

Members: Eric Remington Davey, Hanna Rumpler

Date: 11/5/2019

CS - 436 Networking

Professor: Yanyan Li

## Chatting Application Project

**Introduction:** The requirements for this lab were to implement a chatting system app in python that utilized the TCP socket library. The project specified that there were two programs, one was a client program that was for the user to talk to others, and one was a server to accept connections from multiple client programs so that they could talk to one another. The server also lists who is currently connected, and what their username and availability was.

**Implementation:** We implemented the server side without using the `_pThread` library for python. The main way we went about this was to place each socket into a python dictionary object, with all pertaining information to that socket user from the client (i.e. 'username', 'available' (whether the person is chatting with someone or not), 'chatting\_with' (the name OF the person chatting) and finally the socket information (I.P. address of connecting client, port, ect.)) and to iterate through the set of dictionaries inside of a while loop constantly iteratively checking for an incoming message (`socket.recv()`) from each client connection and then handling the subsequent request depending on the content of the message.

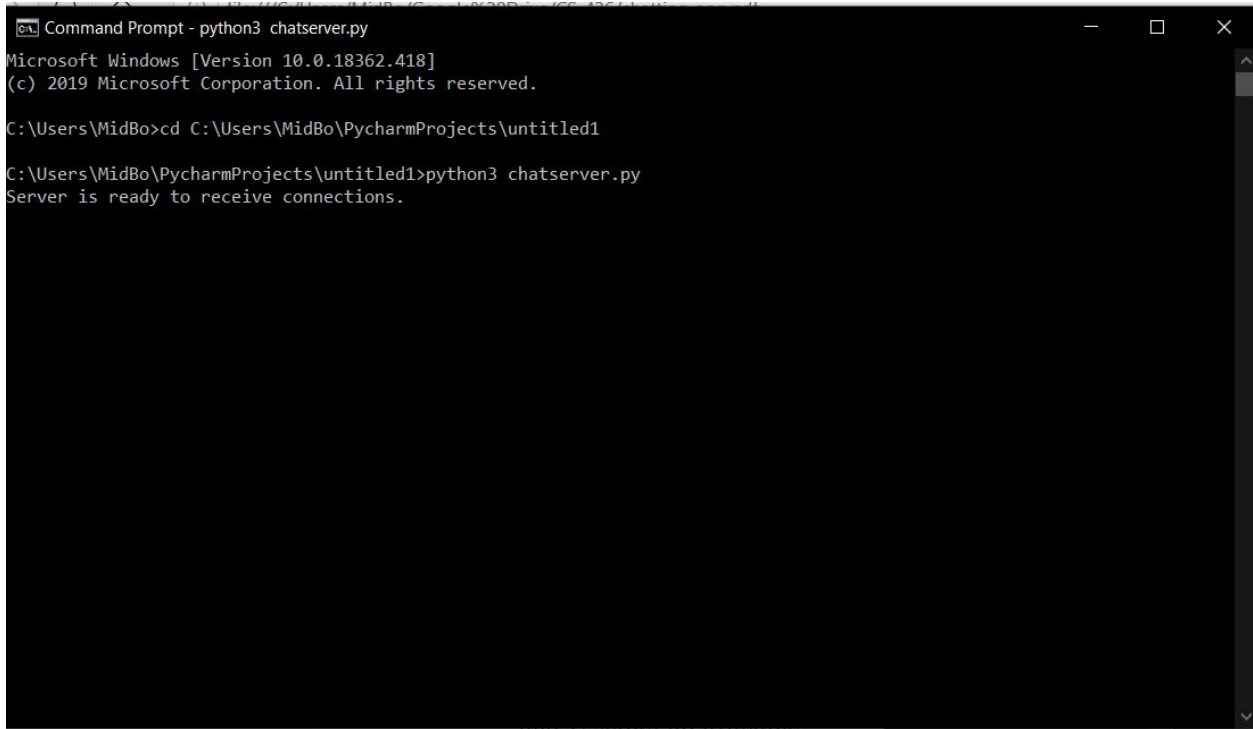
**Results:** The finished lab code resulted in a working chat application that could allow two client connections under two different usernames to connect and chat with one another and for other clients under other usernames to connect and see that A and B are chatting and even connect to other users D and E if they want to.

