TEST DOCUMENT

ASSIGNMENT QUESTION :

Design and implement an emergency server. It should answer the sample clients questions:

1. **Police Station number**
2. **Ambulance Number**
3. **Fire Station Number**
4. **Vehicle Repair Number**
5. **Food Delivery**
6. **Blood Bank Number**

Constraints:

1. **Clients do not know the IP address or hostname of the emergency server.**
2. **Emergency services should be reliable.**
3. **You can assume an Emergency server running on a particular port number.**
4. **Multiple clients may send queries parallelly to the emergency server.**

Server Code :

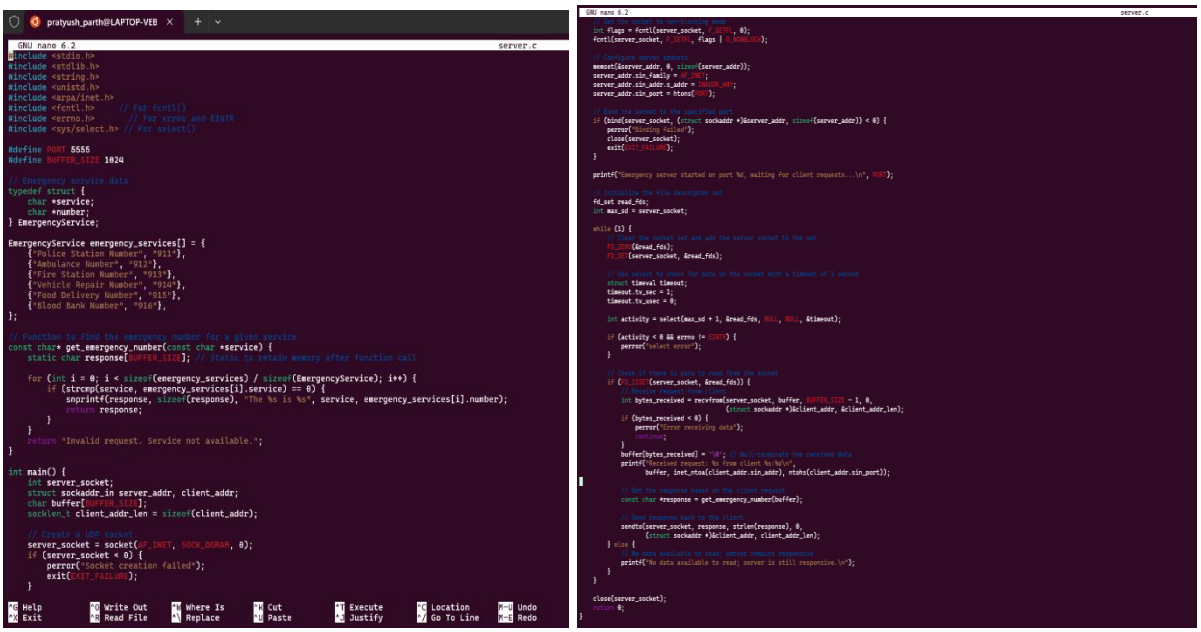

Fig : Server Code

Client Code :

