GTU DEPARTMENT OF COMPUTER ENGINEERING

CSE 222

HOMEWORK #8

EMİR İNCE

200104004025

How to Run The Code?

make: Type “make” to compile the code.

make clean: Type “make clean" to clean the output files.

make run: Type “make run” to run the program.

make doc: Type “make doc” to create Javadoc files.

make cleandoc: Type “make cleandoc” to delete Javadoc files.

Overview

The Java code provided implements a console-based social network analysis tool. It allows users to interactively manage a social network graph by adding and removing people, establishing and dissolving friendships, finding the shortest path between two people, suggesting friends, and counting clusters within the network. The implementation involves three main classes: *Main*, *Person*, and *SocialNetworkGraph*.

Class Descriptions

Main Class

This class serves as the entry point for the application. It contains the main method and handles user interactions through a console menu. The class utilizes the *SocialNetworkGraph* class to perform various operations on the social network.

Attributes

formatter: A *DateTimeFormatter* object for parsing and formatting date-time strings.

Methods

main(String[] args): Contains the main loop for the menu-driven interface, allowing users to select various options for managing the social network.

Menu Options

1. Add Person: Prompts the user to enter details (name, age, hobbies) and adds the person to the network.

2. Remove Person: Prompts the user to enter the person's name and creation timestamp, then removes the person from the network.

3. Add Friendship: Prompts the user to enter the names and timestamps of two people and establishes a friendship between them.

4. Remove Friendship: Prompts the user to enter the names and timestamps of two people and removes the friendship between them.

5. Find Shortest Path: Prompts the user to enter the names and timestamps of two people and finds the shortest path between them using *Breadth-First Search (BFS)*.

6. Suggest Friends: Prompts the user to enter a person's name, timestamp, and the maximum number of friend suggestions, then suggests friends based on mutual friends and common hobbies.

7. Count Clusters: Counts and displays the number of clusters in the network, where a cluster is a group of connected people.

8. Exit: Exits the program.

Person Class

This class represents a person in the social network. It stores the person's name, age, hobbies, and the timestamp of when the person was created. The class includes getter methods for these attributes and overrides the *toString* method to return the person's name.

Attributes

name: The name of the person.

age: The age of the person.

hobbies: A list of hobbies of the person.

timestamp: The timestamp of when the person object was created.

Methods

Person(String name, int age, List<String> hobbies, LocalDateTime timestamp): Constructor to initialize the person's details.

Getter methods: getName(), getAge(), getHobbies(), getTimestamp().

toString(): Returns the name of the person.

SocialNetworkGraph Class

This class manages the social network graph, which consists of people and the friendships between them. It provides methods to add and remove people, add and remove friendships, find the shortest path between two people, suggest friends, and count clusters in the network.

Attributes

formatter: A DateTimeFormatter object for consistent timestamp formatting.

people: A map to store people in the network by their names.

graph: A map representing friendships, mapping each person to their list of friends.

Methods

SocialNetworkGraph(): Constructor initializes the people and graph maps.

addPerson(String name, int age, List<String> hobbies): Adds a new person to the network.

removePerson(String name, LocalDateTime timestamp): Removes a person from the network based on their name and creation timestamp.

addFriendship(String name1, LocalDateTime timestamp1, String name2, LocalDateTime timestamp2): Establishes a friendship between two people.

removeFriendship(String name1, LocalDateTime timestamp1, String name2, LocalDateTime timestamp2): Removes a friendship between two people.

findShortestPath(String name1, LocalDateTime timestamp1, String name2, LocalDateTime timestamp2): Finds the shortest path between two people using BFS.

suggestFriends(String name, LocalDateTime timestamp, int maxSuggestions): Suggests friends for a person based on mutual friends and common hobbies.

countClusters(): Counts and displays the number of clusters in the network.

Key Functionalities

Adding a Person

- Prompts the user for the person's name, age, and hobbies.
- Creates a Person object with the current timestamp.
- Adds the person to the people map and initializes their friends list in the graph.

Removing a Person

- Prompts the user for the person's name and timestamp.
- Validates the person and timestamp, removes the person from the people map and the graph, and removes them from all friends' lists.

Managing Friendships

- Adds or removes friendships between two people by updating their friends lists in the graph.

Finding the Shortest Path

- Uses BFS to find the shortest path between two people and logs the path if found.

Suggesting Friends

- Suggests friends based on mutual friends and common hobbies, scoring each suggestion and sorting them by score.

Counting Clusters

- Uses BFS to explore and count clusters of connected people, logging each cluster's members.

Error Handling

- The code includes error handling for user inputs and operations, printing appropriate error messages if exceptions occur or if people are not found.

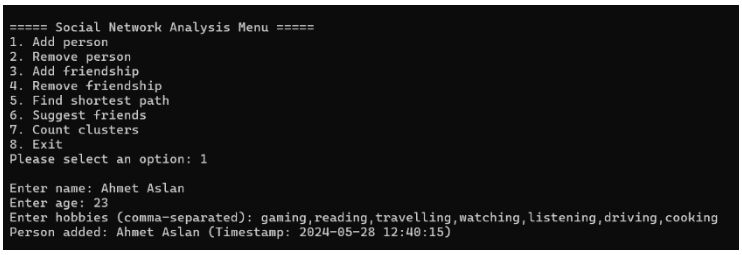
Conclusion

This Java implementation provides a robust tool for analyzing social networks, offering functionalities for managing people and friendships, analyzing connections, and suggesting new friends based on mutual interests and connections. The use of BFS for finding shortest paths and counting clusters ensures efficient traversal of the network. The detailed error handling and user prompts make the tool user-friendly and resilient to incorrect inputs.

