Design Documentation for Social Network System

Output

```
NETMORK USER IMFO:
User: janemiller804, Followers: 29, Following: 29
User: janemiller806, Followers: 29, Following: 38
User: chrisgarciail0, Followers: 29, Following: 38
User: sarahdavis024, Followers: 29, Following: 22
User: janejohnson286, Followers: 22, Following: 22
User: janejohnson286, Followers: 19, Following: 36
User: janejons0844, Followers: 21, Following: 36
User: janejons0844, Followers: 32, Following: 36
User: janejons0844, Followers: 27, Following: 36
User: johnmiller909, Followers: 27, Following: 38
User: johnmiller909, Followers: 27, Following: 39
User: johnmiller909, Followers: 28, Following: 39
User: milydavis665, Followers: 29, Following: 31
User: annapsith13, Followers: 29, Following: 31
User: malponson4272, Followers: 30, Following: 39
User: milydavis665, Followers: 22, Following: 39
User: laurasmith298, Followers: 22, Following: 39
User: annajonson422, Followers: 28, Following: 29
User: johnmartiner025, Followers: 28, Following: 29
User: daviddavis7044, Followers: 28, Following: 29
User: johnmartiner025, Followers: 28, Following: 29
User: johnmartiner025, Followers: 28, Following: 29
User: johnmiller827, Followers: 27, Following: 28
User: laurajohnson607, Followers: 27, Following: 21
User: johnmiller827, Followers: 27, Following: 31
User: johnmiller827, Followers: 27, Following: 31
User: laurajohnson607, Followers: 27, Following: 31
User: laurajohnson607, Followers: 32, Following: 31
User: laurajohnson607, Followers: 32, Following: 31
User: laurajohnson607, Followers: 32, Following: 30
User: johnwilliams27, Followers: 32, Following: 31
User: laurajohnson607, Followers: 32, Following: 31
User: laurajohnson608, Followers: 32, Following: 31
User: laurajohnson608, Followers: 32, Following: 32
User: daviddavis16, Followers: 34, Following: 39
User: laurajohnson609, Followers: 39, Following: 31
User: laurajohnson609, Followers: 39, Following: 3
```

```
User: mikemartinez202, Followers: 24, Following: 27
User: mikemartinez202, Followers: 25, Following: 26
User: pionmartine308, Followers: 24, Following: 26
User: pionmartine308, Followers: 24, Following: 26
User: microdispue202, Followers: 28, Following: 22
User: axambrown887, Followers: 28, Following: 23
User: davidswith089, Followers: 39, Following: 25
User: alexjones799, Followers: 39, Following: 35
User: davidswith089, Followers: 39, Following: 35
User: davidswith089, Followers: 29, Following: 35
User: davidswith089, Followers: 29, Following: 25
User: davidswith089, Followers: 29, Following: 25
User: davidswith089, Followers: 29, Following: 26
User: comilyjohnson083, Followers: 24, Following: 27
User: pionmartine308, Followers: 24, Following: 29
User: pionmartine308, Followers: 24, Following: 29
User: pionmartine308, Followers: 24, Following: 29
User: pionmartine308, Followers: 28, Following: 29
User: pionmartine308, Followers: 28, Following: 29
User: mikejohnson0848, Followers: 28, Following: 29
User: mikejohnson086, Followers: 31, Following: 29
User: mikejohnson086, Followers: 28, Following: 29
User: mikejohnson086, Followers: 29, Following: 29
User: mikejohnson086, Followers: 39, Following: 29
User: mikejohnson086, Followers: 39, Following: 29
User: mikejohnson086, Followers: 39, Following: 29
User: mikejohnson087, Followers: 39, Following: 29
User: mikejohnson087, Followers: 39, Following: 29