**Conclusion:** This project was a lot more work than we expected, and we had to rewrite it a few times. At one point, we completely scrapped what we had and decided to look through the example code provided to us one final time. Our final code managed to implement the code without using multithreading or `_Pthread` library. When we got stuck, we reached out to you in class and searched the internet for examples of similar projects. Some code we borrowed from a tutorial we found ([https://www.youtube.com/watch?v=CV7\\_stUWvBQ](https://www.youtube.com/watch?v=CV7_stUWvBQ)). The only problem with the implementation of this chat app was that there isn't an easy way to interrupt the `input()` command from the keyboard without using `_pthread` (in the case of receiving a request to chat from another user while being prompted to enter input into the keyboard terminal). Overall, this project has taught us a lot about the python programming language and about sockets.

screenshots:

Below you will find screenshots of the chat app working.

1. As shown below, server executes, waits for connections

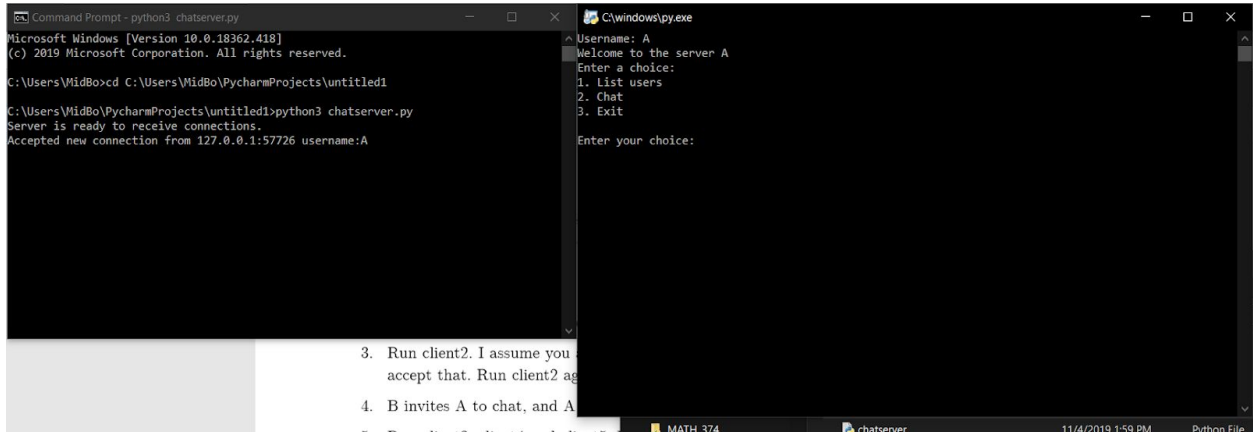


```
Command Prompt - python3 chatserver.py
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\MidBo>cd C:\Users\MidBo\PycharmProjects\untitled1

C:\Users\MidBo\PycharmProjects\untitled1>python3 chatserver.py
Server is ready to receive connections.
```

2. Second window is first client enters username A and server accepts.



```
Command Prompt - python3 chatserver.py
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\MidBo>cd C:\Users\MidBo\PycharmProjects\untitled1

C:\Users\MidBo\PycharmProjects\untitled1>python3 chatserver.py
Server is ready to receive connections.
Accepted new connection from 127.0.0.1:57726 username:A

C:\windows\py.exe
Username: A
Welcome to the server A
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
```

3. Third window is also client, enters A but A is rejected, enters B and is accepted.

```

Command Prompt - python3 chatserver.py
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\WidBo>cd C:\Users\WidBo\PycharmProjects\untitled1
C:\Users\WidBo\PycharmProjects\untitled1>python3 chatserver.py
Server is ready to receive connections.
Accepted new connection from 127.0.0.1:57726 username:A
Rejecting Username, already being used
Accepted new connection from 127.0.0.1:57746 username:B

C:\windows\py.exe
Username: A
Welcome to the server A
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: A
Username already in use
Username: B
Welcome to the server B
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
  
```

4. Third window client B enters 2, requests to chat with A, A enters yes and accepts request to chat, server connects both users and they send two messages ( you have to press enter in order to see the next message from server socket)

```

Command Prompt - python3 chatserver.py
Microsoft Windows [Version 10.0.18362.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\WidBo>cd C:\Users\WidBo\PycharmProjects\untitled1
C:\Users\WidBo\PycharmProjects\untitled1>python3 chatserver.py
Server is ready to receive connections.
Accepted new connection from 127.0.0.1:57726 username:A
Rejecting Username, already being used
Accepted new connection from 127.0.0.1:57746 username:B
Received message from B: 2
Received message from B: A
Asking A if they want to chat with B
Received message from A: yes
connecting A with B
Received message from B: Hello mr A
Received message from A: Hello mr B
Received message from B: How are you?
Received message from A: I'm fine

C:\windows\py.exe
Username: A
Welcome to the server A
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: A
Username already in use
Username: B
Welcome to the server B
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

Server: B would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
A > Hello mr B
B: Hello mr A
A >
B: How are you?
A > I'm fine
A >
  
