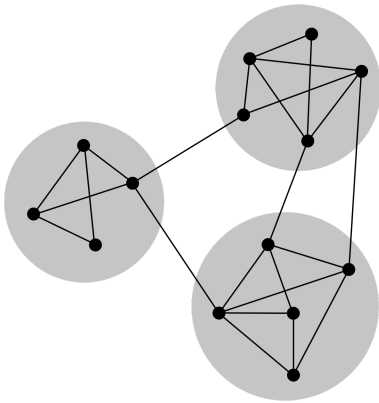# The Sociopath

A. Introduction

Admit that, online learning is boring. No face-to-face, no interaction, no campus life! You don't even know if one is reliable when working on assignments together. To run away from divers and build up your personal reputation, build a friendship monitoring system.

B. Problem Statement

Generate a group of students (10 of them) each with **reputation** (rep), **diving rate** (dive), **lunch starting time** (lunchStart), **lunch period** (lunchPeriod), and **friends** (friends).

Initialization
Initialize these variables for each student as below:
1 <= rep <= 10, 0 < Diving rate (%) < 100, 1100 <= lunch time <= 1400, 5 < lunch period(minutes) < 60, a list of friends as shown in Figure 1 below with their reputation relative to this person.
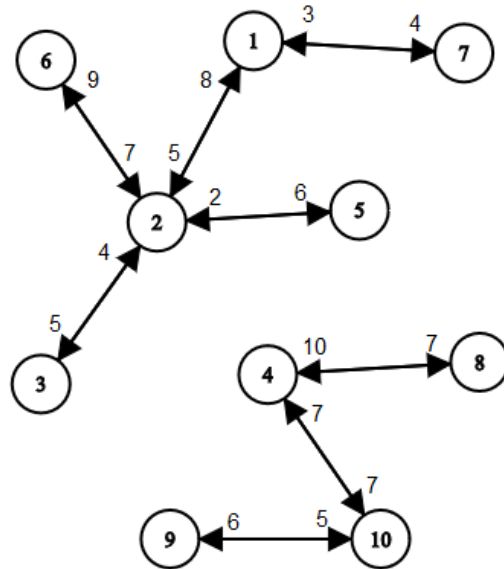
Figure 1: The reputation point relative to other people
Explanation: 1 has 3 rep points relative to 7, meanwhile 7 has 4 rep points relative to 1.

Events
Create multiple events to manipulate the data in your system.

Event 1 - Teaching a stranger to solve lab questions
Here's a stranger who is seeking your help to teach him/her how to solve the Data Structure course's lab question. As a kind person, you will always help them out.

If you did something good to a person you increase your rep points relative to that person by 10. The stranger becomes your friend now. He or she might tell his/her friends about you.
But if you are bad at programming, the learning experience with you might not be pleasant, you will still be friends with him, but your rep points relative to that person will be 2 instead (Note that you were strangers before this, now you guys are friends and your rep point relative to that person is 2).
But how do the messages about you spread or go viral?

Event 2 – Chit-chat
For sure your new friend will have a chit-chatting session with his or her friends. If it's a good message, this increases your rep points relative to them by your rep points relative to your new

friend * 0.5. Otherwise, if it's a bad message, this person will share the same negative rep points that your new friend owns about you. Later, your new friend's friends might tell their friends, so it will multiply and propagate your rep.

For example, A has a rep points of 10 relative to B, B and C chit-chatting and speak about A's good side. So, A rep point will be +5 (50% of A rep points to B) relative to C.
Meanwhile, if B and C are talking about A's bad side, A rep point will be -10 (100% of A rep point to B) relative to C.

Event 3 - Your road to glory

"The deepest principle in human nature is the craving to be appreciated."
-William James

Who doesn't want to feel respected? As a university student, you are ready to build up your reputation. One of the cores of human relationships is food. People like food, and if they usually see you when there is food, they are more likely to like you. If you want to build a good relationship with someone, try to have lunch with them. But you want to be efficient. You don't want to befriend any tom, dick, or harry. You want to maximize the amount of reputation you can gain by befriending people with high reliability.