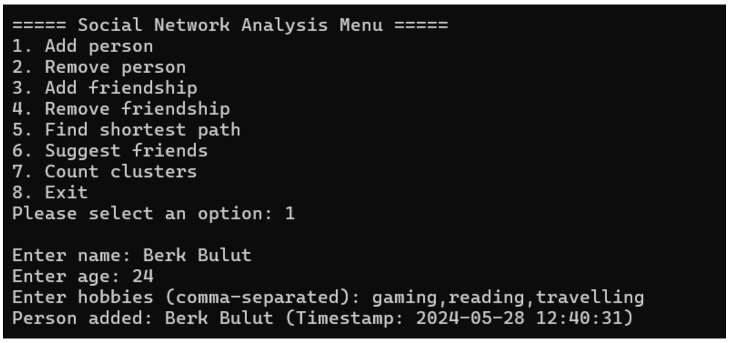
Outputs of the Code

This is the whole outputs of a scenario that I implemented for this program.

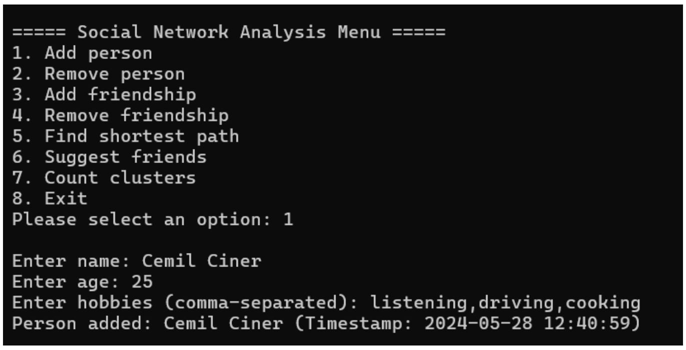
Add person



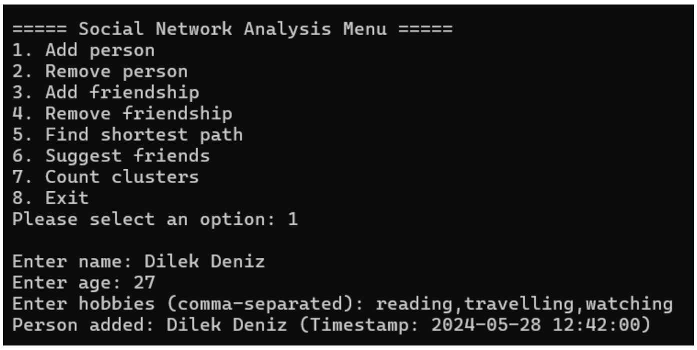
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Ahmet Aslan
Enter age: 23
Enter hobbies (comma-separated): gaming,reading,travelling,watching,listening,driving,cooking
Person added: Ahmet Aslan (Timestamp: 2024-05-28 12:40:15)
```



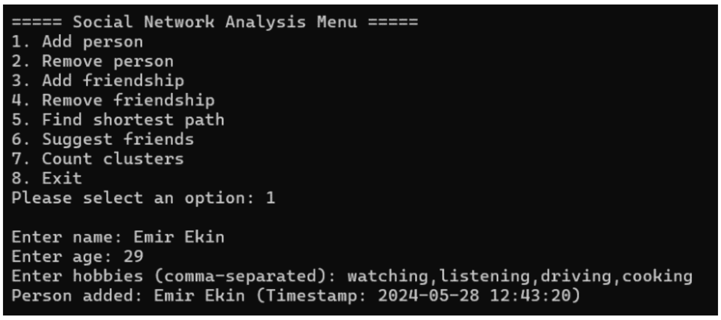
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Berk Bulut
Enter age: 24
Enter hobbies (comma-separated): gaming,reading,travelling
Person added: Berk Bulut (Timestamp: 2024-05-28 12:40:31)
```



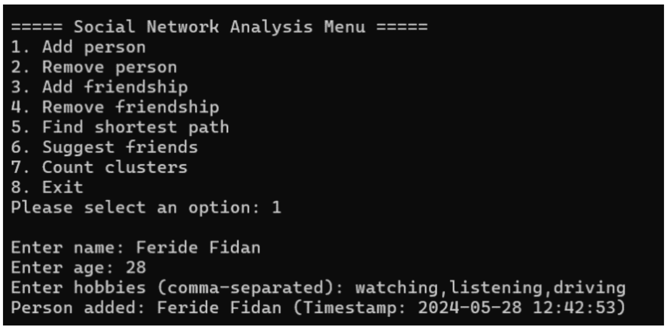
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Cemil Ciner
Enter age: 25
Enter hobbies (comma-separated): listening,driving,cooking
Person added: Cemil Ciner (Timestamp: 2024-05-28 12:40:59)
```



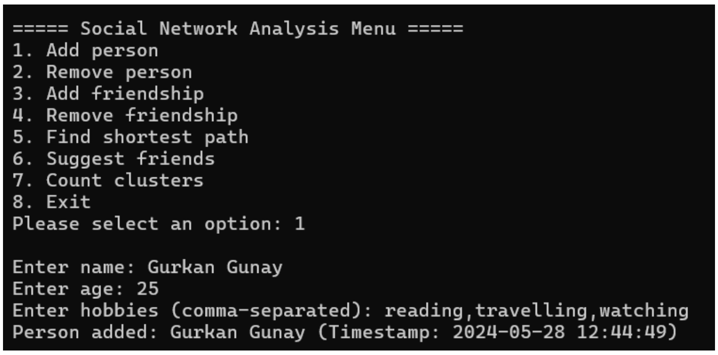
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Dilek Deniz
Enter age: 27
Enter hobbies (comma-separated): reading,travelling,watching
Person added: Dilek Deniz (Timestamp: 2024-05-28 12:42:00)
```