```

file and attach the screenshots

1. Run the server.
2. Run client1. I assume you
3. Run client2. I assume you
4. B invites A to chat, and A, available,
5. Run client3, client4 and c
6. C lists the users. Make su
7. C wants to chat. Make su

## 5. new clients (on the bottom) C, D, E all connect

```
Command Prompt - python3 chatserver.py
C:\Users\WidBo\PycharmProjects\untitled1>python3 chatserver.py
Server is ready to receive connections.
Accepted new connection from 127.0.0.1:57726 username:A
Rejecting Username, already being used
Accepted new connection from 127.0.0.1:57746 username:B
Received message from B: 2
Received message from B: A
Asking A if they want to chat with B
Received message from A: yes
connecting A with B
Received message from B: Hello mr A
Received message from A: Hello mr B
Received message from B: How are you?
Received message from A: I'm fine
Accepted new connection from 127.0.0.1:57776 username:C
Accepted new connection from 127.0.0.1:57777 username:D
Accepted new connection from 127.0.0.1:57778 username:E

C:\windows\py.exe
Username: A
Welcome to the server A
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: B would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
A > Hello mr B
B: Hello mr A
A >
B: How are you?
A > I'm fine
A >

C:\windows\py.exe
Username: B
Welcome to the server B
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: A would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
B > Hello mr A
A: Hello mr B
B > How are you?
A: I'm fine
B >

C:\windows\py.exe
Username: C
Welcome to the server C
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: D
Welcome to the server D
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: E
Welcome to the server E
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
```

## 6. C requests a list of users, server shows that A and B and unavailable and chatting with one another.

```
Command Prompt - python3 chatserver.py
Server is ready to receive connections.
Accepted new connection from 127.0.0.1:57726 username:A
Rejecting Username, already being used
Accepted new connection from 127.0.0.1:57746 username:B
Received message from B: 2
Received message from B: A
Asking A if they want to chat with B
Received message from A: yes
connecting A with B
Received message from B: Hello mr A
Received message from A: Hello mr B
Received message from B: How are you?
Received message from A: I'm fine
Accepted new connection from 127.0.0.1:57776 username:C
Accepted new connection from 127.0.0.1:57777 username:D
Accepted new connection from 127.0.0.1:57778 username:E
Received message from C: 1

C:\windows\py.exe
Username: A
Welcome to the server A
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: B would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
A > Hello mr B
B: Hello mr A
A >
B: How are you?
A > I'm fine
A >

C:\windows\py.exe
Username: B
Welcome to the server B
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: A would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
B > Hello mr A
A: Hello mr B
B > How are you?
A: I'm fine
B >

C:\windows\py.exe
Username: C
Welcome to the server C
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice: 1
Server:
A, Unavailable, Chatting with B
B, Unavailable, Chatting with A
C, available,
D, available,
E, available,
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: D
Welcome to the server D
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: E
Welcome to the server E
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
```

## 7. C requests to chat with E, E rejects request, both clients return to menu

```
Command Prompt - python3 chatserv.py
Received message from B: 2
Received message from B: A
Asking A if they want to chat with B
Received message from A: yes
Connecting A with B
Received message from B: Hello mr A
Received message from A: Hello mr B
Received message from B: How are you?
Received message from A: I'm fine
Accepted new connection from 127.0.0.1:57776 username:C
Accepted new connection from 127.0.0.1:57777 username:D
Accepted new connection from 127.0.0.1:57778 username:E
Received message from C: 1
Received message from C: 2
Received message from C: E
Asking E if they want to chat with C
Received message from E: no

C:\windows\py.exe
Username: A
Welcome to the server A
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: B would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
A > Hello mr B
B: Hello mr A
A >
B: How are you?
A > I'm fine
A >

C:\windows\py.exe
A, available,
B, available,
Who: A
Server: Awaiting response from A
Server: You are now chatting
B > Hello mr A
B >
A: Hello mr B
B > How are you?
B >
A: I'm fine
B >

C:\windows\py.exe
Enter your choice: 2
Server: Please select user to chat with:
Server:
A, Unavailable, Chatting with B
B, Unavailable, Chatting with A
C, available,
D, available,
E, available,
Who: E
Server: Awaiting response from E
Server: E, refused your request to chat
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: D
Welcome to the server D
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Username: E
Welcome to the server E
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: C would like to chat, accept? yes/no
yes/no: no
no no no
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
```

