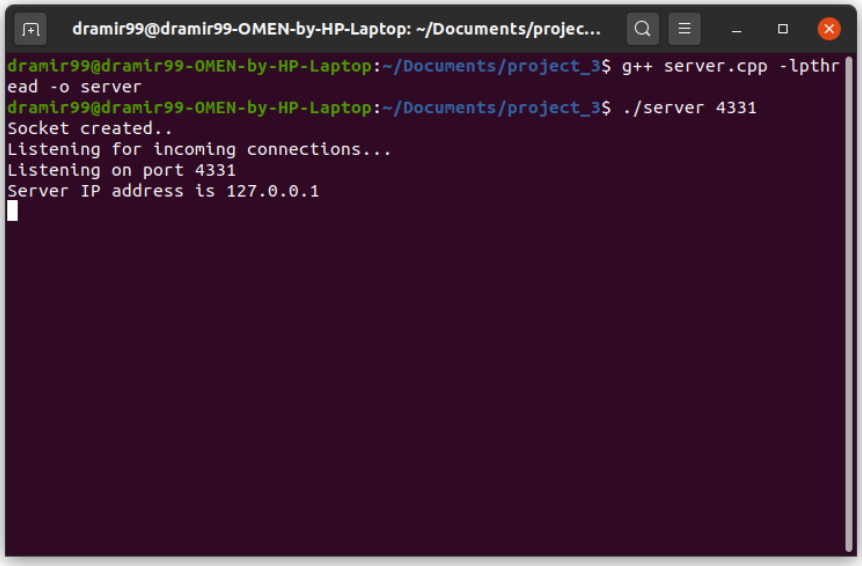
Technical report

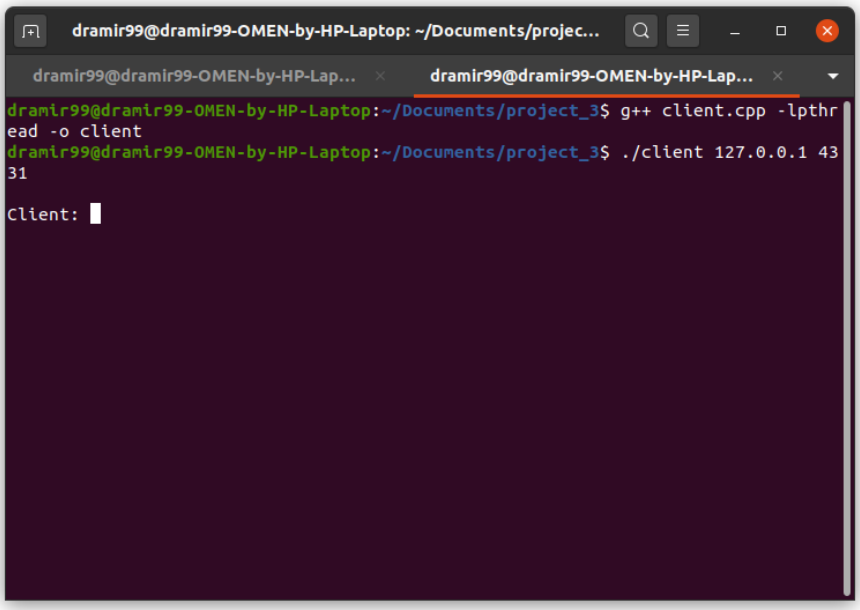
By: Daniel R
Daniel G
Carlos M

1. **Basic Client-Server:**

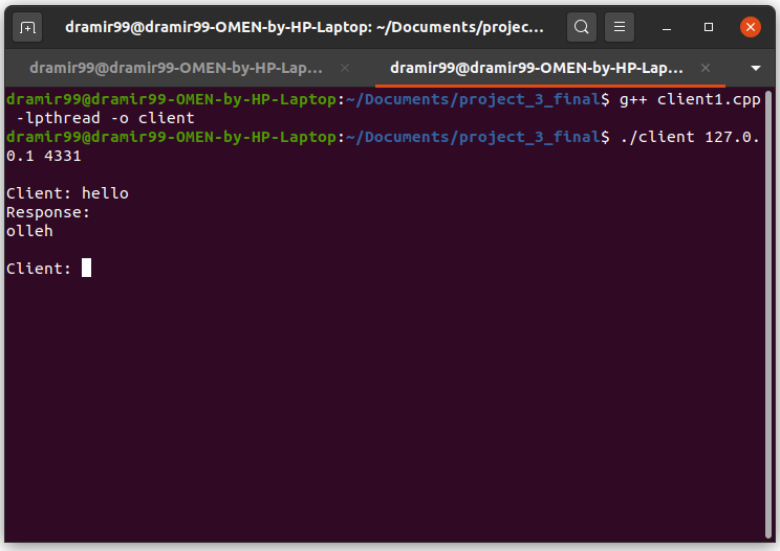
- First compile and launch server1



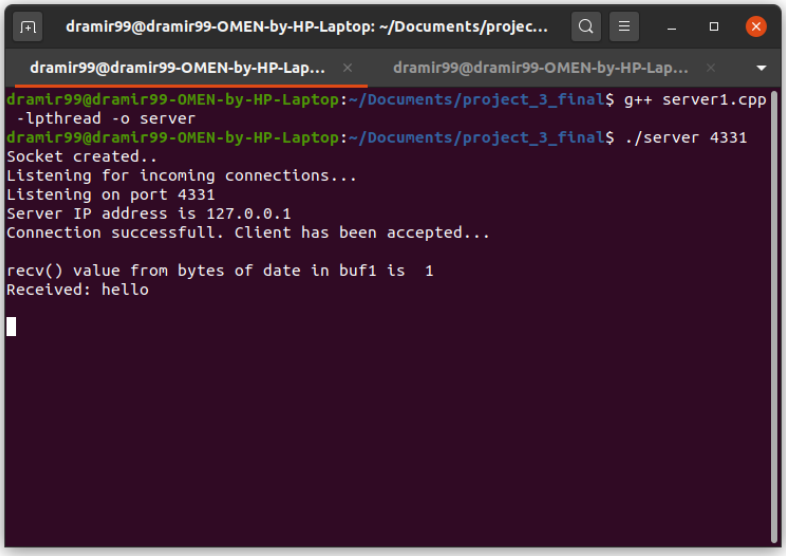
- Then compile and launch client1
    - In order to connect the client and server, in the client compile you must type in the argument from the server so for this example the port address i typed in is 4331 and ip address was set by default as 127.0.0.1



Now that you are in the client and server , in the client if you type a word in it will return the word reversed as shown