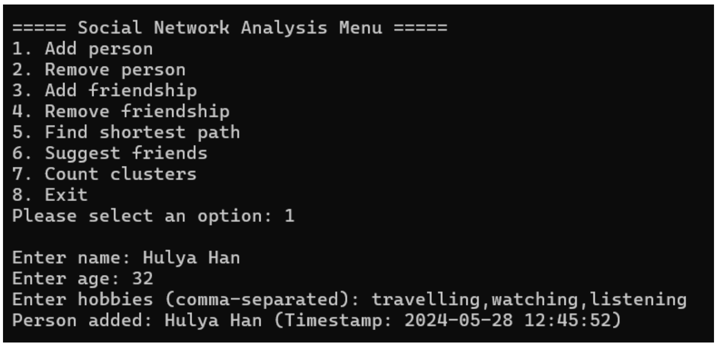
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Emir Ekin
Enter age: 29
Enter hobbies (comma-separated): watching,listening,driving,cooking
Person added: Emir Ekin (Timestamp: 2024-05-28 12:43:20)
```



```
===== 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: 1

Enter name: Feride Fidan
Enter age: 28
Enter hobbies (comma-separated): watching,listening,driving
Person added: Feride Fidan (Timestamp: 2024-05-28 12:42:53)
```

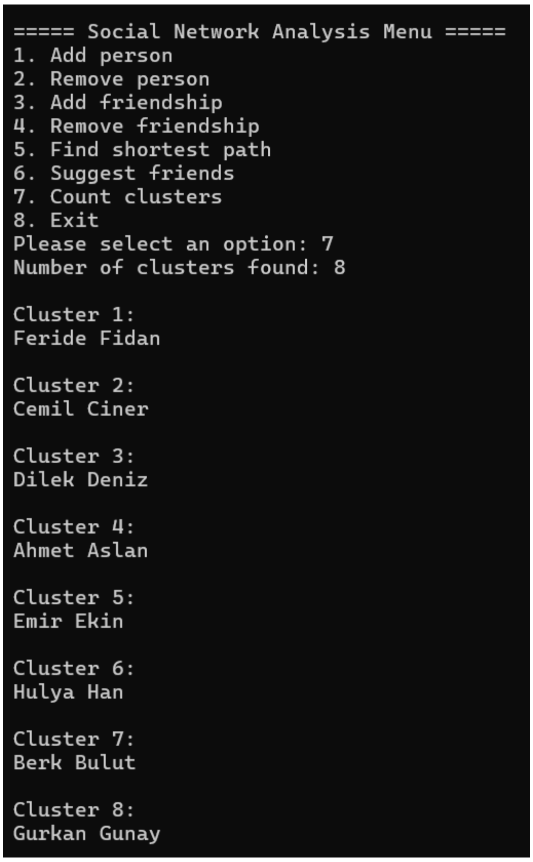


```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Gurkan Gunay
Enter age: 25
Enter hobbies (comma-separated): reading,travelling,watching
Person added: Gurkan Gunay (Timestamp: 2024-05-28 12:44:49)
```



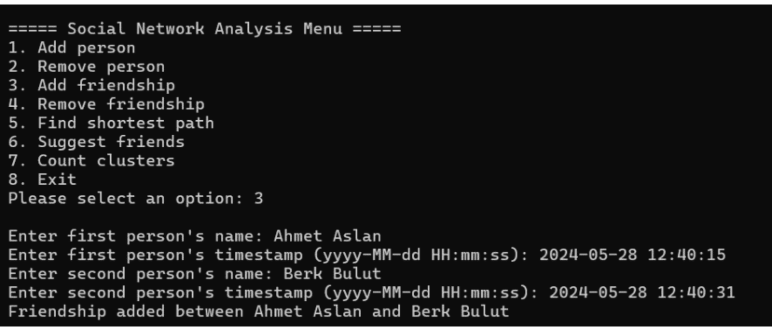
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 1

Enter name: Hulya Han
Enter age: 32
Enter hobbies (comma-separated): travelling,watching,listening
Person added: Hulya Han (Timestamp: 2024-05-28 12:45:52)
```

Clusters

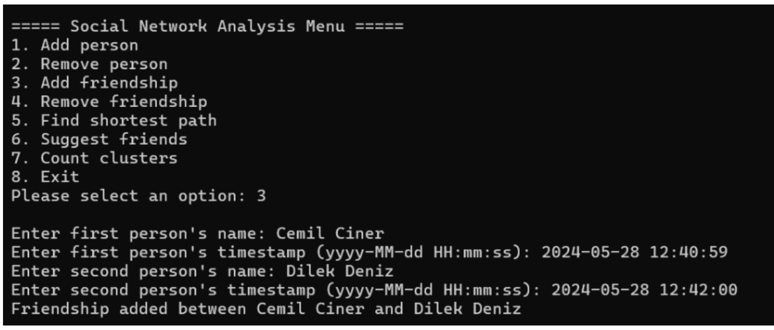


```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 7
Number of clusters found: 8

Cluster 1:
Feride Fidan

Cluster 2:
Cemil Ciner

Cluster 3:
Dilek Deniz

Cluster 4:
Ahmet Aslan

Cluster 5:
Emir Ekin

Cluster 6:
Hulya Han

Cluster 7:
Berk Bulut

Cluster 8:
Gurkan Gunay
```