User: mikejohnson087, Followers: 39, Following: 29
User: mikej
```

```
ser: alexrodriguez714, Followers: 24, Following: 29
ser: chrisjohnson590, Followers: 23, Following: 32
ser: alexjones642, Followers: 30, Following: 26
ser: sarahmiller134, Followers: 19, Following: 26
ser: laurasmith360, Followers: 28, Following: 22
ser: johnmiller182, Followers: 18, Following: 21
ser: annadavis608, Followers: 27, Following: 28
ser: johnjones697, Followers: 32, Following: 16
ser: alexjohnson477, Followers: 28, Following: 24
ser: chriswilliams394, Followers: 33, Following: 32
ser: emilymartinez672, Followers: 22, Following: 32
   ser: emilymartinez072, Followers: 22, Following: 32
  ser: emilymartinez072, Followers: 22, Following: 32
ser: chrissmith987, Followers: 29, Following: 23
ser: annarodriguez894, Followers: 22, Following: 21
ser: mikewilliams458, Followers: 29, Following: 30
ser: annabrown861, Followers: 31, Following: 23
ser: annajones012, Followers: 32, Following: 22
ser: johnwilliams519, Followers: 26, Following: 15
ser: alexjohnson466, Followers: 32, Following: 23
ser: johnrodriguez906, Followers: 25, Following: 25
ser: johnson466, Followers: 25, Following: 25
ser: alexjohnson466, Followers: 32, Following: 23
ser: johnrodriguez966, Followers: 25, Following: 26
ser: laurajones863, Followers: 25, Following: 26
ser: mikewilliams611, Followers: 29, Following: 24
ser: chrisbrown217, Followers: 39, Following: 24
ser: davidgarcia394, Followers: 35, Following: 24
ser: chrisbrown495, Followers: 33, Following: 27
ser: annagarcia146, Followers: 24, Following: 36
ser: mikedavis484, Followers: 34, Following: 33
ser: alexjones403, Followers: 25, Following: 33
ser: alexjones403, Followers: 26, Following: 23
ser: annasmith778, Followers: 26, Following: 21
ser: laurajohnson450, Followers: 20, Following: 23
ser: johnmiller823, Followers: 20, Following: 23
ser: chrissmith223, Followers: 34, Following: 23
ser: chrissmith223, Followers: 34, Following: 26
ser: chrissmith223, Followers: 21, Following: 23
ser: lauramartinez568, Followers: 21, Following: 25
ser: lauramartinez568, Followers: 21, Following: 28
ser: alexjones404, Followers: 32, Following: 25
ser: alexdavis955, Followers: 32, Following: 22
ser: alexdavis955, Followers: 31, Following: 22
ser: laurajones008, Followers: 31, Following: 26
ser: laurajones008, Followers: 31, Following: 26
ser: laurajones008, Followers: 31, Following: 26
ser: davidbrown896, Followers: 29, Following: 26
ser: davidbrown896, Followers: 29, Following: 32
   TWORK STATISTICS:
   stal number of users: 250
   verage number of connections: 28.164
    MOST CONNECTED USERS:
   st Connected Users:
   svidgarcia766
   suragarcia308
   ohnmartinez405
   rahmiller780
   uradavis646
    MOST INFLUENTIAL USERS:
   ost Influential Users:
   arahqarcia864
   suragarcia142
   rahmiller780
   ohnmartinez405
   rismiller840
  RIEND SUGGESTIONS:
   ikerodriguez273
   urabrown533
   inegarcia255
   hnwilliams519
   urajones008
   GREE OF SEPARATION (5 sets of users)
  gree of separation between janemartinez432 and alexrodriguez123: 2
gree of separation between johndavis865 and mikejones788: 2
    gree of separation between annadavis088 and emilyjohnson803: 2
   egree of separation between laurajohnson607 and alexwilliams476: 2
   gree of separation between johnjohnson736 and sarahmiller134: 2
```

