Distributed Systems
Assignment 1
Deadline : 19th day of August, 2018 (Sunday)(Time - 20:55)

Problem Statement:
Implement a client server model for communication amongst multiple connected clients.

Generate **client/server chat room** with sockets.
The server should support up to N clients at one time, which would be given during the start of server.

The client's behavior will be defined as the following:
Client will be prompted to associate itself with a username like USER1.
Once the connection is formed, client can list the active chat rooms / users.
Client can ask to create a new chat room or can join an existing chat room.

command to create a chat room.

```
>> create chatroom chatRoom1
```

command to list all chat rooms.

```
>> list chatrooms
```

command to join existing chat room

```
>> join chatRoom1
```

command to leave existing chat room

```
>> leave
```

The server should then print a message that the client(identified by the username he gave) has left the room.
command to list all users in that chat room.

```
>> list users
```

command to add another user to chat room

```
>> add user2
```

If client is not associated to any chat room or user2 doesn't exist or is associated with another chatroom then it should return error

Do the apt error handling.

No client can be associated with more that one chat room at a time. Creating a chat room automatically joins the client to that chat room.

The chatroom behavior is defined as follows.

If any user in a particular chat room types a message and hits return key, that message should be sent to
the server and broadcasted to all other clients of that particular chat room.

command to send message

```
User1                                     User2 (same chatroom)
>> reply "message content"                >>
                                          User1: message content
>> reply A.txt tcp                        >>
Sending A.txt                             Receiving A.txt from User1
Sent file                                 Received A.txt from User1
>> reply B.txt udp                        >>
Sending B.txt                             Receiving B.txt from User1
Sent file                                 Received B.txt from User1
```

If no user is associated with a particular chat room then that chat room is destroyed.

The following requirements should meet.
1. Should be able to send messages (**Half-Duplex** communication, using TCP sockets).
2. Should be able to send **any kind** of file to each other using both Datagram socket based communication (UDP socket) and Stream sockets based communication (TCP socket). File will be received in the present working directory of client. (**Half-Duplex**).

Upload format:

1. You can implement it in either C, C++ or Java.
2. Readable Code : Nicely structured code, with coding standards followed
3. README : Should include the running directions and the features implemented.
4. Format: Include server file (server.sh) and client file (client.sh). Executable with commands **bash server.sh <client_limit>** and **bash client.sh <client_name>**
5. Zip : Include all the files (<RollNumber>_Assignment_1.zip)

PS : Plagiarism and copy cases will be awarded a 0 in this assignment.