

## Group 1

Hrishikesh Rajesh Menon -COE18B024  
Katepalli Naga Sai Bhargav - COE18B028  
Vamsi Krishna Thigulla - COE18B056  
Srinivasan R Sharma - COE18B065  
Paleti Krishnasai - CED18I039  
Vasupalli Surya Satya Darshan - CED18I056

# Multithreading Client Server Communication

**1: Develop a client and server program, where the server generates a separate thread for each incoming client request. Server shows the client accepted port number and ip address.**

## APPROACH :

- The chat program does not differ in its basic working.
- On the server side, when the server is accepting connections from the clients, it creates a new process, thus allowing multiple clients to connect simultaneously.
  - The messages sent to the server from a client and messages sent to the client from the server are private and not viewable by the other clients.
- On the client side, the client will connect to the server through the respective port and send and receive messages to and from the server.
- The client who has sent the latest message will receive a reply immediately.

The server binds and accepts multiple clients using process/thread creation through **fork()** .

## CODE :

The code is highly modular and reusable ( code provided in the folder along with the demo).

## WORKING:

### Server:

1. The server is started on a port and waits for connections.
2. When a client(C1) is connected, it displays the details of the client.
3. The server then waits for the client to send a message, and then the server can reply to the client.
4. When a second client(C2) connects to the server, then the server will send a reply to C2.
5. Now, the server can simultaneously receive and send messages from and to C1 and C2.

- a. When C1 sends a message, reply goes to C1.
  - b. When C2 sends a message, reply goes to C2.
  - c. If two messages arrive from both clients, then the earliest message gets the priority in receiving a reply.
    - i. If C1 sends a message and then C2 sends a message, the reply from the server will go to C2 first.
    - ii. After C2 receives a reply, C1 can receive a reply.
6. The server can stop communication at any point by sending an 'exit' message.

#### Client:

1. The client will connect to a server online through the respective port.
2. When a client (C1) connects to the server, it can send and receive messages.
3. If another client (C2) connects to the server as well and sends a message, the two clients C1 and C2 will be independent of each other.
  - a. The messages sent by C1 will not be visible or accessible by C2 and vice versa.
4. In a situation where both clients send a message without receiving a reply from the server immediately, the server first replies to the latest message and then the earlier client can get a reply.
  - a. If C1 sends a message first and C2 sends a message after C1, then C2 will have the priority in receiving a reply.
  - b. After C2 receives a reply, C1 is open to receiving replies.
5. At any point, the client can stop the communication by sending an 'exit' message.

Note: Video demonstration contains the outputs of the server and the last client.

#### Client pk:

```
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ ls
client  client.c~  server.c  tcpmc  tcpms
client.c  server  server.c~  tcpmc.c  tcpms.c
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ ./client

Enter Name: pk
Enter Port: 8080
[+] Server Connected
Enter exit to stop
Enter message to send

Enter message to send
pk here
Hi PK

Enter message to send
hello vamsi how are you?
Hi KB

Enter message to send
hello vamsi - pk
Hi PK

Enter message to send
exit
Do you want to exit (Enter 1 to exit): 1
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ cd ..
```

### Client kb:

```
client client.c client.c- server server.c server.c- tcpnc tcpnc.c tcpns tcpns.c
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ ./client

Enter Name: kb
Enter Port: 8080
[+] Server Connected
Enter exit to stop
Enter message to send

Enter message to send
Hi this is kb
Hi PK, I'm good

Enter message to send
exit
Do you want to exit (Enter 1 to exit): 1
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ cd
```

### Client Srimi:

```
MATE Terminal File Edit View Search Terminal Help
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ ./client

Enter Name: Srimi
Enter Port: 8080
[+] Server Connected
Enter exit to stop
Enter message to send

Enter message to send
hello
Hello Srimi

Enter message to send
exit
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$
```

### Client dvss:

```
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$ ./client

Enter Name: dvss
Enter Port: 8080
[+] Server Connected
Enter exit to stop
Enter message to send

Enter message to send
dvsshere
Hi DVSS

Enter message to send
exit
Do you want to exit (Enter 1 to exit): 1
vamsikrshnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q1/bmc$
```

- Here, client pk has received a message that was meant for client kb.
  - This occurrence was due to pk connecting before kb could receive a reply, and thus the buffer is then rerouted to pk.
- Later, kb receives a message which was meant for pk.
  - Here, after pk has received a reply for his message, the next message from the server will be sent to the next waiting client, which is the client kb.

