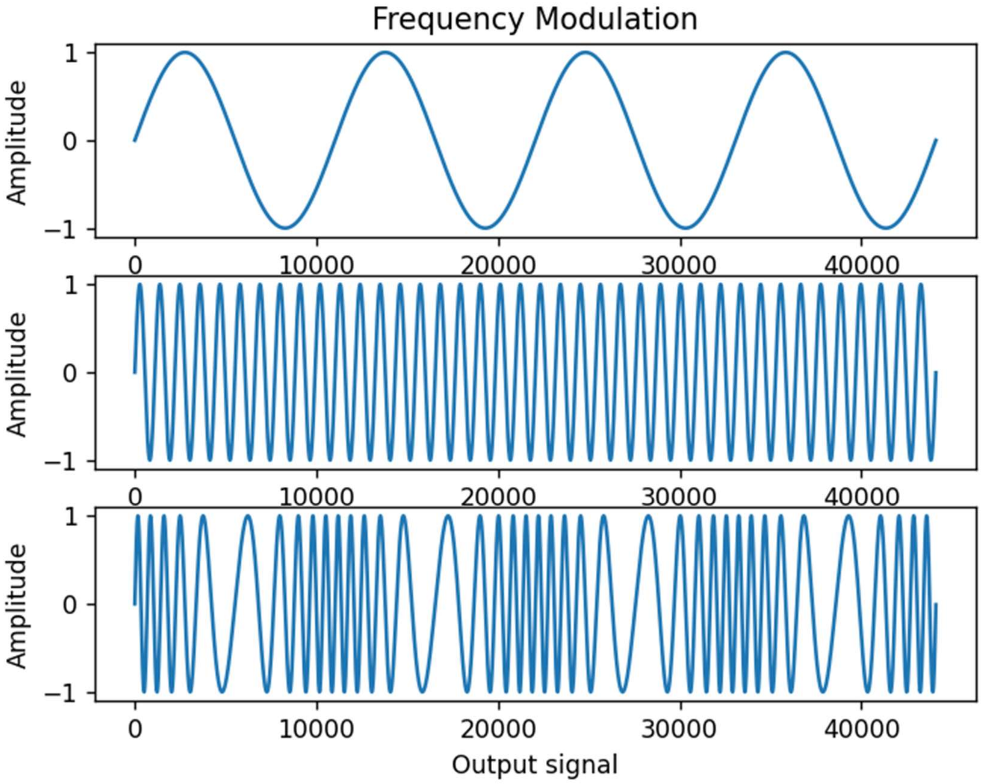
```
[0.74074074 1.40740741]
Solving the linear equations
[ 5.  3. -2.]



To find inverse of the matrix
[[ 1.28571429 -0.28571429 -0.14285714]
 [-0.47619048  0.14285714  0.23809524]
 [ 0.19047619  0.14285714 -0.0952381 ]]



To find Pseudo Inverse of a Matrix
[[ 1.28571429 -0.28571429 -0.14285714]
 [-0.47619048  0.14285714  0.23809524]
 [ 0.19047619  0.14285714 -0.0952381 ]]



To find the Determinant of the Matrix
-21.0
```

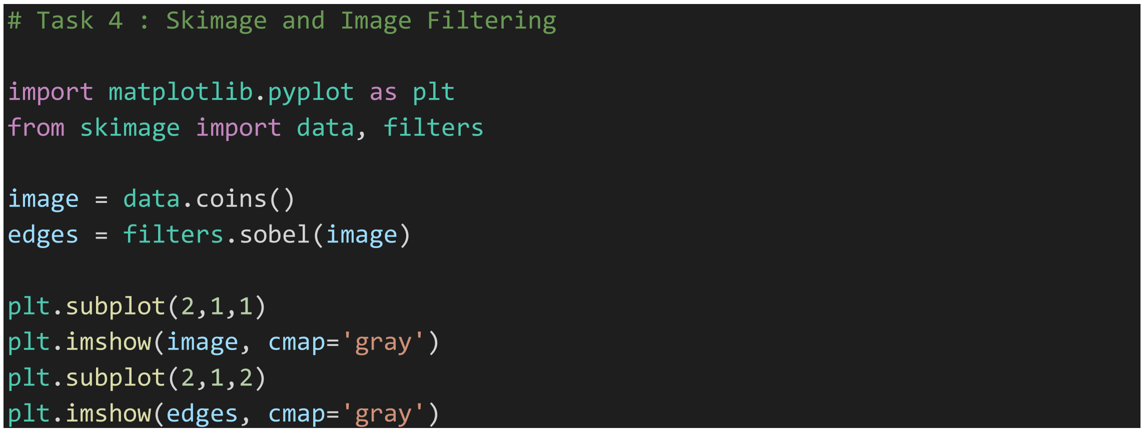


```
To find the Determinant of the Matrix
-21.0



To find the Eigen Values and Eigen Vectors of the Matrix
[-4.77879032+0.j  0.72601723+0.j  6.05277309+0.j]
```



```
# Task 3 : Generate FM Signal

import numpy as np
import matplotlib.pyplot as plt

modulator_frequency = 4.0
carrier_frequency = 40.0
modulation_index = 1.0

time = np.arange(44100.0) / 44100.0
modulator = np.sin(2.0 * np.pi * modulator_frequency * time) *
modulation_index
carrier = np.sin(2.0 * np.pi * carrier_frequency * time)
product = np.zeros_like(modulator)

for i, t in enumerate(time):
    product[i] = np.sin(2. * np.pi * (carrier_frequency * t + modulator[i]))

plt.subplot(3, 1, 1)

plt.title("Frequency Modulation")

plt.plot(modulator)
plt.ylabel('Amplitude')
plt.xlabel('Modulator signal')

plt.subplot(3, 1, 2)

plt.plot(carrier)
plt.ylabel('Amplitude')
plt.xlabel('Carrier signal')

plt.subplot(3, 1, 3)

plt.plot(product)
plt.ylabel('Amplitude')
plt.xlabel('Output signal')

plt.show()
```





```
# Task 4 : Skimage and Image Filtering

import matplotlib.pyplot as plt
from skimage import data, filters

image = data.coins()
edges = filters.sobel(image)

plt.subplot(2,1,1)
plt.imshow(image, cmap='gray')
plt.subplot(2,1,2)
plt.imshow(edges, cmap='gray')
```