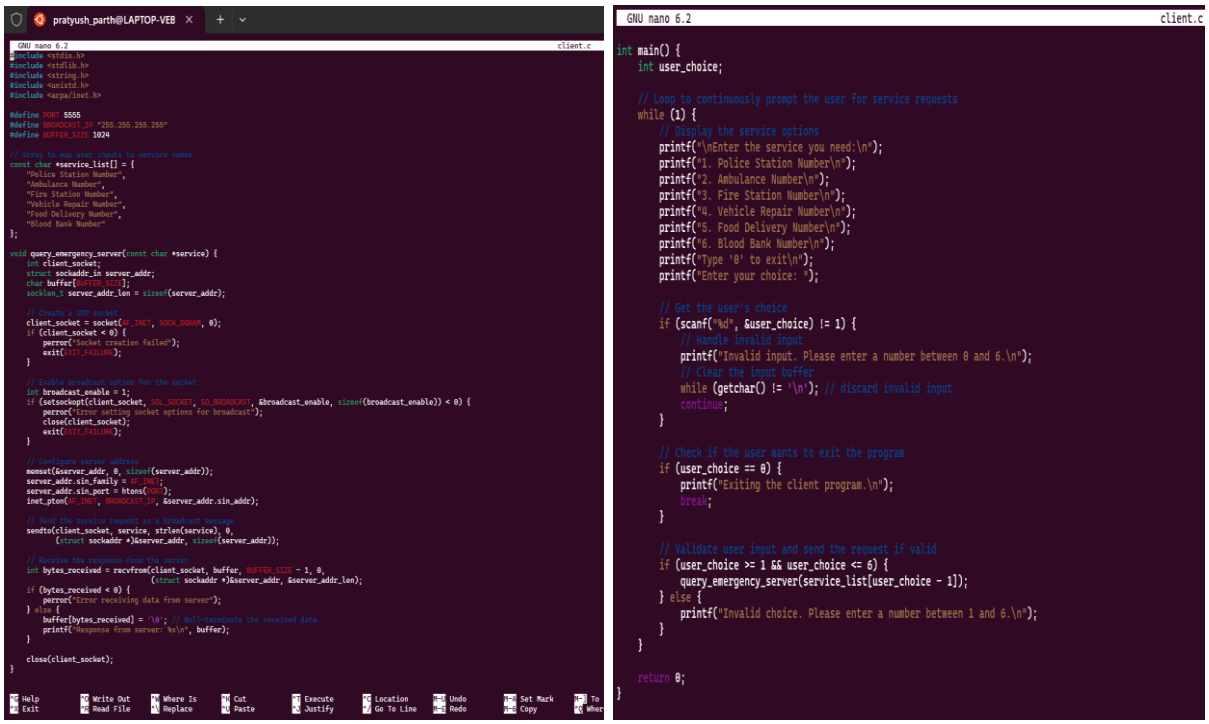

Fig : Client Code

TEST CASES :

1. **On starting the server first, it displays the message: 'Emergency server started on port 5555, waiting for client requests...'**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./server
Emergency server started on port 5555, waiting for client requests...
```

2. **On starting the client, it displays the following message in the menu-driven option format where the client is asked to write the choice from 1 to 6 and 0 to terminate.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice:
```

3. **On the server side, if there is no data available to read from the socket, the server displays a message indicating that it is still responsive.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./server
Emergency server started on port 5555, waiting for client requests...
No data available to read; server is still responsive.
```

4. **Demonstration of multiple clients on a single server, namely 1 server and 2 clients (Client 1 with IP Address 172.16.108.50 and Client 2 with IP Address 172.16.108.90)**

```
pratyush-parth@pratyush-parth-VirtualBox:~/Downloads/Emergency(1)/Emergency$ ./server
Emergency server started on port 5555, waiting for client requests...
No data available to read; server is still responsive.
No data available to read; server is still responsive.
No data available to read; server is still responsive.
No data available to read; server is still responsive.
No data available to read; server is still responsive.
Received request: Police Station Number from client 172.16.108.50:58051
Received request: Police Station Number from client 172.16.108.90:45000
Received request: Ambulance Number from client 172.16.108.50:56989
```

5. **On the client side, output and response from the server on choosing option 1.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: 1
Response from server: The Police Station Number is 911
```

6. **On the server side, output when the client makes a request of contact number of option 1.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./server
Emergency server started on port 5555, waiting for client requests...
No data available to read; server is still responsive.
Received request: Police Station Number from client 172.22.153.162:39380
```

7. **On the client side, output and response from the server on choosing option 6.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: 6
Response from server: The Blood Bank Number is 916
```

8. **On the server side, output when the client makes second request of contact number of option 6. Also, it demonstrates client with repeated IP and port sending multiple requests.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./server
Emergency server started on port 5555, waiting for client requests...
No data available to read; server is still responsive.
Received request: Police Station Number from client 172.22.153.162:55092
Received request: Blood Bank Number from client 172.22.153.162:53554
```

9. **On starting the client first before starting the server and choosing option 2.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency_System$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: 2
```

10. **On the server side, this is the output on the request of client.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency_System$ ./server
Emergency server started on port 5555, waiting for client requests...
Received request: Ambulance Number from client 127.0.0.1:60286
```

11. **On the client side, this is the response from the server.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency_System$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: 2
Response from server: The Ambulance Number is 912
```

12. **On the client side, output when the client chooses a text instead of appropriate numbers within 0 to 6.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: abcd
Invalid input. Please enter a number between 0 and 6.
```

13. **On the client side, output when the client chooses a number outside the range 1 to 6.**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: 8
Invalid choice. Please enter a number between 1 and 6.
```

14. **On the client side, on choosing 0, client terminates .**

```
pratyush_parth@LAPTOP-VEBSJD0U:~/Emergency$ ./client

Enter the service you need:
1. Police Station Number
2. Ambulance Number
3. Fire Station Number
4. Vehicle Repair Number
5. Food Delivery Number
6. Blood Bank Number
Type '0' to exit
Enter your choice: 0
Exiting the client program.
```