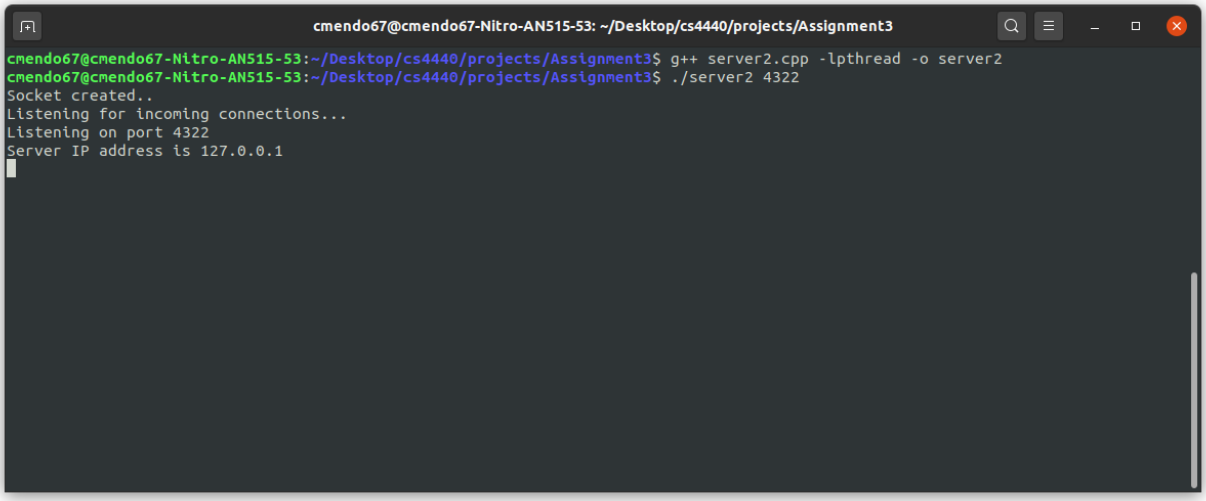


The server will show that it received the word

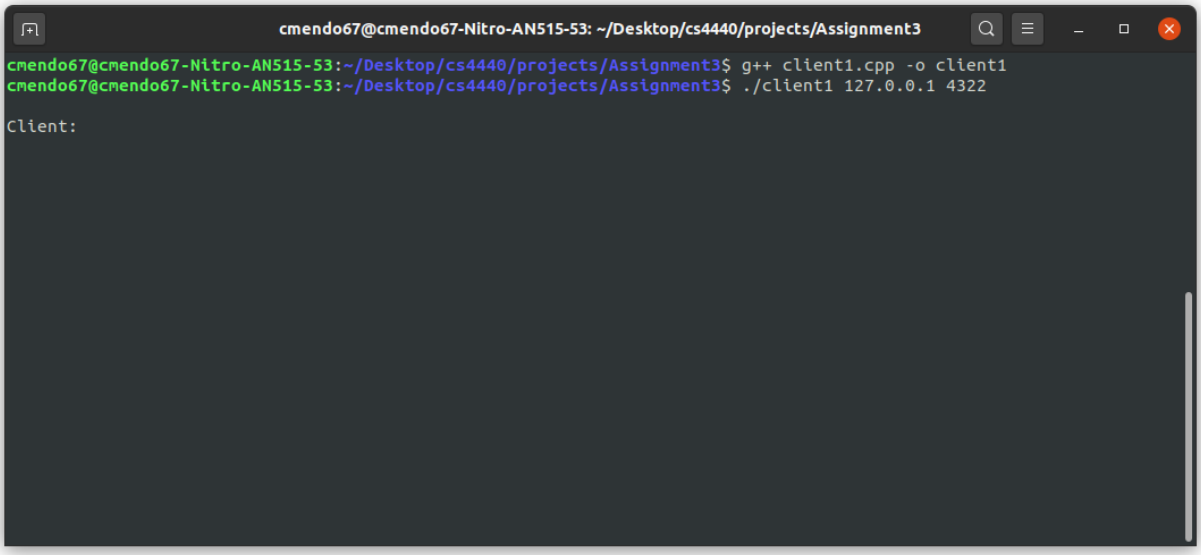


2. **Directory listing server:**

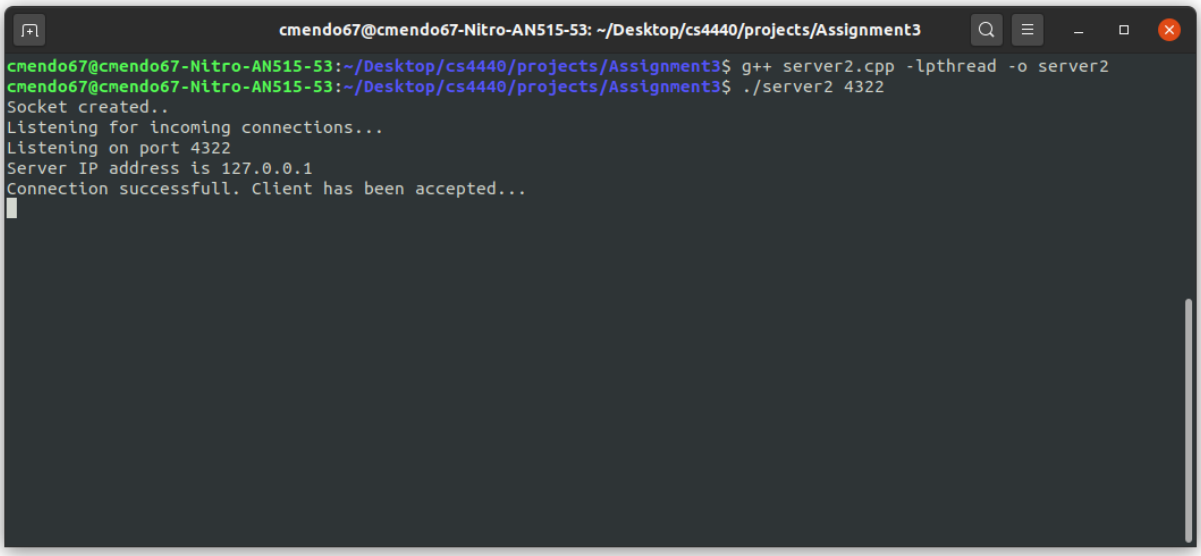