## 8. C requests to chat with D, they both exchange messages.

```
Command Prompt - python3 chatserv.py
Accepted new connection from 127.0.0.1:57776 username:C
Accepted new connection from 127.0.0.1:57777 username:D
Accepted new connection from 127.0.0.1:57778 username:E
Received message from C: 1
Received message from C: 2
Received message from C: E
Asking E if they want to chat with C
Received message from E: no
Received message from C: 2
Received message from C: D
Asking D if they want to chat with C
Received message from D: yes
Connecting D with C
Received message from D: Hello mr C
Received message from C: Hello mr D
Received message from D: How are you?
Received message from C: I'm not doing so well

C:\windows\py.exe
Enter your choice: 2
Server: Please select user to chat with:
Server:
A, Unavailable, Chatting with B
B, Unavailable, Chatting with A
C, available,
D, available,
E, available,
Who: D
Server: Awaiting response from D
Server: You are now chatting
C >
D: Hello mr C
C > Hello mr D
C >
D: How are you?
C > I'm not doing so well
C >

C:\windows\py.exe
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: C would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
D > Hello mr C
C: Hello mr D
D >
C: How are you?
D >
C: I'm not doing so well
D >

C:\windows\py.exe
A, available,
B, available,
Who: A
Server: Awaiting response from A
Server: You are now chatting
B > Hello mr A
B >
A: Hello mr B
B > How are you?
B >
A: I'm fine
B >
B >

C:\windows\py.exe
Username: E
Welcome to the server E
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: C would like to chat, accept? yes/no
yes/no: no
no no no
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
```

9. E requests a list of users who are chatting, sees that A and B are unavailable and chatting with each other, C and D are unavailable and chatting with each other.

```

Command Prompt - python3 chatserver.py
Accepted new connection from 127.0.0.1:57777 username:D
Accepted new connection from 127.0.0.1:57778 username:E
Received message from C: 1
Received message from C: 2
Received message from C: E
Asking E if they want to chat with C
Received message from E: no
Received message from C: 2
Received message from C: D
Asking D if they want to chat with C
Received message from D: yes
connecting D with C
Received message from D: Hello mr C
Received message from C: Hello mr D
Received message from D: How are you?
Received message from C: I'm not doing so well
Received message from E: 1

C:\windows\py.exe
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:
Server: B would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
A > Hello mr B
B: Hello mr A
A >
B: How are you?
A > I'm fine
A >
A >

C:\windows\py.exe
Enter a choice:
1. List users
2. Chat
3. Exit
Who: A
Server: Awaiting response from A
Server: You are now chatting
B > Hello mr A
B >
A: Hello mr B
B > How are you?
B >
A: I'm fine
B >
B >
B >

C:\windows\py.exe
Enter your choice: 2
Server: Please select user to chat with:
Server:
A, Unavailable, Chatting with B
B, Unavailable, Chatting with A
C, available,
D, available,
E, available,
Who: D
Server: Awaiting response from D
Server: You are now chatting
C >
D: Hello mr C
C > Hello mr D
C >
D: How are you?
C > I'm not doing so well
C >

184
185
186
for client_sockets in sockets_list:
    if skip == 0:
        skip = 1