To do this, you will keep track of 2 observations for each person you are interested in, the time that they will have lunch, and the duration of their meal to find out the average time of their lunch hour and the average time it takes them to finish their lunch. You will then use these statistics to calculate an interval for each person (i.e., the start and end times). Use these to find the maximum reputation you can obtain in any day given that you can only have lunch with one person at one time. Each person that had lunch with you will increase your rep with them by 1 point.

Event 4 - Arranging books
To beat the boredom, you joined a volunteering program to help UM library to arrange the book. The librarian has a special request: make sure the height of a row of books is in non-increasing order.

Example of non-increasing order:
1) 4, 4, 3, 3, 2
2) 4, 3, 2, 1, 0

There is only one row of books on the bookshelf. In this row of books, every book has different heights.

For each round, you'll move from left to right to pick out the book that did not meet the request. At the end of each round, you'll be on the right side of the bookshelf. You put the book(s) down because it's heavy to carry.

Then the next round starts, you start over again moving from left to right.

For each round
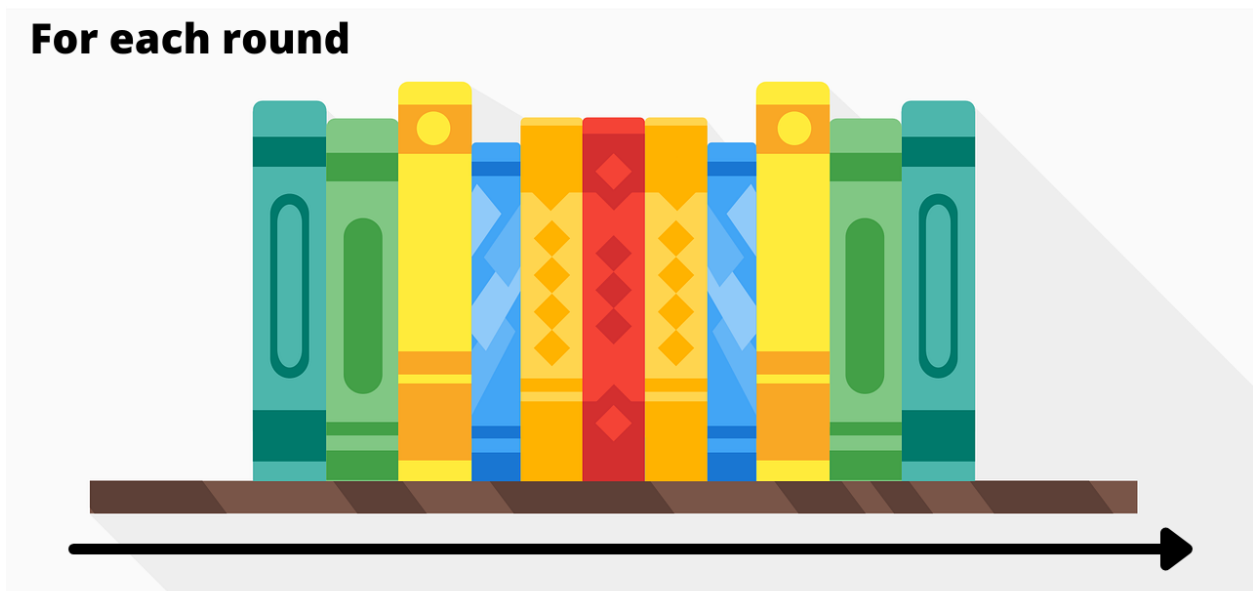
Since the books are heavy, you plan to use a special way to meet the librarian's request.
While you're moving from left to right, you will just check if a book is higher than the book on the left and take it out.