Overview

This system simulates a basic social network with functionality to retrieve information about users, their connections, and statistical data. It includes features like determining the degree of separation between two users, suggesting friends, and identifying the most connected and influential users.

System Architecture

The system is designed using an object-oriented approach. The core components include:

- **User Struct**: Represents a user in the network.
- Adjacency List Class: Stores connections between users
- **Graph Class**: Represents the entire social network, containing the users and managing the relationships between them.
- **Main Program**: Implements various functions to interact with the graph object and display information to the user.

Key Components

1. Graph Class

The graph class serves as the main data structure for storing and managing the social network. It contains the following attributes and methods:

• Attributes:

- vertices: A self-balancing AVL tree (or similar data structure) containing the users of the network.
- o usernames: A dynamic array of strings containing the usernames of the users.
- o numUsrs: The total number of users in the network.
- o numCncts: The total number of connections in the network.

Methods:

- o **graph()**: Constructor that loads user data from a CSV file, initializes the usernames array, and establishes connections between users randomly.
- ~graph(): Destructor that cleans up dynamically allocated memory.
- o usrCt(): Returns the number of users in the network.
- o **avgConnectionCT()**: Returns the average number of connections in the network.
- sepDegree(): Calculates the degree of separation between two users, either by username or by user indices.

- o **print()**: Prints information about all the users, including their username, first and last names, and the number of followers and following.
- o **printFriendSuggestions()**: Prints friend suggestions for a given user based on mutual connections.
- o **printSeparationDegree()**: Prints the degree of separation between two users, either by username or by indices.
- o **printMostConnectedUser()**: Prints the most connected users based on the total number of followers and following.
- printMostInfluentialUser(): Prints the most influential users based on the number of followers' followers.
- o **printNumberOfUsers()**: Prints the total number of users in the network.
- printAverageNumberOfConnections(): Prints the average number of connections in the network.

2. User Class

The user class represents an individual user in the social network. It contains basic user information and manages user connections (followers and following).

Attributes:

- username: A string representing the user's unique identifier.
- o firstname: A string representing the user's first name.
- o lastname: A string representing the user's last name.
- o numFollowers: The number of users following this user.
- o numFollowing: The number of users this user is following.

Methods:

- o follow(user): Allows a user to follow another user, increasing the number of followers for the other user and the number of followings for this user.
- o *unFollow(user)*: Allows a user to unfollow another user, decreasing the number of followers for the other user and the number of followings for this user.
- view(string): Checks if a user is following another user.

Algorithms and Data Structures

1. Friend Suggestions

The suggestFriends method generates friend suggestions based on mutual friends (second-degree connections). It works by iterating through the user's following list and then retrieving their

followers' following lists. The result is sorted by the frequency of suggestions, ensuring that the most relevant users are suggested first.

• **Time Complexity**: O(n * m), where n is the number of users the given user is following, and m is the average number of users each friend is following.

2. Degree of Separation

The sepDegree method uses a breadth-first search (BFS) approach to find the degree of separation between two users. It starts at the first user and explores their connections until it finds the target user or exhausts all possible connections.

• **Time Complexity**: O(V + E), where V is the number of users and E is the number of connections (edges).

3. Most Connected Users

The mostConnected method ranks users by the sum of their followers and following counts. It uses a priority queue (heap) to extract the top resultCt most connected users efficiently.

• **Time Complexity**: O(n log n), where n is the number of users.

4. Most Influential Users

The mostInfluential method calculates the influence of each user by summing the followers' follower counts. It uses a priority queue to sort the users by their influence scores and returns the top resultCt influential users.

• **Time Complexity**: O(n * m log n), where n is the number of users and m is the average number of followers per user.

CSV Data Format

The program expects the CSV file user_data.csv to contain the following data format:

username,first_name,last_name

johndoe, John, Doe

janedoe, Jane, Doe

...

• Each row corresponds to a user, with the username, first name, and last name as commaseparated values.

Main Program Logic

- 1. **Initialization**: A graph object is created, and the user data is loaded from the CSV file. Connections between users are established randomly.
- 2. Information Display:

- Displays the network's user information, including the total number of users, average number of connections, and the most connected and influential users.
- 3. **Friend Suggestions**: For a specific user (e.g., "emilyrodriguez859"), the program generates and displays 5 friend suggestions.
- **4. Degree of Separation**: The program randomly selects pairs of users and calculates their degree of separation, displaying the result.

Conclusion

This social network simulation provides essential features for analyzing user relationships, including finding friend suggestions, determining degrees of separation, and identifying the most connected and influential users. The design follows object-oriented principles, making it easy to expand and adapt to new features.