First compile and run server2



Then run and compile client1

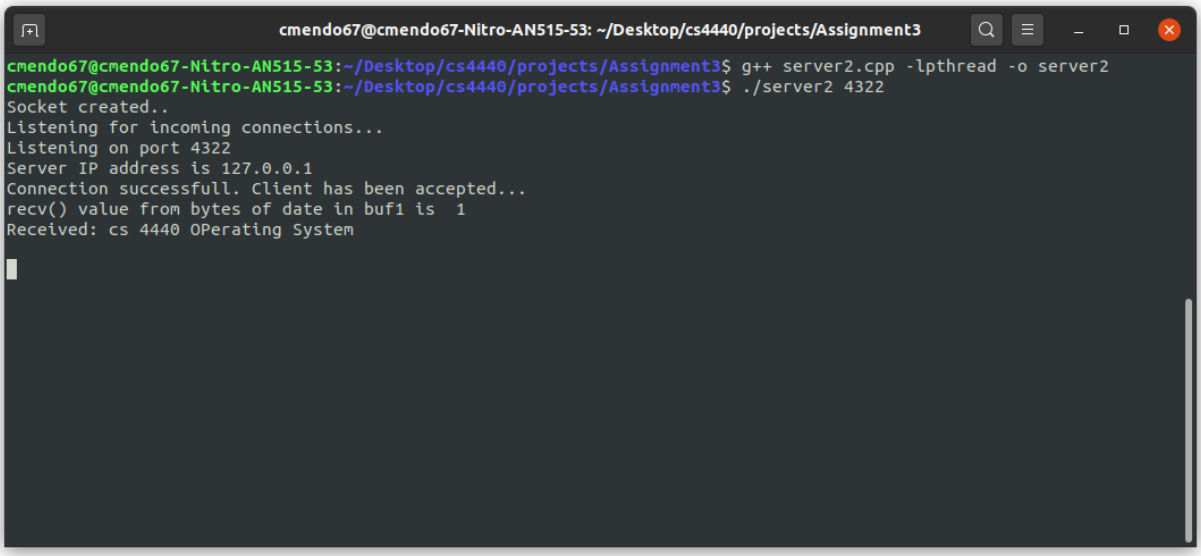


Server displays “Connection successful. Client has been accepted…” when client connects to server.

