CS3093D Networks Laboratory - Assignment 4

Anandu B Ajith - B180288CS

20 February 2022

Question 1

Instructions

- Code can be compiled using "make" and binaries will be placed in "./bin".

- Server code is in "1/src/server.c" and Client code is in "1/src/client.c",
  and common headers in "1/src/common.h", and common function in "1/src/common.c"

- Client takes nickname as input, and connects to the server

- If the nickname is already in use, the server will send message and client will disconnect.

- Client messages are sent to server, and server broadcasts it to other connected clients.

- Server uses "poll()" to keep track of which sockets have data to read

Screenshots

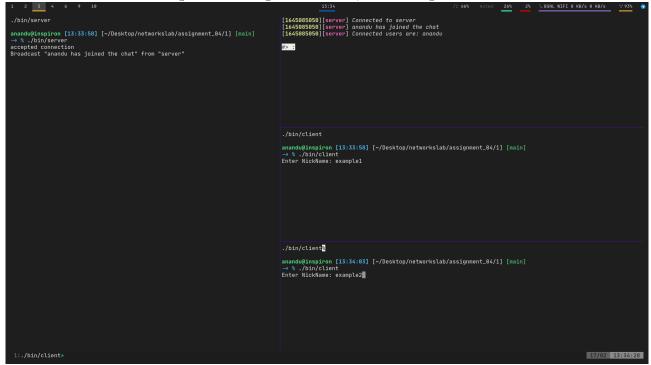Figure 1: Starting the server left, and starting 3 clients

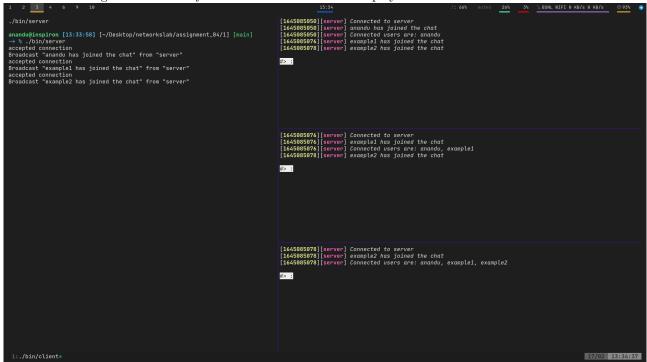Figure 2: Currently connected usernames are displayed when new client connects

```
1   2   3   4   6   9   10                                    13:34        /: 66%   muted   26%   3%   ⚲ BSNL WIFI 0 KB/s 0 KB/s   ⏱ 93%

./bin/server                                                  [1645085050][server] Connected to server
                                                              [1645085050][server] anandu has joined the chat
anandu@inspiron [13:33:58] [~/Desktop/networkslab/assignment_04/1] [main]  [1645085050][server] Connected users are: anandu
→ % ./bin/server                                              [1645085076][server] example1 has joined the chat
accepted connection                                           [1645085078][server] example2 has joined the chat
Broadcast "anandu has joined the chat" from "server"
accepted connection                                           #> :
Broadcast "example1 has joined the chat" from "server"
accepted connection
Broadcast "example2 has joined the chat" from "server"


                                                              [1645085076][server] Connected to server
                                                              [1645085076][server] example1 has joined the chat
                                                              [1645085076][server] Connected users are: anandu, example1
                                                              [1645085078][server] example2 has joined the chat

                                                              #> :


                                                              [1645085078][server] Connected to server
                                                              [1645085078][server] example2 has joined the chat
                                                              [1645085078][server] Connected users are: anandu, example1, example2

                                                              #> :

1:./bin/client*                                                                                           17/02  13:34:37
```

Figure 3: Messages being broadcast to all connected clients

```
1   2   3   4   6   9   10                                    13:34        /: 66%   muted   26%   3%   ⚲ BSNL WIFI 0 KB/s 0 KB/s   ⏱ 93%

./bin/server                                                  [1645085050][server] Connected to server
                                                              [1645085050][server] anandu has joined the chat
anandu@inspiron [13:33:58] [~/Desktop/networkslab/assignment_04/1] [main]  [1645085050][server] Connected users are: anandu
→ % ./bin/server                                              [1645085076][server] example1 has joined the chat
accepted connection                                           [1645085078][server] example2 has joined the chat
Broadcast "anandu has joined the chat" from "server"          [1645085089][anandu] hi there
accepted connection                                           [1645085092][example1] hello anandu
Broadcast "example1 has joined the chat" from "server"        [1645085095][example2] hello again
accepted connection
Broadcast "example2 has joined the chat" from "server"        #> :
Broadcast "hi there " from "anandu"
Broadcast "hello anandu" from "example1"
Broadcast "hello again" from "example2"


                                                              [1645085076][server] Connected to server
                                                              [1645085076][server] example1 has joined the chat
                                                              [1645085076][server] Connected users are: anandu, example1
                                                              [1645085078][server] example2 has joined the chat
                                                              [1645085089][anandu] hi there
                                                              [1645085092][example1] hello anandu
                                                              [1645085095][example2] hello again

                                                              #> :


                                                              [1645085078][server] Connected to server
                                                              [1645085078][server] example2 has joined the chat
                                                              [1645085078][server] Connected users are: anandu, example1, example2
                                                              [1645085089][anandu] hi there
                                                              [1645085092][example1] hello anandu
                                                              [1645085095][example2] hello again

                                                              #> :

1:./bin/client*                                                                                           17/02  13:34:58
```
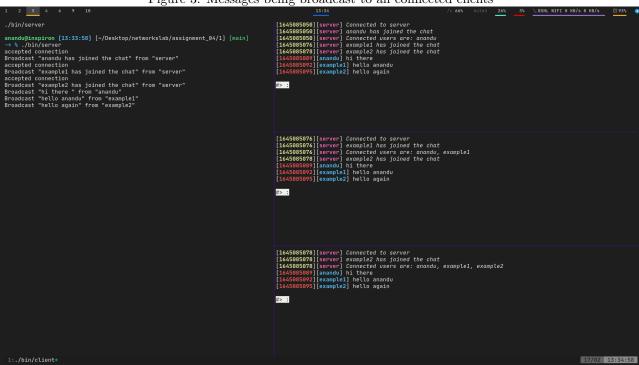
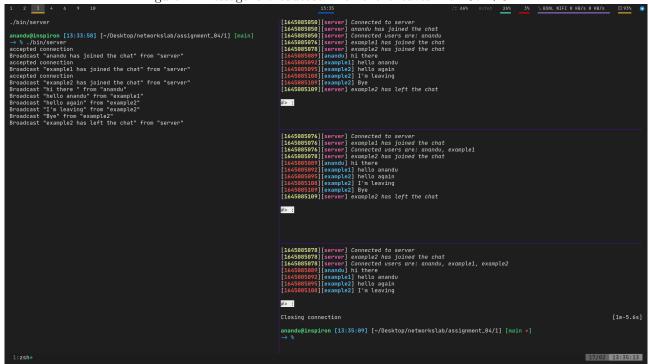Figure 4: Message is broadcast when client leaves with Ctrl+C

Figure 5: Server disconnects when trying to join with already existing username

Question 2

Instruction

- Code can be compiled using "make" and binaries will be placed in "./bin".

- Server code is in "2/src/server.c" and Client code is in "2/src/client.c", and common headers in "common.h".

- Server will "fork()" itself when it receives a new connection from client.

- Client can send queries, server will respond with result, Ctrl-C can be used to exit client.

Screenshots