**Assignment 3**

**IDS**



**Name:** Mohammad Ahmed Shahbaz

**Reg. #:** FA21-BSE-050

**Section:** BSE-C

**Question no 1**

matrix = [

[0, 0, 1, 0, 1, 0, 1],

[0, 1, 0, 0, 0, 1, 0],

[1, 0, 0, 0, 0, 1, 1],

[0, 0, 0, 0, 1, 0, 0],

[1, 0, 0, 1, 0, 0, 0],

[0, 1, 1, 0, 0, 0, 1],

[1, 0, 1, 0, 0, 1, 0]

]

import networkx as nx

import matplotlib.pyplot as plt

adj\_matrix = {

'a': {'c', 'e', 'g'},

'b': {'b', 'f'},

'c': {'a', 'f', 'g'},

'd': {'e'},

'e': {'a', 'd'},

'f': {'b', 'c', 'g'},

'g': {'a', 'c', 'f'}

}

G = nx.Graph()

G.add\_nodes\_from(adj\_matrix.keys())

for node, neighbors in adj\_matrix.items():

for neighbor in neighbors:

G.add\_edge(node, neighbor)

nx.draw(G, with\_labels=True, node\_color='lightblue', node\_size=1000)

plt.title("Undirected Graph Visualization")

plt.show()

if any(nx.cycle(G)):

print("There is a loop in the graph.")

print("The cycle(s) in the graph:", list(nx.simple\_cycles(G)))

else:

print("There is no loop in the graph.")

**Question no 2**

G = nx.Graph()

num\_nodes = len(matrix)

nodes = ('a', 'b', 'c', 'd', 'e', 'f', 'g')

G.add\_nodes\_from(nodes)

self\_loops = []

for i in range(num\_nodes):

for j in range(num\_nodes):

if matrix[i][j] == 1:

num = random.randint(2, 10)

G.add\_edge(nodes[i], nodes[j], weight=num)

nx.draw(G, with\_labels = True)

shortest\_path = nx.dijkstra\_path(G, source='a', target='b')

shortest\_path\_length = nx.dijkstra\_path\_length(G, source='a', target='b')

print("Shortest path:", shortest\_path)

print("Length of the shortest path:", shortest\_path\_length)