5.1 How to run my code.

I used kali Linux to compile and run my code. Locate where you’ve saved the C source files in your linux machine. Open terminal there and type ***ls*** to confirm the files are in the same directory as you. We will first compile and run the server-side program. Type ***gcc server-side.c -o server -lpthread*** and hit enter.

gcc is used to compile our c program into an executable program. The -o command specifies the name of our executable program and -lpthread command tells the gcc compiler that it must link the pthread library to the compiled executable. After hitting enter now type ***./server*** to run our program. If ready the program will display listening for incoming connections.

Open another tab in the same directory.Type ***gcc client-side.c -o client -lpthread*** to compile the program and hit enter. Then type ***./client*** to run the compiled program. The server program will communicate with the client program and display the results on both terminals.

5.2 Performance description

In the concurrent server, it takes lesser time for it to execute than the iterative server since it doesn’t need to wait for any process to exit so as the other process can start as in iterative server.

|  |  |  |
| --- | --- | --- |
| **Measurement #** | **delay\_IS (ms)** | **delay\_CS (ms)** |
| 1 | 33.974 | 7.037 |
| 2 | 34.335 | 5.895 |
| 3 | 36.394 | 13.89 |
| 4 | 35.101 | 10.634 |
| 5 | 40.282 | 13.336 |
| 6 | 34.625 | 6.08 |
| 7 | 33.73 | 9.585 |
| 8 | 37.15 | 11.974 |
| 9 | 35.359 | 12.094 |
| 10 | 37.872 | 12.381 |
| **Average** | 35.8822 | 10.2906 |
| **Standard deviation** | 2.063394819 | 2.998854604 |