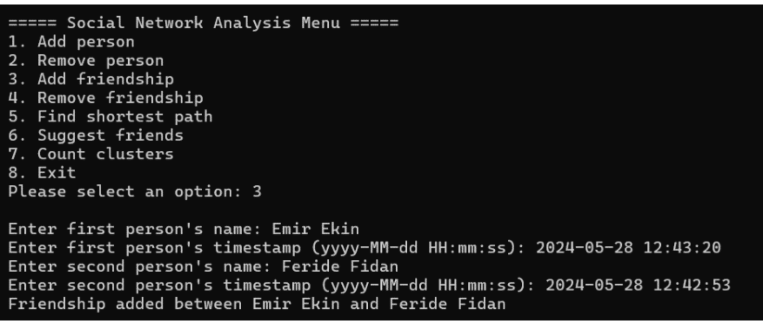
Add Friendship



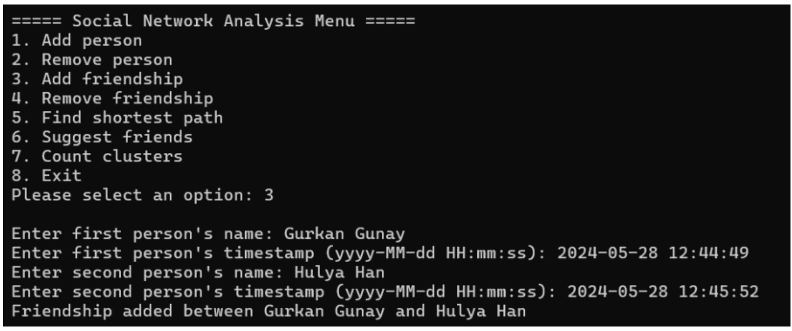
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3

Enter first person's name: Ahmet Aslan
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:15
Enter second person's name: Berk Bulut
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:31
Friendship added between Ahmet Aslan and Berk Bulut
```



```
===== 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: Cemil Ciner
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:59
Enter second person's name: Dilek Deniz
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:42:00
Friendship added between Cemil Ciner and Dilek Deniz
```

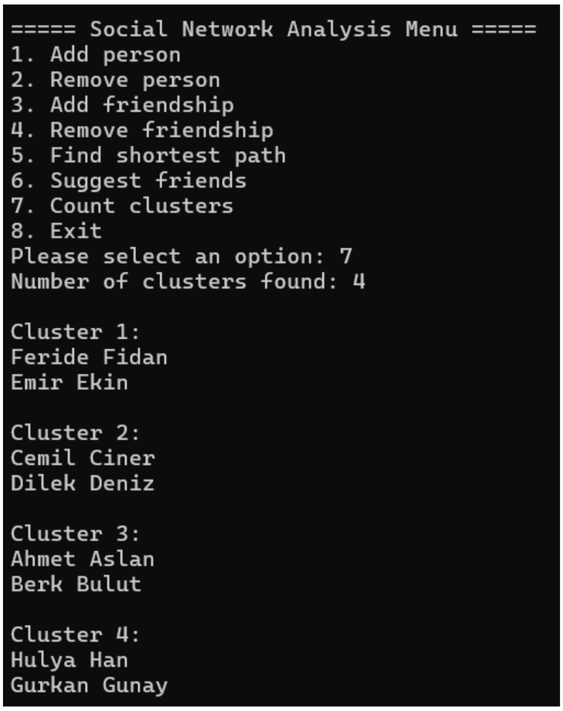


```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3

Enter first person's name: Emir Ekin
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:43:20
Enter second person's name: Feride Fidan
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:42:53
Friendship added between Emir Ekin and Feride Fidan
```



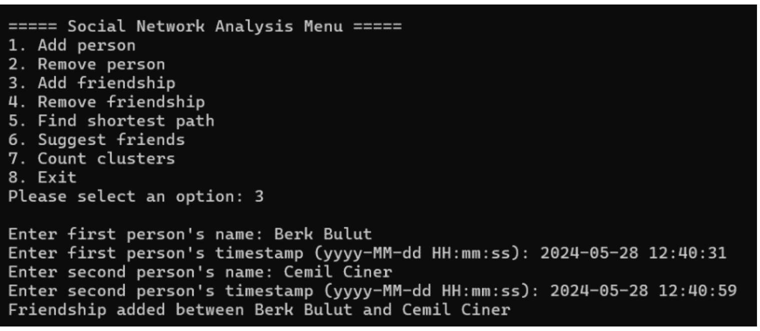
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3

Enter first person's name: Gurkan Gunay
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:44:49
Enter second person's name: Hulya Han
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:45:52
Friendship added between Gurkan Gunay and Hulya Han
```