**2: Develop a game , where multiuser participate in the game. Server has 5 questions with four options. if users are connected to the server, the server starts sending the question one by one with a timestamp reply of 1 minute. If a multiuser plays the game, then the winner is decided by the server based on the average minimum time taken by a client to reply to all answers. If a client gives a wrong reply to anyone of the answers, then the server sends a message to the client "better luck next time" and terminates the connection of the client.**

#### **APPROACH :**

The server binds and accepts multiple clients using process/thread creation through **fork()** . Several APIs and Functions were used to clock the response time and thus decide the winner based on the time clocked.

**Select** function has been used to clock the timeout and synchronization of input and output

**select, pselect, FD\_CLR, FD\_ISSET, FD\_SET, FD\_ZERO** - synchronous I/O multiplexing

- **int select(int *nfds*, fd\_set \**readfds*, fd\_set \**writfds*,fd\_set \**exceptfds*, struct timeval \**timeout*);**
- **void FD\_CLR(int *fd*, fd\_set \**set*);**
- **int FD\_ISSET(int *fd*, fd\_set \**set*);**
- **void FD\_SET(int *fd*, fd\_set \**set*);**
- **void FD\_ZERO(fd\_set \**set*);**

select() and pselect() allow a program to monitor multiple file descriptors, waiting until one or more of the file descriptors become "ready" for some class of I/O operation (e.g., input possible). A file descriptor is considered ready if it is possible to perform the corresponding I/O operation.

**select()** uses a timeout that is a *struct timeval* (with seconds and microseconds), while **pselect()** uses a *struct timespec* (with seconds and nanoseconds).

**select()** may update the *timeout* argument to indicate how much time was left. **pselect()** does not change this argument.

## The timeout

The time structures involved are defined in [`<sys/time.h>`](#) and look like

```
struct timeval {  
    long   tv_sec;      /* seconds */  
    long   tv_usec;     /* microseconds */  
};
```

An *fd\_set* is a fixed size buffer. Executing **FD\_CLR()** or **FD\_SET()** with a value of *fd* that is negative or is equal to or larger than **FD\_SETSIZE** will result in undefined behavior. Moreover, POSIX requires *fd* to be a valid file descriptor.

## CODE :

The code is highly modular and reusable . (code provided in folder).