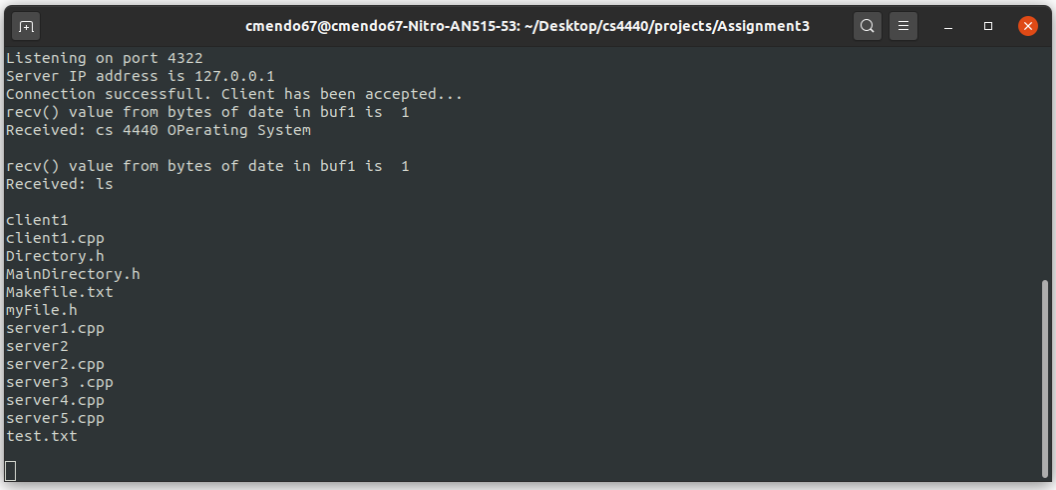


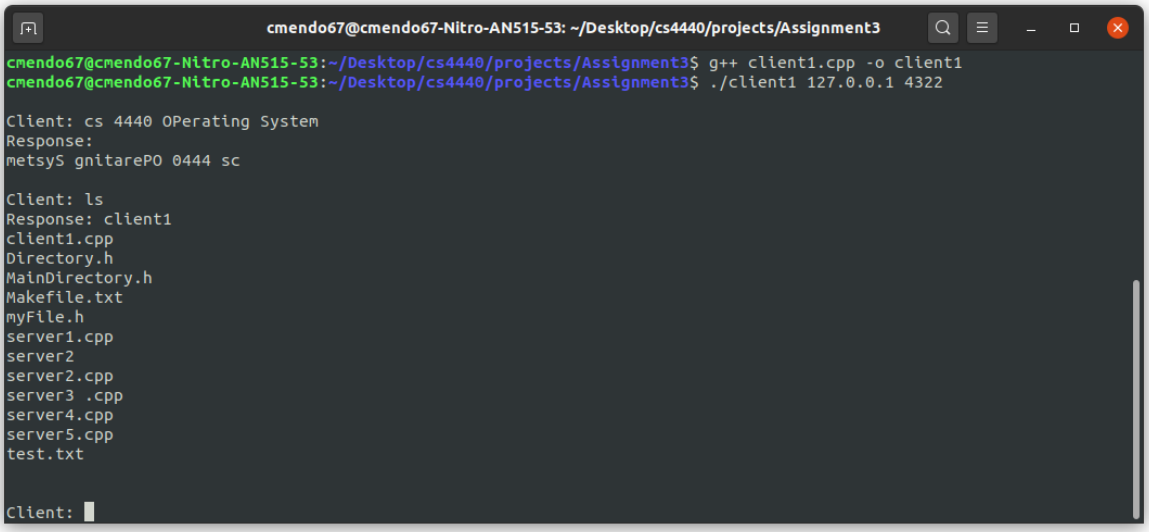
Client has connected to server. You can open terminal and do the same. Multiple clients can connect.

Now, If you enter a word in client, the server receives it and returns the input in reverse order. First problem works in problem 2 as well.



Now, when you insert ls from client side, it lists all the information from current folder. The server gets the output and returns output to client.