Clusters

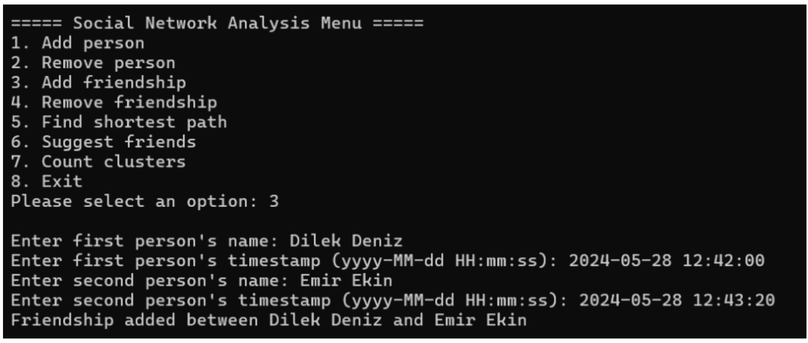


```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 7
Number of clusters found: 4

Cluster 1:
Feride Fidan
Emir Ekin

Cluster 2:
Cemil Ciner
Dilek Deniz

Cluster 3:
Ahmet Aslan
Berk Bulut

Cluster 4:
Hulya Han
Gurkan Gunay
```

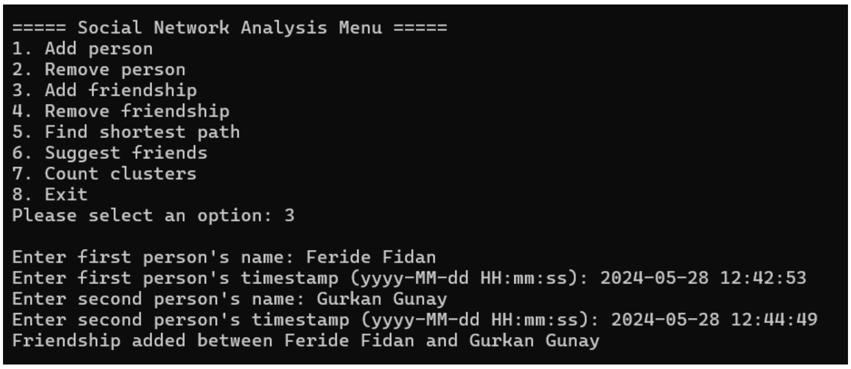
Add Friendship



```
===== 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: 3

Enter first person's name: Berk Bulut
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:31
Enter second person's name: Cemil Ciner
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:59
Friendship added between Berk Bulut and Cemil Ciner
```

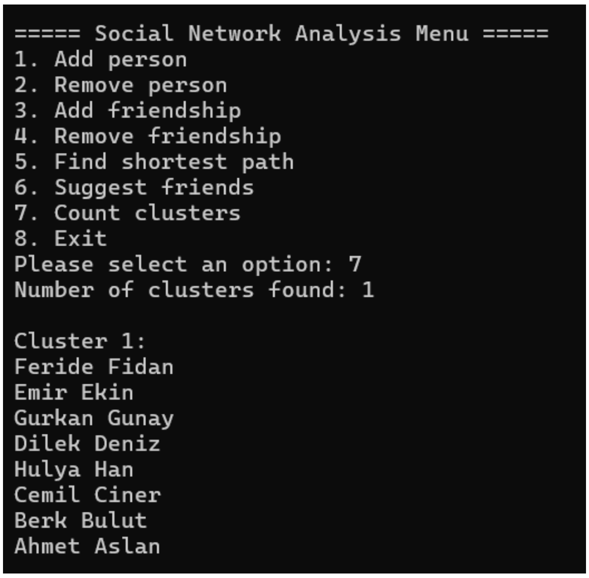


```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3

Enter first person's name: Dilek Deniz
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:42:00
Enter second person's name: Emir Ekin
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:43:20
Friendship added between Dilek Deniz and Emir Ekin
```



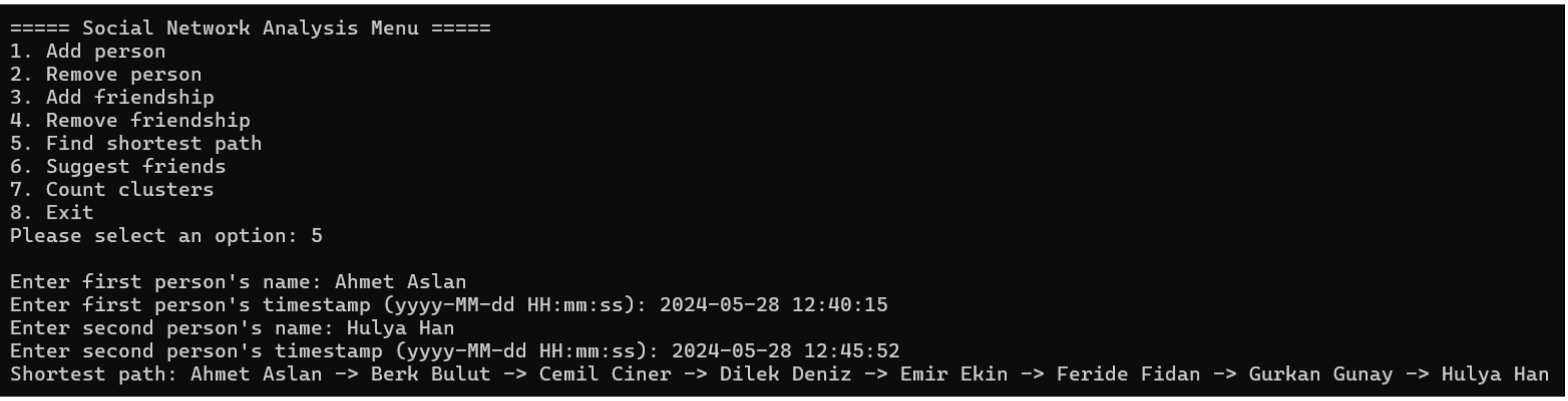
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3