## WORKING:

### Server:

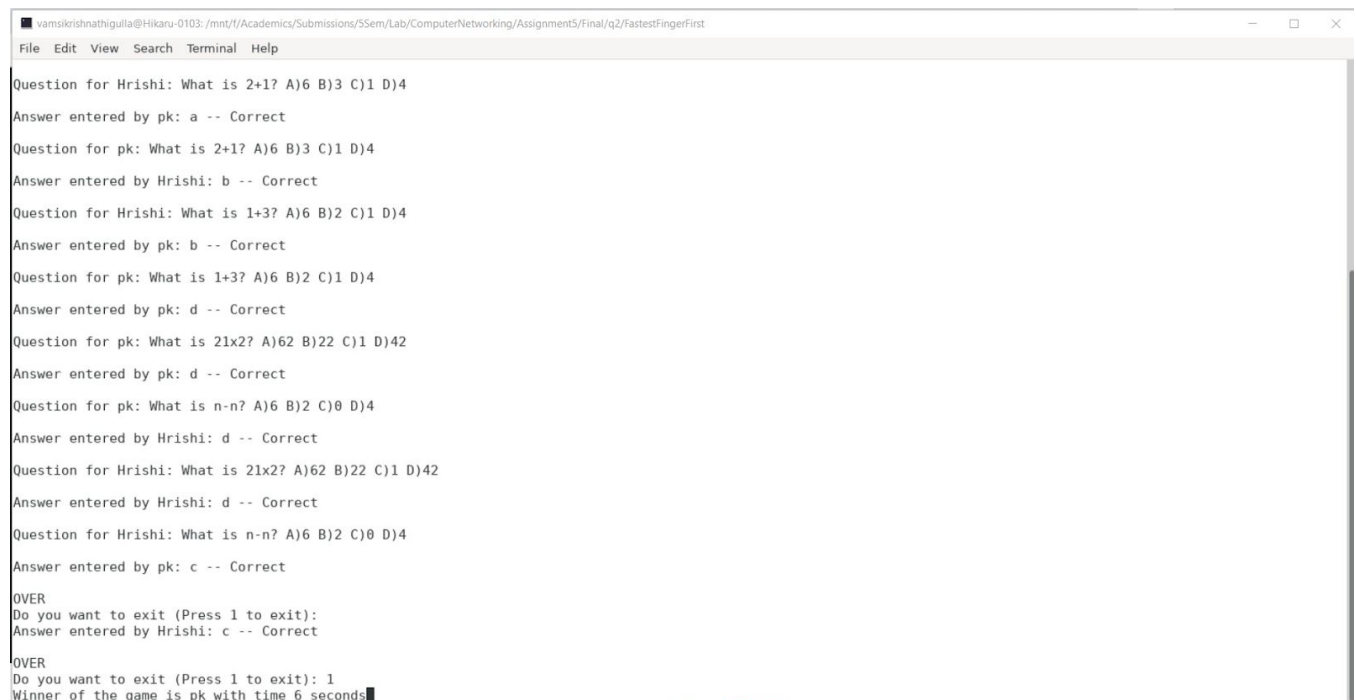
1. Start the server on a port and let the players i.e. clients connect to the server through that port.
2. Whenever a connection is accepted, the server displays the details of the player that has entered the server i.e. entered the game.
  - The name of the player and the port through which he has entered will be displayed.
3. The server displays the question and the player answering that question, and then displays the answer given by the player, followed by whether it was correct, wrong or timed out, and displays the time taken.
4. Then the server sends the next question and so on.
5. After the players have finished the game, the server can choose to end that game, and then the server displays the winner of the game along with the time taken for their attempt.
  - Winner of the game is the player who has answered all questions correctly.
  - If two players have the same score (answered the same number of questions) then the player with the fastest attempt i.e. player that has taken the least time to finish the game, will be declared the winner.
6. After the winner is declared, if the players connect again, we can start another game.

### Client:

1. Once the server is online, the clients i.e. the players can connect to the server and start the game at their own leisure.
  - No player has to wait for the other players to join to start playing his own game.
2. Each player's attempts are independent of the other players and will not interfere with the functioning of the other players' games.
3. Once the player provides the correct answer, he can proceed to the next question.
  - The player has to answer the question in a maximum time of 10 seconds, before it goes into timeout.
    - This maximum time for timeout can be changed any time through simple modifications in the code.
4. If a player provides a wrong answer at any point during the game, he will be no longer eligible to play for the remainder of the game and will be automatically disqualified.
  - Such a player will have to login (connect) again and start over if he wants to play again after giving a wrong answer in the previous attempt.
5. Once an attempt is complete, the player can be shown the statistics of the attempt, like the number of questions answered correctly, the total time taken for the attempt, etc.
6. A player can then choose to play again and start over his attempt.

