PROJECT

ON

Basic Social Network Analysis Tool

(Data Structures and Algorithms)

Overview:

This document outlines a basic social network analysis tool implemented in Python. The tool focuses on graph representation, degree centrality calculation, and basic visualization.?

Project Goals

The primary goals of this project are:

1. **Create a Graph-Based Representation of a Social Network**: Develop a structure where individuals are represented as nodes and their relationships as edges within a graph.
2. **Implement Degree Centrality Calculation**: Measure the significance of each individual within the network by calculating the number of connections (degree centrality) they have.
3. **Visualize the Social Network**: Generate a visual representation of the network, making it easier to understand the relationships and central figures within the social network.
4. **Provide a Simple and Accessible Tool**: Ensure the tool is straightforward to use, requiring only basic knowledge of Python, with clear and easy-to-understand outputs.

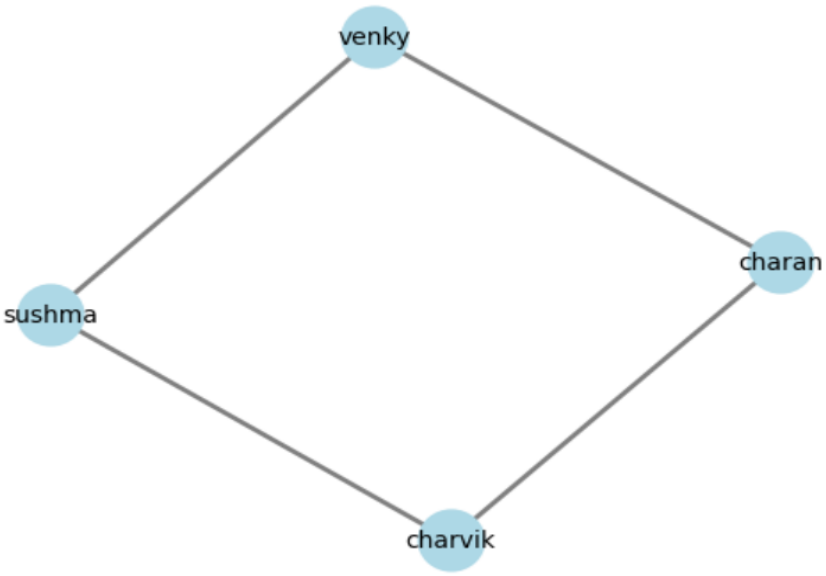
Code Structure:

Python:

```
import matplotlib.pyplot as plt

import networkx as nx

G = nx.Graph()
```

```
G.add_node("charvik")
G.add_node("sushma")
G.add_node("venky")
G.add_node("charan")

G.add_edge("charvik", "sushma")
G.add_edge("sushma", "venky")
G.add_edge("venky", "charan")
G.add_edge("charan", "charvik")

pos = nx.spring_layout(G)
nx.draw_networkx_nodes(G, pos, node_size=700, node_color="lightblue")
nx.draw_networkx_edges(G, pos, width=2, edge_color="gray")

nx.draw_networkx_labels(G, pos, font_size=10, font_family="sans-serif")

plt.axis("off")
plt.show()
```

Output:



Project Workflow

1. **Importing Libraries**:
    - The project begins by importing the necessary Python libraries: NetworkX for graph creation and analysis, and Matplotlib for visualization.
2. **Creating the Graph**:
    - An empty graph object is created, and nodes representing individuals are added to the graph. Edges are then added to represent the relationships between these individuals.
3. **Calculating Degree Centrality**:
    - Once the graph is constructed, the degree centrality for each node is calculated. This provides a measure of how many direct connections each individual has within the network.
4. **Visualizing the Network**:
    - The final step involves visualizing the graph. The nodes and edges are drawn, and labels are added to indicate the names of the individuals. The visualization is displayed in a window, providing a clear and concise view of the social network.

Explanation of the Project:

Here's a step-by-step explanation of the project:

Step 1: Importing Libraries

- import matplotlib.pyplot as plt: Imports the Matplotlib library, which is used for creating static, animated, and interactive visualizations. The as plt part assigns the alias plt to the library for easier access.
- import networkx as nx: Imports the NetworkX library, which is used for creating and analyzing complex networks. The as nx part assigns the alias nx to the library for easier access.

Step 2: Creating a Graph:-

- G = nx.Graph(): Creates a new, empty graph object G using NetworkX.

Step 3: Adding Nodes:-

- G.add_node("charvik"): Adds a node with the label "charvik" to the graph.
- G.add_node("sushma"): Adds a node with the label "sushma" to the graph.
- G.add_node("venky"): Adds a node with the label "venky" to the graph.
- G.add_node("charan"): Adds a node with the label "charan" to the graph.

Step 4: Adding Edges:-

- G.add_edge("charvik", "sushma"): Adds an edge between the nodes "charvik" and "sushma".
- G.add_edge("sushma", "venky"): Adds an edge between the nodes "sushma" and "venky".
- G.add_edge("venky", "charan"): Adds an edge between the nodes "venky" and "charan".
- G.add_edge("charan", "charvik"): Adds an edge between the nodes "charan" and "charvik".

Step 5: Positioning Nodes:-

- pos = nx.spring_layout(G): Uses the Fruchterman-Reingold force-directed algorithm to position the nodes in the graph. This algorithm tries to position nodes in a way that minimizes edge crossings and keeps nodes at a similar distance from each other.

Step 6: Drawing Nodes:-

- nx.draw_networkx_nodes(G, pos, node_size=700, node_color="lightblue"): Draws the nodes in the graph using the positions calculated in Step 5. The nodes are colored light blue, and their size is set to 700.

Step 7: Drawing Edges:-

- nx.draw_networkx_edges(G, pos, width=2, edge_color="gray"): Draws the edges in the graph using the positions calculated in Step 5. The edges are colored gray, and their width is set to 2.

Step 8: Drawing Labels:-

- nx.draw_networkx_labels(G, pos, font_size=10, font_family="sans-serif"): Draws the labels for each node in the graph using the positions calculated in Step 5. The labels are displayed in sans-serif font with a font size of 10.

Step 9: Showing the Plot

- plt.axis("off"): Turns off the axis for the plot, so only the graph is displayed.
- plt.show(): Displays the plot in a window.

This project creates a simple social network visualization using NetworkX and Matplotlib.

Applications and Use Cases

- **Educational Tool**: This project serves as an educational tool for those learning about social network analysis and graph theory. It provides a hands-on approach to understanding key concepts such as graph representation, centrality measures, and network visualization.
- **Basic Social Network Analysis**: The tool can be used for basic analysis of small social networks, helping users identify key individuals and understand the structure of their network.
- **Foundation for Advanced Projects**: While this tool is basic, it can serve as a foundation for more advanced social network analysis projects. Users can expand on the existing code to include additional metrics, larger networks, or more complex visualizations.

Conclusion:

This project provides a simple yet powerful introduction to social network analysis using Python. By representing a social network as a graph, calculating degree centrality, and visualizing the network, users can gain valuable insights into the structure and key players within a social network. The project is designed to be accessible and easy to use, making it a useful tool for both educational purposes and basic analysis.