Enter first person's name: Feride Fidan
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:42:53
Enter second person's name: Gurkan Gunay
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:44:49
Friendship added between Feride Fidan and Gurkan Gunay
```

Clusters



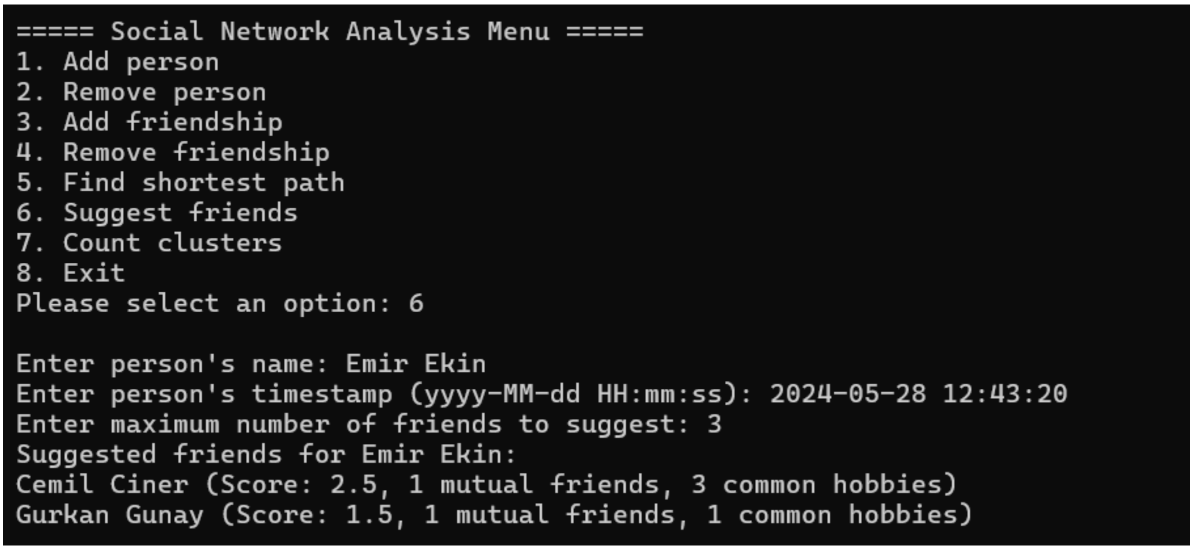
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 7
Number of clusters found: 1

Cluster 1:
Feride Fidan
Emir Ekin
Gurkan Gunay
Dilek Deniz
Hulya Han
Cemil Ciner
Berk Bulut
Ahmet Aslan
```

Shortest Path



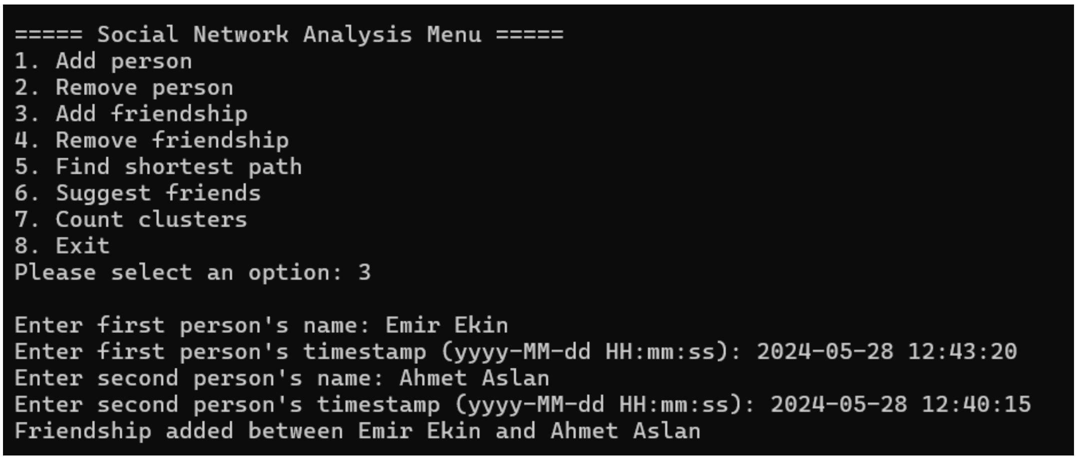
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 5