Apart from these cases,

- **In case of any network issue, when the server is temporarily unreachable and client sends a request, the client handles the error gracefully and displays the message 'Error receiving data from server.'**
- **On checking the server timeout test, if the client makes a request after a period of inactivity, the server correctly responds to the request without requiring a restart.**
- **In case of test with maximum data size, i.e., buffer size limit, the server handles the buffer size without crashing, responding correctly if the message size is within the limit.**

DEMONSTRATION OF HANDLING MULTIPLE CLIENTS ON A SINGLE SERVER

We successfully handled multiple clients on different systems from a single server, where we had 1 server and 3 clients, namely Client 1 with IP Address 172.16.108.50, Client 2 with IP Address 172.16.108.90 and Client 3 with IP Address 172.16.108.53 respectively, which we can clearly observe from the further screenshots in the next page.

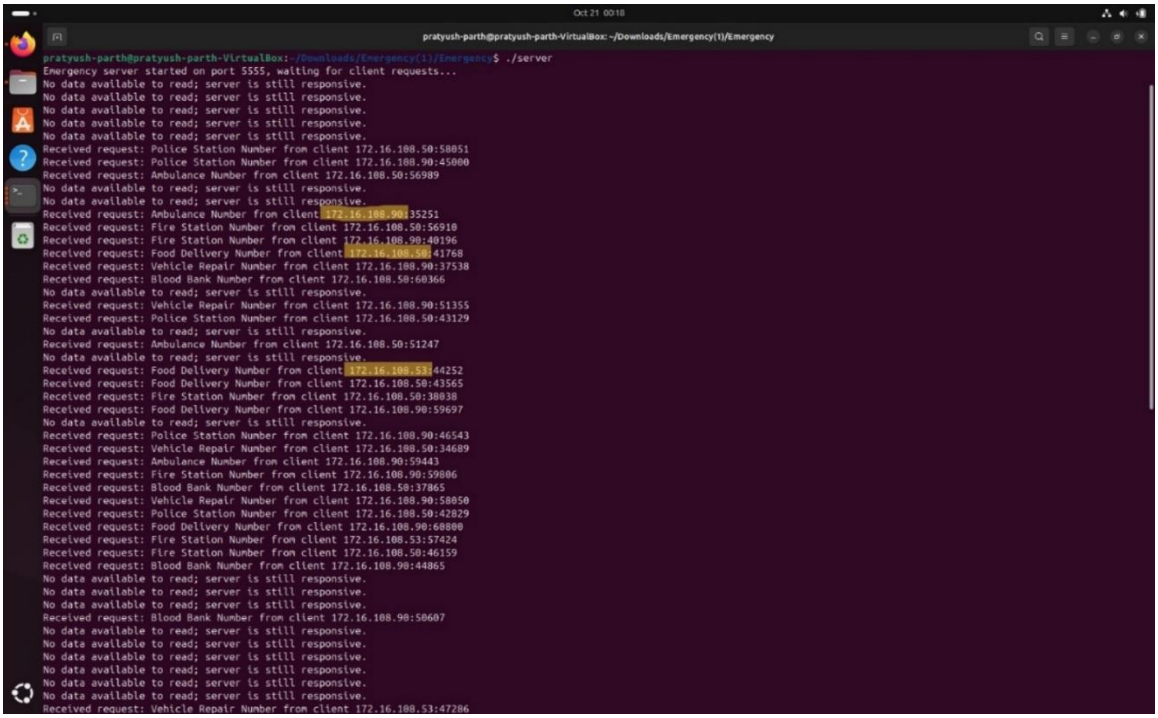

Fig: Server handling multiple clients parallely

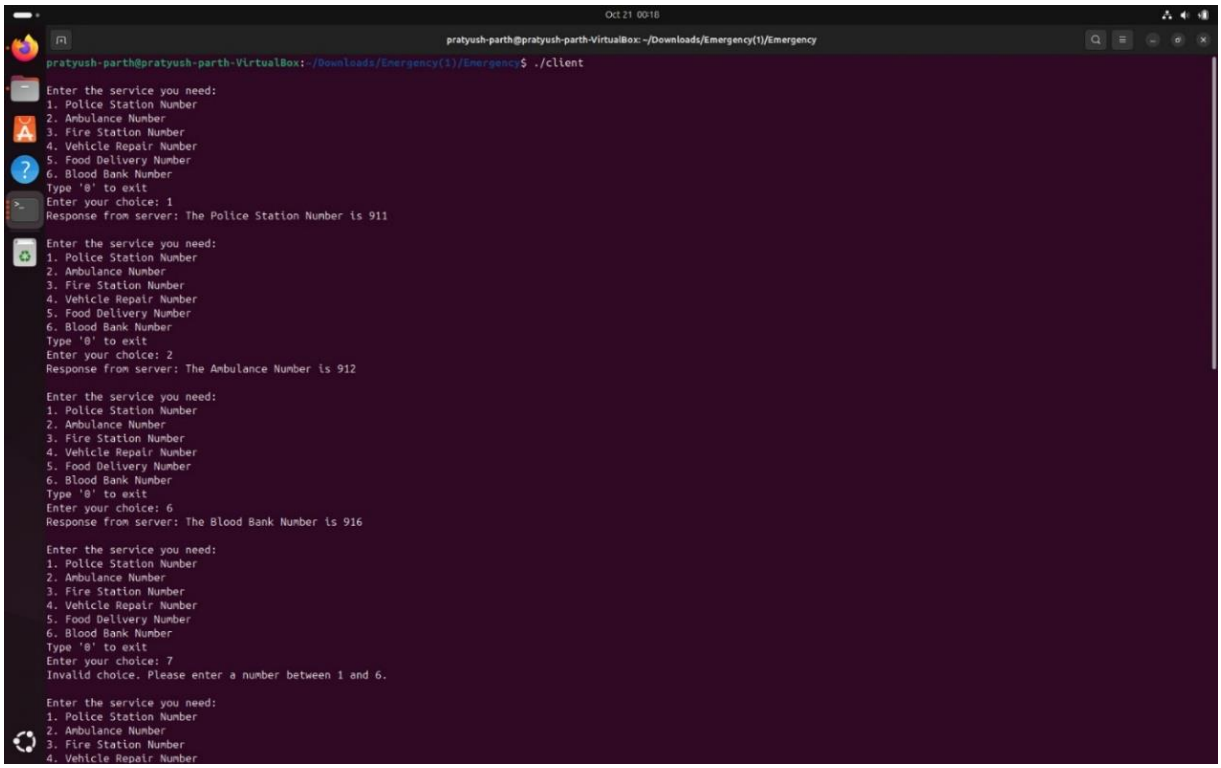

Fig: Client 1 with IP Address 172.16.108.50

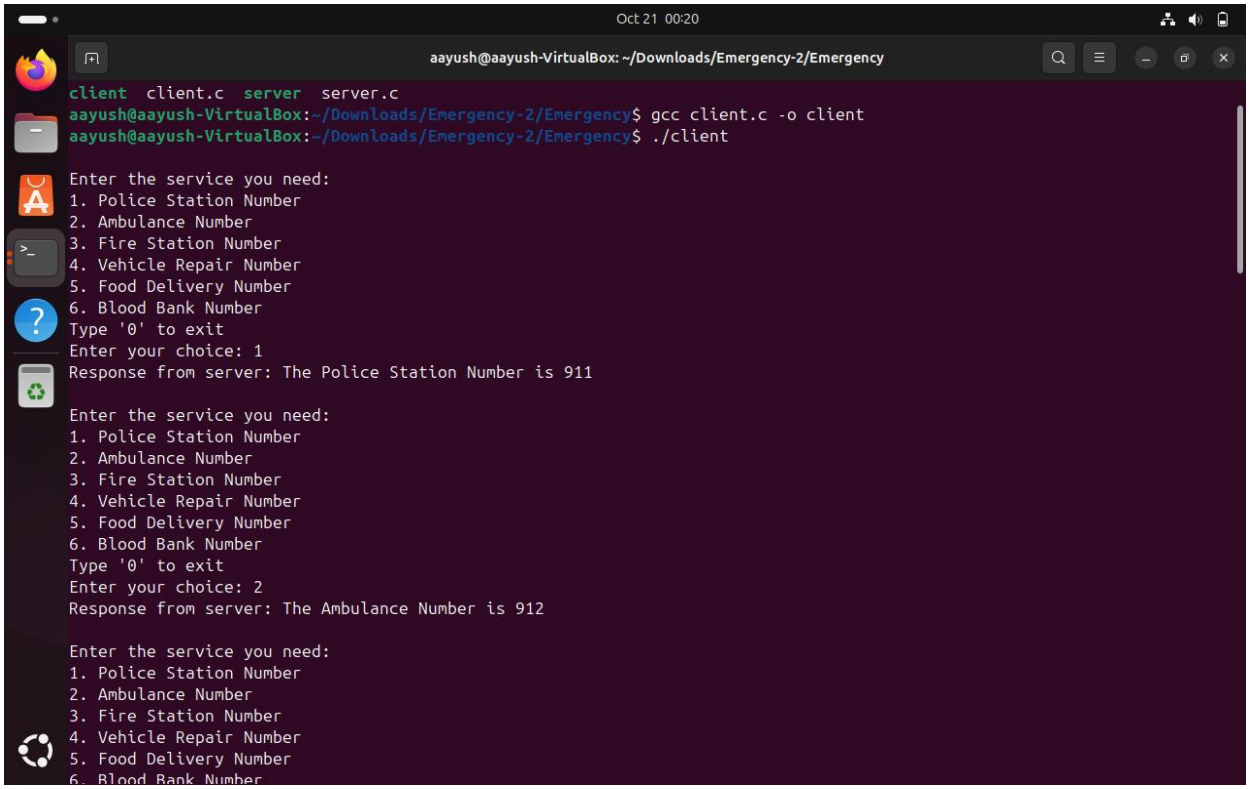

Fig: Client 2 with IP Address 172.16.108.90

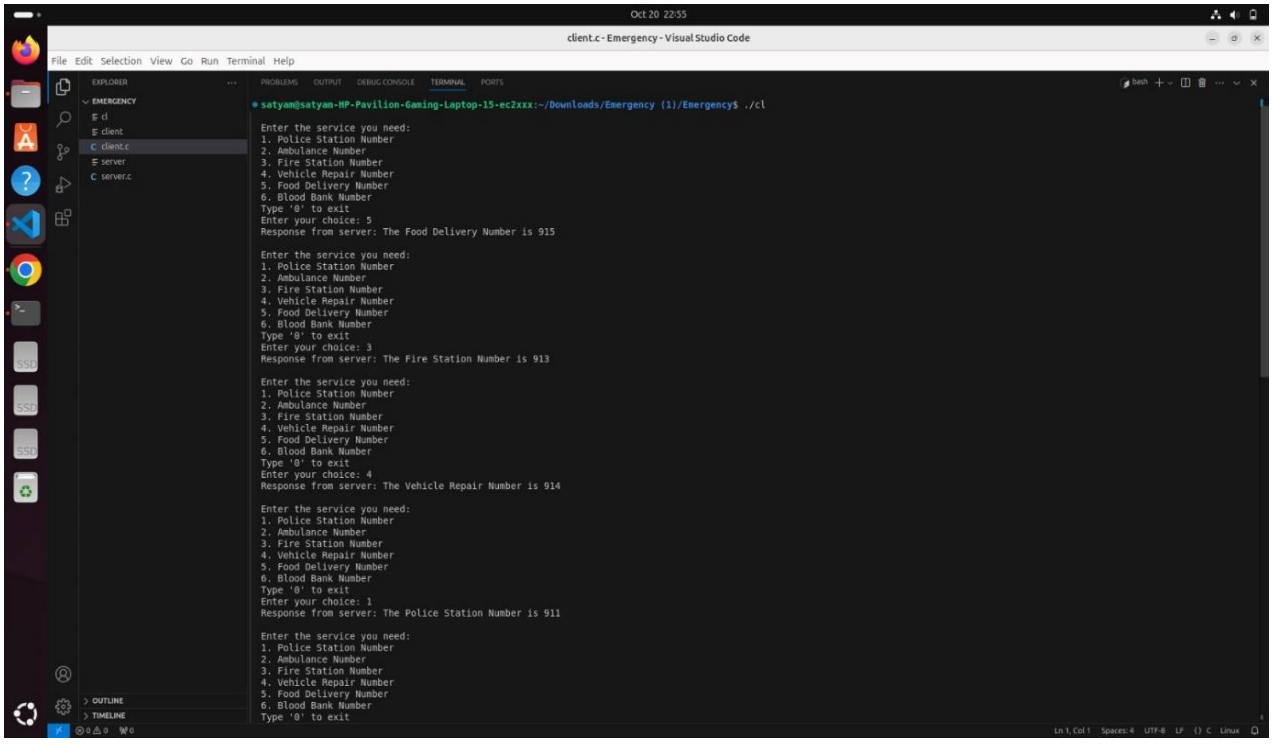

Fig: Client 3 with IP Address 172.16.108.53

************************************THANK************************************
************************************YOU************************************