CSE222 Homework8 Report

Akif Safa Angi

200104004079

1-) Introduction

This assignment involves the comprehensive implementation of a social network analysis system using graph data structures and algorithms. The objective is to simulate a social network and perform extensive analyses on the network, such as suggesting friends and counting clusters. Detailed instructions for each task will be provided to ensure clarity.

2-) Implementation Details

Main.java

1. Attributes

- `network`: An instance of `SocialNetworkGraph` to manage the social network.

- `scanner`: A `Scanner` object for reading user input.

- `formatter`: A `DateTimeFormatter` for formatting timestamps.

2. Main Method

- Initializes the social network with some default people.

- Continuously displays a menu to the user for performing various operations like adding/removing persons, adding/removing friendships, finding shortest paths, suggesting friends, and counting clusters.

- Handles user inputs and calls corresponding methods.

3. addPerson Method

- Prompts the user for the person's name, age, and hobbies.

- Adds the person to the network with the current timestamp.

4. removePerson Method

- Prompts the user for the person's name and timestamp.

- Removes the person from the network if they exist.

5. addFriendship Method

- Prompts the user for the names and timestamps of the two persons.

- Adds a friendship between the two persons if they exist.

6. removeFriendship Method

- Prompts the user for the names and timestamps of the two persons.

- Removes the friendship between the two persons if they exist.

7. findShortestPath Method

- Prompts the user for the start and end persons' names and timestamps.

- Finds and prints the shortest path between the two persons if they exist.

8. suggestFriends Method

- Prompts the user for the person's name, timestamp, and maximum number of friends to suggest.

- Suggests friends based on mutual friends and common hobbies.

9. countClusters Method

- Counts and prints the number of clusters in the network.

SocialNetworkGraph.java

1. Attributes

- `people`: A map of person names to `Person` objects.

- `friendships`: A map of `Person` objects to their list of friends.

2. addPerson Method

- Adds a new person to the network with the provided name, age, hobbies, and the current timestamp.

3. removePerson Method

- Removes a person from the network and updates the friendships accordingly.

4. addFriendship Method

- Adds a bidirectional friendship between two persons if they exist in the network.

5. removeFriendship Method

- Removes the bidirectional friendship between two persons if they exist in the network.

6. findShortestPath Method

- Implements BFS to find the shortest path between two persons.

- Prints the path if found, otherwise indicates no path exists.

7. countClusters Method

- Uses BFS to find and count clusters of connected people in the network.

- Prints the number of clusters and the members of each cluster.

8. suggestFriends Method

- Suggests friends for a person based on mutual friends and common hobbies.

- Ranks potential friends by the number of mutual friends and common hobbies.

Person.java

1. Attributes

- `name`: The name of the person.
- `age`: The age of the person.
- `hobbies`: A list of hobbies of the person.
- `timestamp`: The timestamp when the person was added to the network.

2. Constructor

- Initializes the person's attributes.

3. Getters

- Provides methods to retrieve the person's name, age, hobbies, and timestamp.

4. toString Method

- Returns a string representation of the person.

3-) How to run

To run program:

Writ to terminal “**make clean**” clean *.class files.

Write “**make**” to compile

Write “**make run**” to run program

To create a JavaDoc write “**make doc**” to terminal

To delete JavaDoc files write “**make cleandoc**” to terminal

4-) The Parts I Have Completed

I completed all tasks.

5-) Tests

Adding Persons

```
// Adding some people for demonstration
network.addPerson("Akif Angi", 25, Arrays.asList("reading", "hiking", "cooking", "watching", "basketball", "gaming"));
network.addPerson("Safa Angi", 22, Arrays.asList("swimming", "cooking", "watching", "reading"));
network.addPerson("Ece Tek", 27, Arrays.asList("hiking", "painting"));
network.addPerson("Burak Kal", 30, Arrays.asList("reading", "swimming", "hiking"));
network.addPerson("Ahmet Ali", 28, Arrays.asList("running", "swimming", "watching", "painting"));
network.addPerson("Zeybep Bir", 26, Arrays.asList("reading", "hiking", "cooking", "running"));
```

```
Person added: Akif Angi (Timestamp: 2024-05-29 19:52:48)
Person added: Safa Angi (Timestamp: 2024-05-29 19:52:48)
Person added: Ece Tek (Timestamp: 2024-05-29 19:52:48)
Person added: Burak Kal (Timestamp: 2024-05-29 19:52:48)
Person added: Ahmet Ali (Timestamp: 2024-05-29 19:52:48)
Person added: Zeybep Bir (Timestamp: 2024-05-29 19:52:48)
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option:
```

Adding Friendships

```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3
Enter first person's name: Akif Angi
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter second person's name: Ece Tek
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Friendship added between Akif Angi and Ece Tek
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3
Enter first person's name: Safa Angi
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter second person's name: Ahmet Ali
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Friendship added between Safa Angi and Ahmet Ali
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3
Enter first person's name: Burak Kal
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter second person's name: Zeybep Bir
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Friendship added between Burak Kal and Zeybep Bir
```

```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 7
Number of clusters: 3
Cluster 1:
Safa Angi
Ahmet Ali
Cluster 2:
Burak Kal
Zeybep Bir
Cluster 3:
Akif Angi
Ece Tek
```

Adding more friendships and shortest path

```
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3
Enter first person's name: Ahmet Ali
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter second person's name: Ece Tek
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Friendship added between Ahmet Ali and Ece Tek
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3
Enter first person's name: Ece Tek
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter second person's name: Burak Kal
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Friendship added between Ece Tek and Burak Kal
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 5
Enter start person's name: Akif Angi
Enter start person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter end person's name: Ahmet Ali
Enter end person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Shortest path: Akif Angi -> Ece Tek -> Ahmet Ali
```

Suggest Friend

```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 6
Enter person's name: Akif Angi
Enter person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter maximum number of friends to suggest: 3
Suggested friends for Akif Angi:
Burak Kal (Score: 2.0, 1 mutual friends, 2 common hobbies)
Ahmet Ali (Score: 1.5, 1 mutual friends, 1 common hobbies)
```

Remove Friendship

```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 4
Enter first person's name: Ece Tek
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Enter second person's name: Ahmet Ali
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-29 19:52:48
Friendship removed between Ece Tek and Ahmet Ali
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 7
Number of clusters: 2
Cluster 1:
Safa Angi
Ahmet Ali
Cluster 2:
Burak Kal
Zeybep Bir
Ece Tek
Akif Angi
```

Remove Person and exit