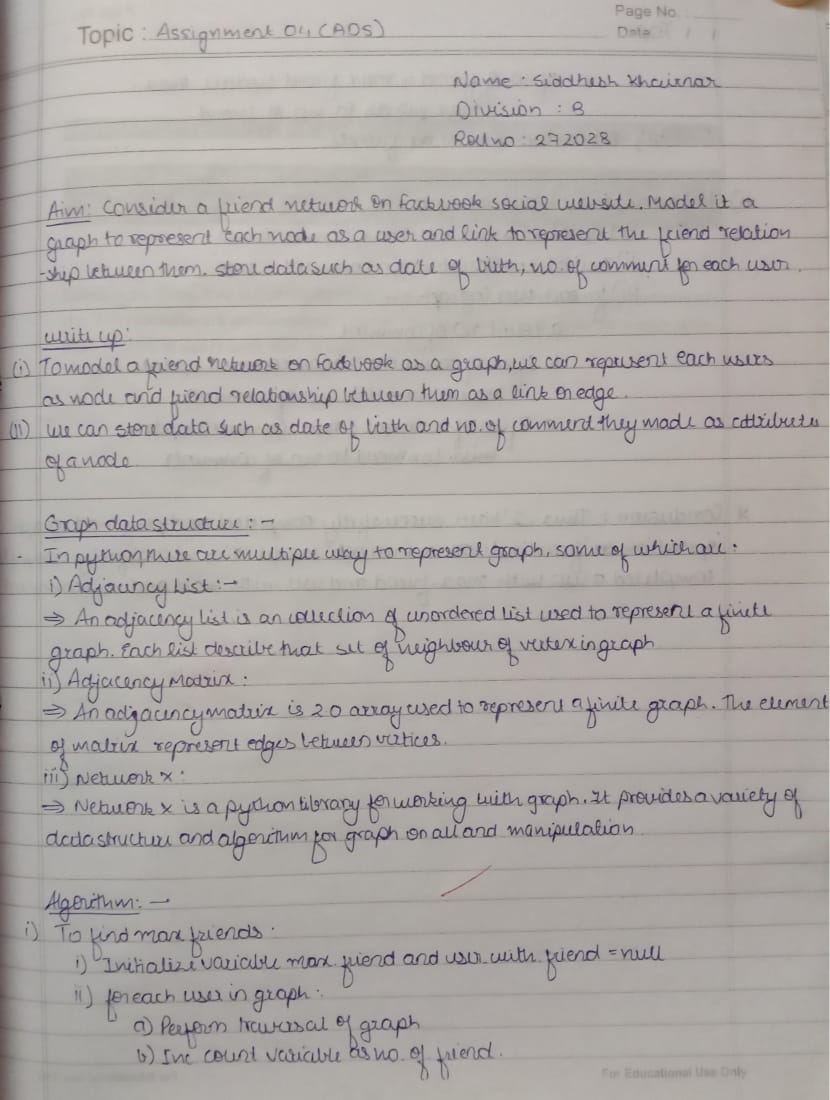
|  |  |  |  |
| --- | --- | --- | --- |
|  | Bansilal Ramnath Agarwal Charitable Trust's  Vishwakarma Institute of Information Technology  **Department of**  **Artificial Intelligence and Data Science** | | |
| Name: Siddhesh Dilip Khairnar | | | |
| Class: SY | Division: B | | Roll No: 272028 |
| Semester: IV | | Academic Year: 2022-2023 | |
| Subject Name & Code: Advanced Data Structure, ADUA22202 | | | |
| Title of Assignment: Consider a friends’ network on Facebook social website. Model it as a graph to represent each node as a user and a link to represent the friend relationship between them. Store data such as date of birth, number of comments for each User. | | | |

Aim: Consider a friends’ network on Facebook social website. Model it as a graph to represent each node as a user and a link to represent the friend relationship.

between them. Store data such as date of birth, number of comments for each

User:

1. Find who has maximum friends.
2. Find who has posted maximum and minimum comments.



Text, letter

Description automatically generated

Experiment:

import networkx as nx

# Create an empty graph

G = nx.Graph()

# Add nodes to the graph

G.add\_node('User1', dob='2000-01-01', num\_comments=100)

G.add\_node('User2', dob='2001-03-15', num\_comments=90)

G.add\_node('User3', dob='2002-07-30', num\_comments=120)

G.add\_node('User4', dob='2003-05-30', num\_comments=220)

G.add\_node('User5', dob='2003-03-13', num\_comments=360)

# Add edges to the graph

G.add\_edge('User1', 'User2')

G.add\_edge('User2', 'User3')

G.add\_edge('User3','User1')

G.add\_edge('User2','User4')

G.add\_edge('User3','User4')

G.add\_edge('User4','User5')

# Find the user with maximum friends

max\_friends = max(G.degree, key=lambda x: x[1])[0]

print(f"The user with maximum friends is {max\_friends}")

# Find the user with maximum comments

max\_comments = max(G.nodes, key=lambda x: G.nodes[x]['num\_comments'])

print(f"The user with maximum comments is {max\_comments}")

# Find the user with minimum comments

min\_comments = min(G.nodes, key=lambda x: G.nodes[x]['num\_comments'])

print(f"The user with minimum comments is {min\_comments}")

# Find users with birthdays in this month

import datetime

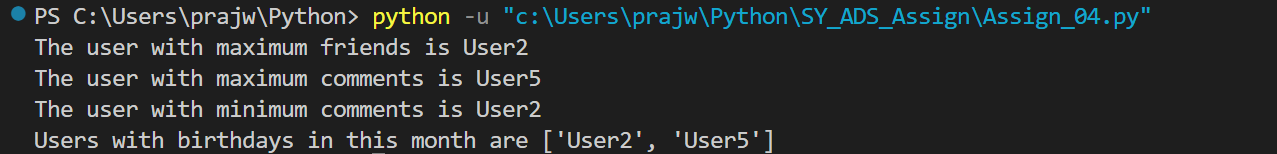
current\_month = datetime.datetime.now().month

birthday\_users = [

node for node in G.nodes if int(G.nodes[node]['dob'].split('-')[1]) == current\_month

]

print(f"Users with birthdays in this month are {birthday\_users}")

Output:  


Conclusion: Thus, I’ve successfully completed and performed graph operations on the ‘Friends network on Facebook’ Model. I’ve successfully computed a user with max. friends & users with min. & max. comments.