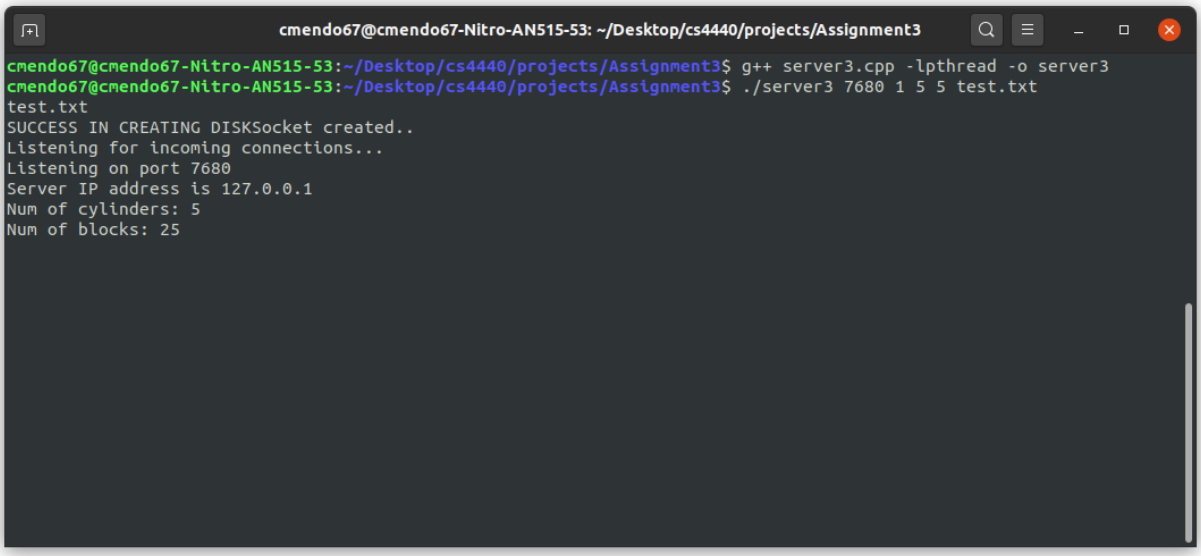


3) **Basic Disk Storage System**

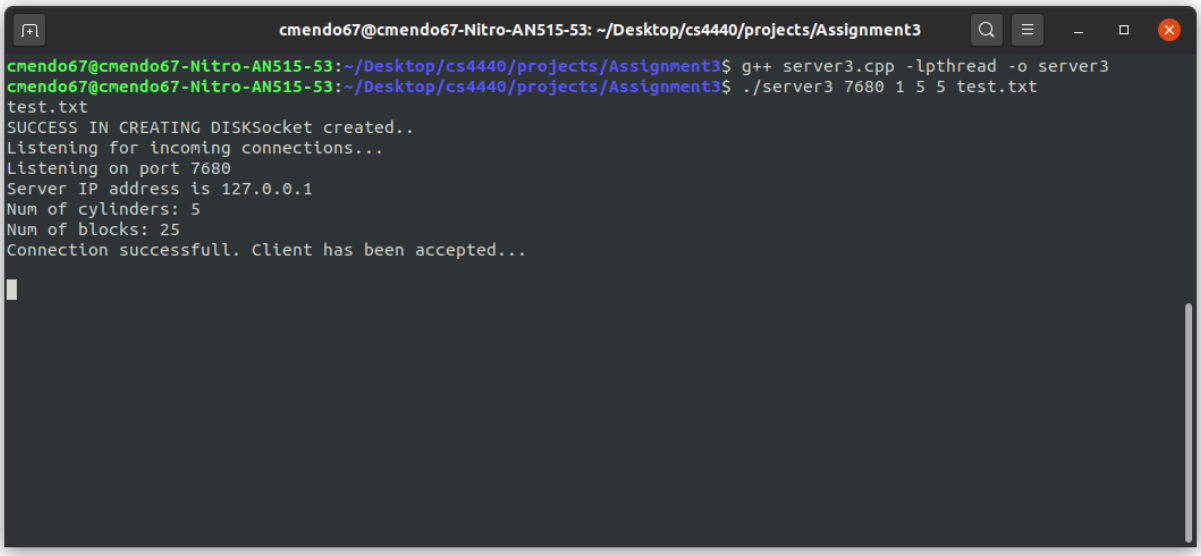
First compile and run server.