### Server:

Screenshot of the game showing the winner with simultaneous players



```
vamsikrishnathigulla@Hikaru-0103: /mnt/1/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst
File Edit View Search Terminal Help
Question for Hrishi: What is 2+1? A)6 B)3 C)1 D)4
Answer entered by pk: a -- Correct
Question for pk: What is 2+1? A)6 B)3 C)1 D)4
Answer entered by Hrishi: b -- Correct
Question for Hrishi: What is 1+3? A)6 B)2 C)1 D)4
Answer entered by pk: b -- Correct
Question for pk: What is 1+3? A)6 B)2 C)1 D)4
Answer entered by pk: d -- Correct
Question for pk: What is 21x2? A)62 B)22 C)1 D)42
Answer entered by pk: d -- Correct
Question for pk: What is n-n? A)6 B)2 C)0 D)4
Answer entered by Hrishi: d -- Correct
Question for Hrishi: What is 21x2? A)62 B)22 C)1 D)42
Answer entered by Hrishi: d -- Correct
Question for Hrishi: What is n-n? A)6 B)2 C)0 D)4
Answer entered by pk: c -- Correct
OVER
Do you want to exit (Press 1 to exit):
Answer entered by Hrishi: c -- Correct
OVER
Do you want to exit (Press 1 to exit): 1
Winner of the game is pk with time 6 seconds
```

Server for the player who answered a wrong question

Question for Srini: What is 5+1? A)6 B)2 C)1 D)4

Answer entered by Srini: a -- Correct

Question for Srini: What is 2+1? A)6 B)3 C)1 D)4

Answer entered by Srini: b -- Correct

Question for Srini: What is 1+3? A)6 B)2 C)1 D)4

Answer entered by Srini: c -- Wrong -- Expected d

OVER

Do you want to exit (Press 1 to exit): █

**Client:**

All Correct Answers:

```
vamsikrishnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$ ./client
Enter Name: pk
Enter Port: 8080
[+] Server Connected

Question: What is 5+1? A)6 B)2 C)1 D)4
a
Enter Answer:
Correct!

Question: What is 2+1? A)6 B)3 C)1 D)4
b
Enter Answer:
Correct!

Question: What is 1+3? A)6 B)2 C)1 D)4
d
Enter Answer:
Correct!

Question: What is 21x2? A)62 B)22 C)1 D)42
d
Enter Answer:
Correct!

Question: What is n-n? A)6 B)2 C)0 D)4
c
Enter Answer:
Correct!

Result:
Correct: 5
Wrong: 0
Late: 0
Total no of questions: 5
Total time: 6
```

```
vamsikrishnathigulla@Hikaru-0103: /mnt/f/Academics/Submissions/5Sem/Lab/ComputerN...
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$ ./client

Enter Name: Hrishi
Enter Port: 8080
[+] Server Connected

Question: What is 5+1? A)6 B)2 C)1 D)4
a
Enter Answer:
Correct!

Question: What is 2+1? A)6 B)3 C)1 D)4
b
Enter Answer:
Correct!

Question: What is 1+3? A)6 B)2 C)1 D)4
d
Enter Answer:
Correct!

Question: What is 21x2? A)62 B)22 C)1 D)42
d
Enter Answer:
Correct!

Question: What is n-n? A)6 B)2 C)0 D)4
c
Enter Answer:
Correct!

Result:
Correct: 5
Wrong: 0
Late: 0
Total no of questions: 5
Total time: 8
Do you want to exit (Enter 1 to exit): 1
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$
```

Client giving a wrong answer:

```
MATE Terminal File Edit View Search Terminal Help
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$ ./client

Enter Name: Srini
Enter Port: 8080
[+] Server Connected

Question: What is 5+1? A)6 B)2 C)1 D)4
a
Enter Answer:
Correct!

Question: What is 2+1? A)6 B)3 C)1 D)4
b
Enter Answer:
Correct!

Question: What is 1+3? A)6 B)2 C)1 D)4
c
Enter Answer:
vamsikrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sem/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$
```



Client going into timeout:

```

vanstkrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sen/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$ ./client

Enter Name: kb
Enter Port: 8080
[+] Server Connected

Question: What is 5+1? A)6 B)2 C)1 D)4
a
Enter Answer:
Correct!

Question: at is 2+1? A)6 B)3 C)1 D)4
Enter Answer: TimeOut
vanstkrishnathigulla@Hikaru-0103:/mnt/f/Academics/Submissions/5Sen/Lab/ComputerNetworking/Assignment5/Final/q2/FastestFingerFirst$

```