Note that if book A = 15cm, book B = 16cm, book C = 17cm and they're arranged in the order ABC, both book B and C will be taken out, and only book A left for the next round of picking.

For example, height = [3, 1, 10, 7, 3, 5, 6, 6]
After first round = [3, 1, 7, 3, 6]
After second round = [3, 1, 3]
After third round = [3, 1]

Hence, you will need 3 rounds to meet the librarian's request.

You are given the heights of the books. Determine the number of rounds needed to make the height of the books in non-increasing order.
Make your program accept the input and print the number of rounds needed for you to make the row of books in non-increasing order.

Used below string as input

```
7
6 5 8 4 7 10 9
```

Output

```
2
```


Figure 2: Sample Input and Output

Event 5 – meet your crush

While you join the volunteering program, there's this guy/gal. He/she is really cute and talented. You haven't really spoken to him/her, but the minute you saw him/her you got a crush on him/her. You can't stop thinking about your crush.

Sadly, there will be rumors out of nowhere from one of the strangers. By the nature of chit-chatting, you know it will be a disaster if your crush knows about the false allegation.

So, who do you have to convince to clarify the rumor about you to prevent it from getting to your crush? The rumor will start in the stranger's cluster and your crush is in another cluster. You might identify someone connected between these 2 clusters. Then, you can convince him/her on the rumor so that your crush won't hear the rumor. If there is more than one person that needs to be convinced to prevent the rumor from spreading, then you can't do it because you can only convince one person per day.

Calculate that the rumor propagates at one jump per day and identify if you can convince enough people to prevent your crush from knowing about the rumor.

You must refer to the following friendship list to determine who's connected to who.
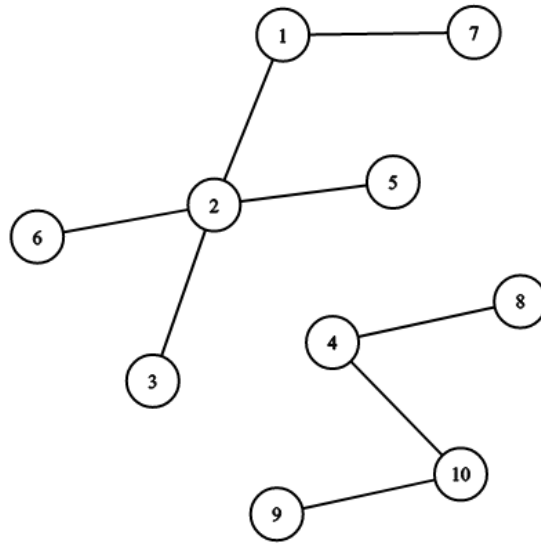

Figure 3: The two clusters and the connection.

Table 1: The person ID and list of friends.

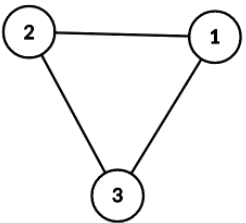| Person	List of Friends
1	2, 7
2	1, 3, 5, 6
3	2
4	8, 10
5	2
6	2
7	1
8	4
9	10
10	4, 9

Event 6 - Friendship
University is not only for you to get a degree but is also a place for you to meet new people, and some of the friendships even last a lifetime! There are many ways for us to meet new friends, one of the ways is we will make new friends out of our friends' friends through some
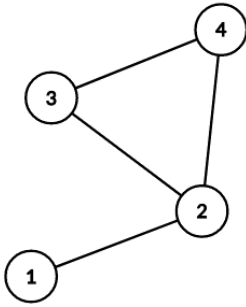
opportunities. For example, A and B are friends and B and C are friends, A will eventually meet C! (Note that friendship is undirected which means A - B is equal to B - A, A-B-C is equal to C-B-A).

You are given a list of an integer n, followed by n lines of existing friendships between 2 individuals. You need to find the total number of unique ways the friendship can be formed. It is guaranteed that the relationship between all individuals are connected (all individuals will eventually meet each other).

