Final Project

Consider a group chat program with one server and multiple clients. The server may receive multiple client requests as the same time and run each client request in parallel. Following this approach, each user is assigned a unique thread. The messages are exchanged in the server using shared memory. The communication protocol is TCP/IP. Each client signs in with its username. The server and clients may have different IP addresses.

1) Initially, the server (e.g. at 131.114.218.213) shall be launched with the portID as argument. When the post is unavailable, the program shall exit gracefully.

Shell (server)

```
bash~$ ./server
usage: ./server portID
bash~$ ./server 8080
```

Then each client shall be launched with host-ID, username and port-ID as its arguments. For example, Giovanni will enter the chat with

Shell (client Giovanni)

```
bash~$ ./client
usage client hostID myname portID
bash~$ ./client 131.114.218.213 Giovanni 8080
```

The program may not crash if the connection fails. When the connection is established, the server detecting the username and the IP address of the client, writes to `stdout`

Shell (server)

```
131.114.218.212:Giovanni connected
```

and the client prints out the message

Shell (client Giovanni)

```
connected.
```

Hint: use mutex to ensure that IP address and username arrive atomically.

2) Several clients shall sign-up to the chat-room. A client may contact another client with the keyword "**@<To>:<message>**". The received message at a target client appears as "**> <From>: <message>**". Only the target client shall receive the dedicated message. The program may not crash when the partner does not exist or a syntax error has occurred. Suppose there are

three clients Giovanni, Maria, and Marco online and Giovanni wants to contact Marco.

Shell (client Giovanni)

```
@Marco: Hello! How are you?
```

Shell (client Marco)

```
>Giovanni: Hello! How are you?
@Giovanni: I feel great.
@Maria: Are you around?
```

Shell (client Giovanni)

```
@Marco: Hello! How are you?
>Marco: I feel great.
```

Shell (client Maria)

```
>Marco: Are you around?
```