Date	November 20
Time	Submit by 10 PM
Assignment Code	CS1340_A4P2
Total Marks	20
Weightage	10%

Submission Instructions

- The assignment has two parts. You must attempt exactly one (A4P2-A or B)
- A4P2-A carries a bonus grade of 5%
- A4P2-B carries a bonus grade of 7%
- **You must upload a compressed folder (containing client and server programs) in the classroom**
- Name your file as follows: yourname_CS1340_A4P2-A/B.zip

A4P2-A

► Mandatory Part ———————————   [Marks = 20]

- Course instructors keep (mostly) students grades in an excel sheet. Every time students ask for their grades (client side), the instructor indexes the excel sheet and returns the grade (server side). Write a TCP-based client-server system to implement the above functionality. An instructor can run the server-side to implement a course-grade directory. A student can run the client-side program to remotely connect to this directory and fetch their latest scores.

- Instantiate your server with the records of at least three students; each record consists of marks ([max marks, marks obtained]) in three assignments A1, A2, A3 and midterm exam.

- A connected client must send a student's name to fetch the record.

► Optional Part (Bonus grade) ———————————   [Marks = 10 = 5 (2.5%) + 5 (2.5%)]

- Implement a first-time registration and authentication functionality into this system.

    - During first time registration, only students from the course are allowed to register to this system using their ashoka id credentials. After the registration, passwords will be issued to the users.   [5]
    - A valid password is required to access the directory.   [5]

A4P2-B

[Marks = 30 = 20 (weightage - 10%) + Bonus 10 (weightage - 7%)]

For this assignment you will write a TCP client-server system. The server offers the service of running a remote voting. The exact interaction is given below.

Execution Flow

| The server application is running at a host with ip address a.b.c.d and port number 5545 in passive mode listening for a transmission from the client.

→ A client makes a TCP connection with the the server (at a.b.c.d and port 5545).

← Once the connection is accepted by the server, the client is presented with the following information:

  - Welcome! You can participant in the vote by presenting your password. Reply with a "1" if you want to participate now; with a "2" if you want to see the results; and with "3" other wise.

$\rightarrow$ The client replies with a "1/2/3"

- Case1: if the reply is "3", the server disconnects and close the connection [2]
- Case2: if the reply is "1", and the password presented is correct (A first time registration must be implemented using their ashoka-id credentials. After the registration, a unique password is provided to the user who must later present it (Case2) to participate in the voting), the server responds with a multiple choices to choose from. [3 + 5 = 8]
- Case3: if the reply is "2", the password presented for seeing results is correct, and the voting is over, the server responds with the following information: "the number of responses each candidate has received". In the case when either password is incorrect or voting not yet closed, the server responses with an error and disconnects. [8]

$\rightarrow$ In Case2, the client participates in the voting and responds with its chosen answer.

$\leftarrow$ The server records the vote and responds with the following message: "Thank you for participating. Your response is registered against your IP address - "provide here the client's IP address". It also presents to the client a password which it can use it at a later point (when the voting is over) to see results. It then disconnects. [7]

- The program must additionally satisfy the following requirement: If a client tries to participate twice using the same machine (IP address), it must not be allowed. [5]