Enter first person's name: Ahmet Aslan
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:15
Enter second person's name: Hulya Han
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:45:52
Shortest path: Ahmet Aslan -> Berk Bulut -> Cemil Ciner -> Dilek Deniz -> Emir Ekin -> Feride Fidan -> Gurkan Gunay -> Hulya Han
```

Suggest Friend



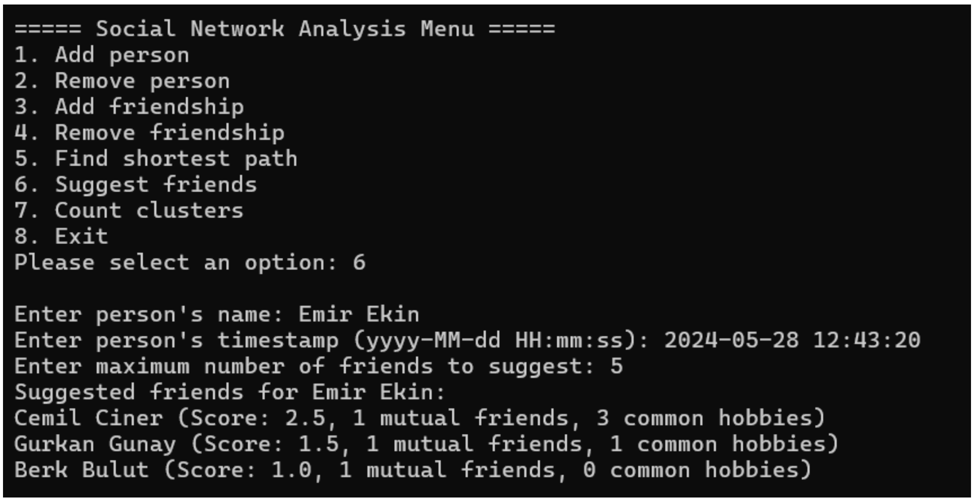
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 6

Enter person's name: Emir Ekin
Enter person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:43:20
Enter maximum number of friends to suggest: 3
Suggested friends for Emir Ekin:
Cemil Ciner (Score: 2.5, 1 mutual friends, 3 common hobbies)
Gurkan Gunay (Score: 1.5, 1 mutual friends, 1 common hobbies)
```

Add Friendship



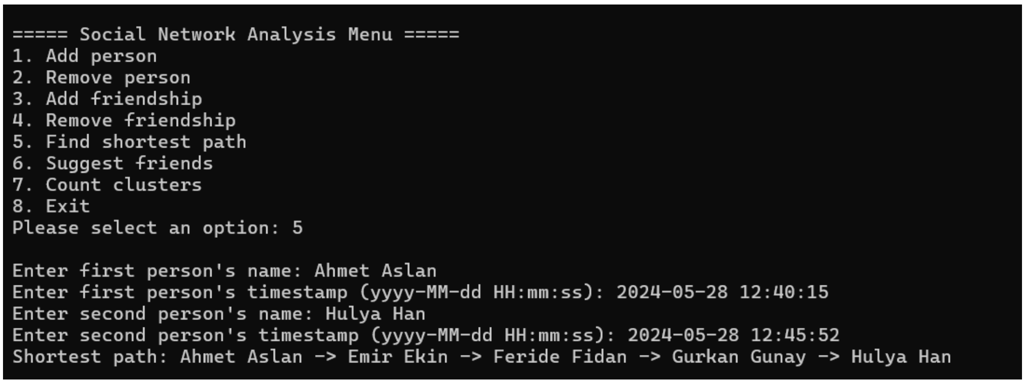
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 3

Enter first person's name: Emir Ekin
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:43:20
Enter second person's name: Ahmet Aslan
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:15
Friendship added between Emir Ekin and Ahmet Aslan
```

Suggest Friendship



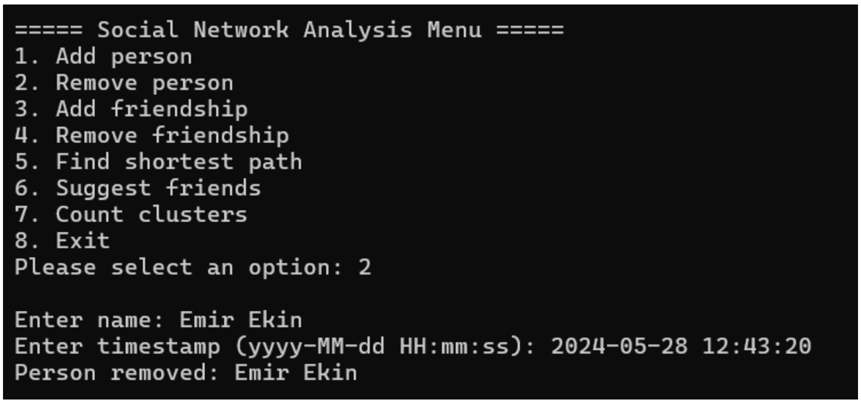
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 6

Enter person's name: Emir Ekin
Enter person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:43:20
Enter maximum number of friends to suggest: 5
Suggested friends for Emir Ekin:
Cemil Ciner (Score: 2.5, 1 mutual friends, 3 common hobbies)
Gurkan Gunay (Score: 1.5, 1 mutual friends, 1 common hobbies)
Berk Bulut (Score: 1.0, 1 mutual friends, 0 common hobbies)
```

Shortest Path



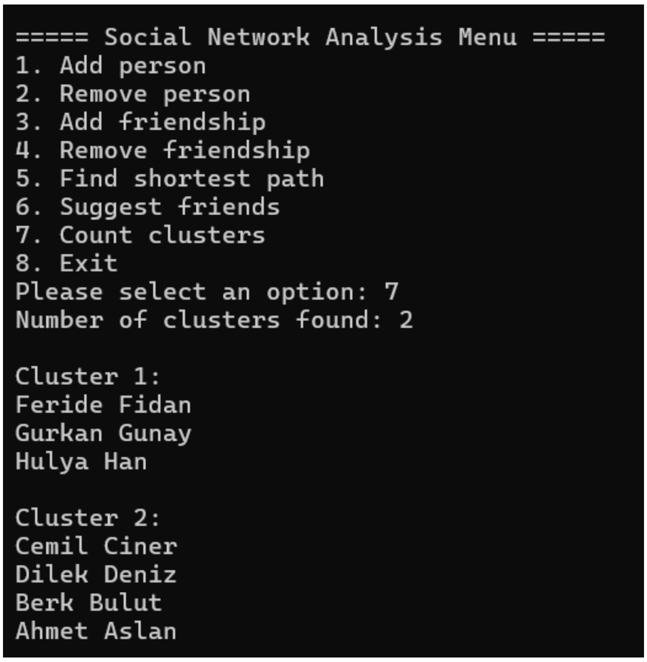
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 5