First example:

Input: 3 1 2 2 3 1 3	Output: 6 Explanation: You can form the following friendship(s): 1. [1, 2] 2. [1, 3] 3. [2, 3] 4. [1, 2, 3] 5. [1, 3, 2] 6. [2, 1, 3]

Second example:

Input: 4 1 2 2 3 3 4 4 2	Output: 11 Explanation: You can form the following friendship(s): 1. [1, 2] 2. [2, 3] 3. [3, 4] 4. [2, 4]

```
5.  [1, 2, 3]
6.  [1, 2, 4]
7.  [2, 3, 4]
8.  [2, 4, 3]
9.  [3, 2, 4]
10. [1, 2, 3, 4]
11. [1, 2, 4, 3]
```

C. Additional Challenge

1. **Graphical User Interface (GUI)**
   GUI would be fun too, you can learn simple HTML, CSS and JS for displaying the data of the system, creating a dashboard of the system and controlling it.

2. **Social dynamics**
   Let's be real. Not everyone's your friend. Social networks are complicated and having simple edges representing 2 friends isn't enough modelling power for you to conquer the world. Implement different kinds of edges like enemies and frenemies with their own unique dynamics and interactions. Explain the motivations for your implementation and give us a glimpse of your sociopathic mind!

3. **Matchmaker**
   Ah what a good time to be alive, my friend groups are all healthy and going well. Would be a shame if it was ruined over love. (fake) Research shows that 89.3% of friend groups split because of romantic rivalry. This would be terrible for you and you are going to do everything within your power to prevent a rivalry from dividing your friend group. In order to do this, you are going to have to play matchmaker. Implement an algorithm that calculates a stable matching for your friend group so that you can prevent rivalries from developing by matching your friends with a stable partner.
   Context on stable matching and the mating algorithm:
   https://www.youtube.com/watch?v=5RSMLgy06Ew

4. **Parallel farming**
   You've gained a bit of influence and now have the power to invite people to eat lunch at your table. So why eat with 1 person when the table is big enough for 3? Upgrade your system in Event 3 to be able to calculate the maximum reputation you can obtain given that you can eat with 3 people at any one time.

5. **Life's not a game that you play to get even**
   Implement functionality to ingest data from social media. Using the sociopath app has greatly improved your ranking in the social ladder. But it's so tedious! So many relationships to keep track off, bits and pieces of information that all need to be manually keyed. What a hassle. Fortunately, people nowadays post all their social information on mysterious sites called social media. Wouldn't it be great if you could dynamically scrape

the data (whether A is a friend of B and if they comment on each other's posts) from these social media sites to get the latest insights into the social atmosphere?

6. **Herd Immunity**

   All this sociopath work has you feeling a little selfish and a bit of an egomaniac. Maybe you could give back by contributing your knowledge to a good cause. Vaccines are scarce in the beginning of a post COVID world and it would be great if they go to people who would make the most impact in stopping it from spreading among clusters of friends. Analysis of the theory of six degrees of separation has found that it is usually achieved by a few people who link multiple friend circles. These people are perfect bridges for the spread of the virus and would be ideal vaccination targets for stopping the spread of the virus. Create an algorithm that takes the number of vaccines as input and identifies who should be vaccinated such that the maximum size of any friend cluster is minimal.

7. **Six degrees to Ken Thompson**

   Speaking of six degrees of separation, one of the demos is a huge fan of Ken Thompson. For those of you who are unaware, Ken Thompson is one of the creators of UNIX and has worked on many influential programming languages like C, C++ and Go. Use the sociopath app to calculate a path of correspondence between you and Ken Thompson that is less than or equal to 6 hops. Present your methodology on identifying and validating this path.

D. Tips and comments

1. Use git to do version control
2. Learn graph algorithms
3. Use graph databases

E. Questions

If you have any questions regarding the question, you can email
helpdeskdatastructure@gmail.com with the title (Sociopath).