```

10. A and B, C and D all enter quit into chat to quit chatting, return to menu, E enters 3 to exit program, server closes connection from E.

```

Command Prompt - python3 chatserver.py
Received message from C: 2
Received message from C: E
Asking E if they want to chat with C
Received message from E: no
Received message from C: 2
Received message from C: D
Asking D if they want to chat with C
Received message from D: yes
connecting D with C
Received message from D: Hello mr C
Received message from C: Hello mr D
Received message from D: How are you?
Received message from C: I'm not doing so well
Received message from E: 1
Received message from B: quit
Received message from D: quit
Closed connection from E

C:\windows\py.exe
Who: D
Server: Awaiting response from D
Server: You are now chatting
C >
D: Hello mr C
C > Hello mr D
C >
D: How are you?
C > I'm not doing so well
C >
Server: Quitting chat
D: quit
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Enter your choice:
Server: C would like to chat, accept? yes/no
yes/no: yes
Server: You are now chatting
D > Hello mr C
D >
C: Hello mr D
D > How are you?
D >
C: I'm not doing so well
D > quit
Server: Quitting chat
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

C:\windows\py.exe
Enter a choice:
Server: Awaiting response from A
Server: You are now chatting
B > Hello mr A
B >
A: Hello mr B
B > How are you?
B >
A: I'm fine
B >
B >
B > quit
Server: Quitting chat
Enter a choice:
1. List users
2. Chat
3. Exit
Enter your choice:

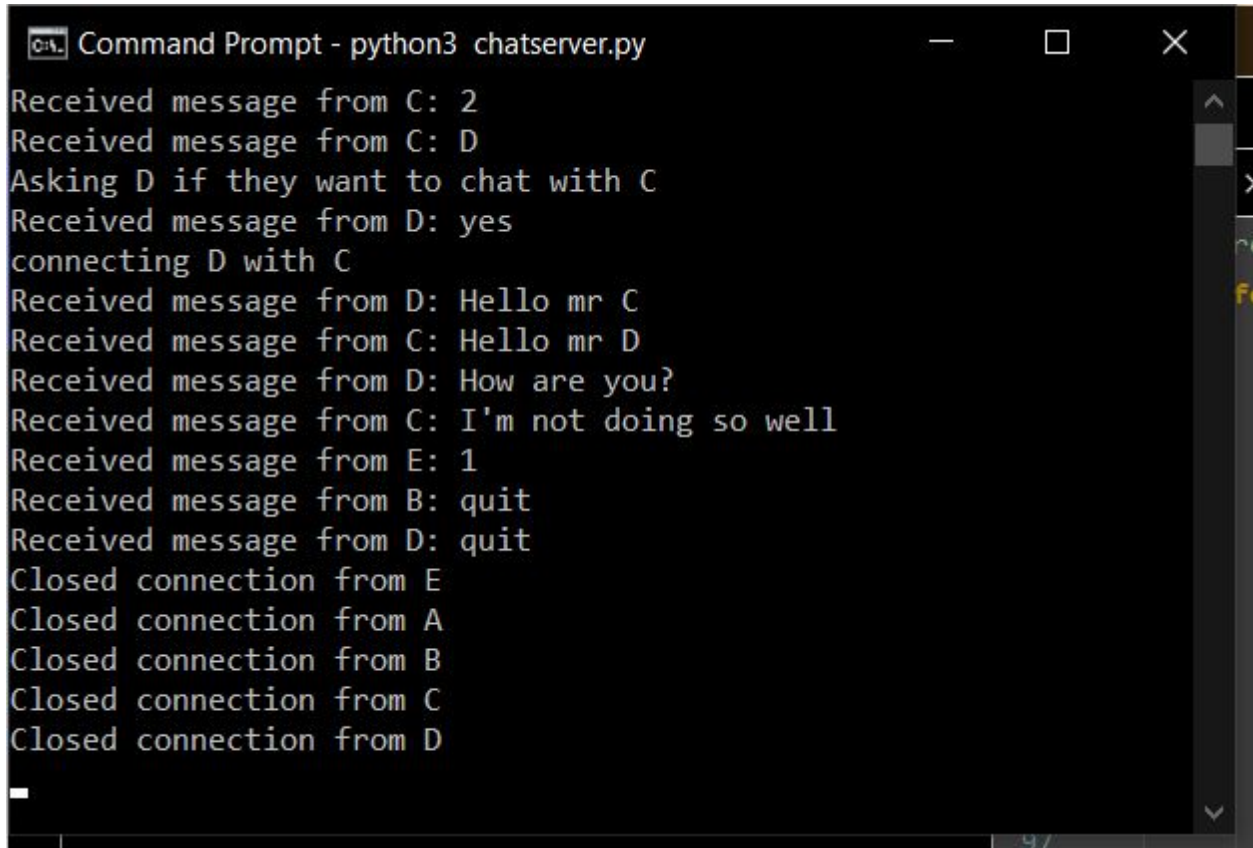
s[1] username={user['data'].decode()}"
code()}"
s[1] = True

```

11.  
No group chat function implemented



12. A, B, C, and D press 3 and exit out, server closes the connection from all clients.



```
Command Prompt - python3 chatserver.py
Received message from C: 2
Received message from C: D
Asking D if they want to chat with C
Received message from D: yes
connecting D with C
Received message from D: Hello mr C
Received message from C: Hello mr D
Received message from D: How are you?
Received message from C: I'm not doing so well
Received message from E: 1
Received message from B: quit
Received message from D: quit
Closed connection from E
Closed connection from A
Closed connection from B
Closed connection from C
Closed connection from D
```