Enter first person's name: Ahmet Aslan
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:15
Enter second person's name: Hulya Han
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:45:52
Shortest path: Ahmet Aslan -> Emir Ekin -> Feride Fidan -> Gurkan Gunay -> Hulya Han
```

Remove Person



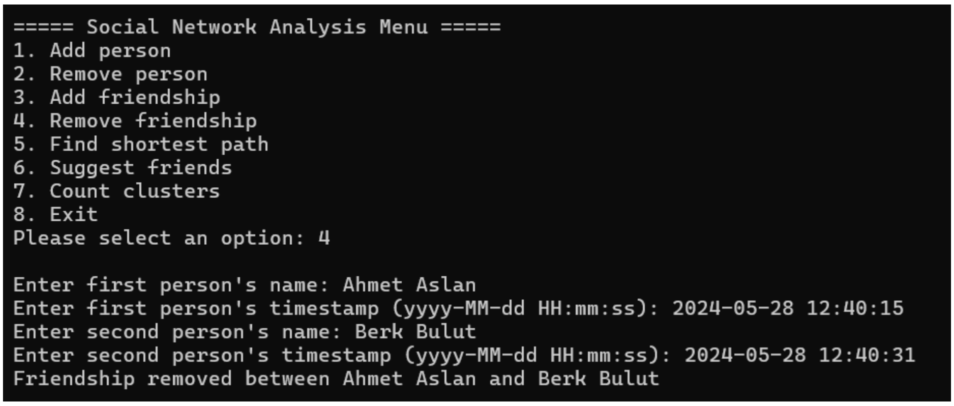
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 2

Enter name: Emir Ekin
Enter timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:43:20
Person removed: Emir Ekin
```

Clusters



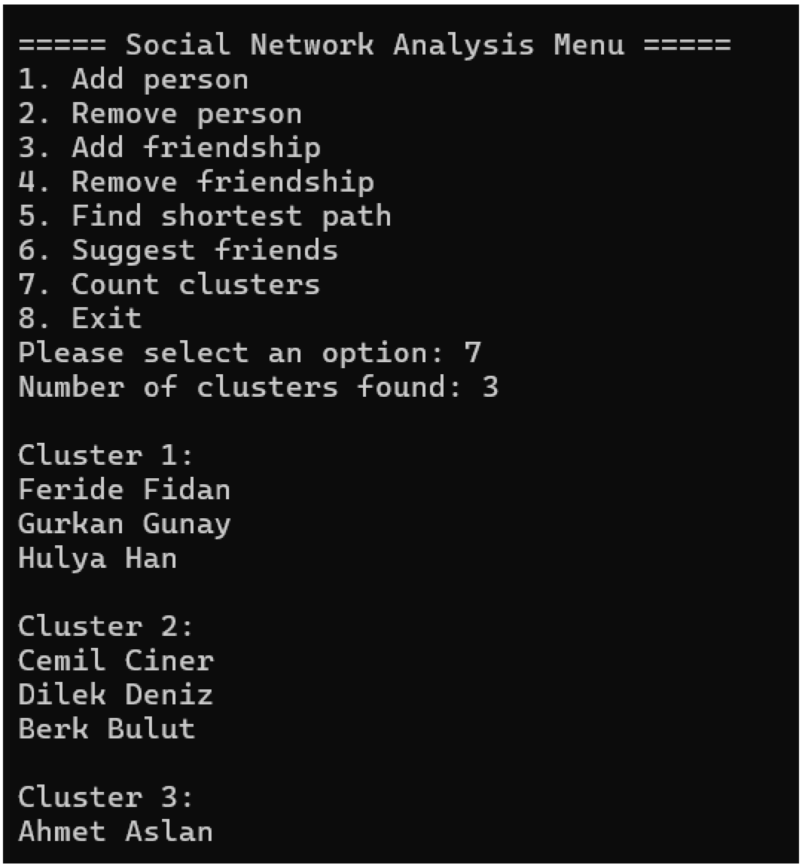
```
===== Social Network Analysis Menu =====
1. Add person
2. Remove person
3. Add friendship
4. Remove friendship
5. Find shortest path
6. Suggest friends
7. Count clusters
8. Exit
Please select an option: 7
Number of clusters found: 2

Cluster 1:
Feride Fidan
Gurkan Gunay
Hulya Han

Cluster 2:
Cemil Ciner
Dilek Deniz
Berk Bulut
Ahmet Aslan
```

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: Ahmet Aslan
Enter first person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:15
Enter second person's name: Berk Bulut
Enter second person's timestamp (yyyy-MM-dd HH:mm:ss): 2024-05-28 12:40:31
Friendship removed between Ahmet Aslan and Berk Bulut
```

Clusters



```
===== 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 found: 3

Cluster 1:
Feride Fidan
Gurkan Gunay
Hulya Han

Cluster 2:
Cemil Ciner
Dilek Deniz
Berk Bulut

Cluster 3:
Ahmet Aslan
```