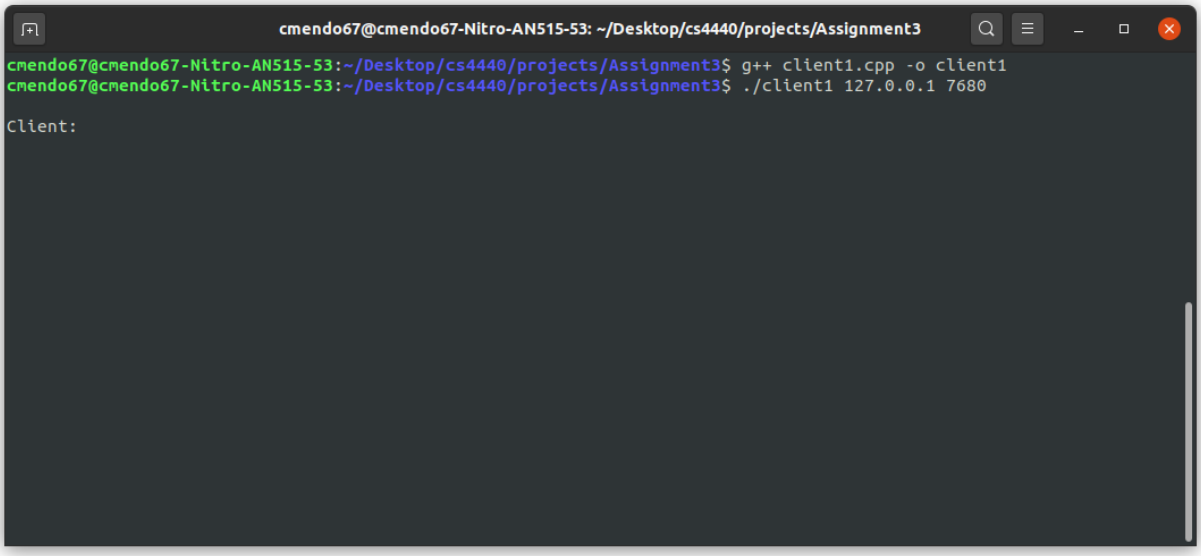
g++ server3.cpp -lpthread -o server3

./server3 <port Number> <Track to Track Time> <Cylinder> <Sectors> < File Name>



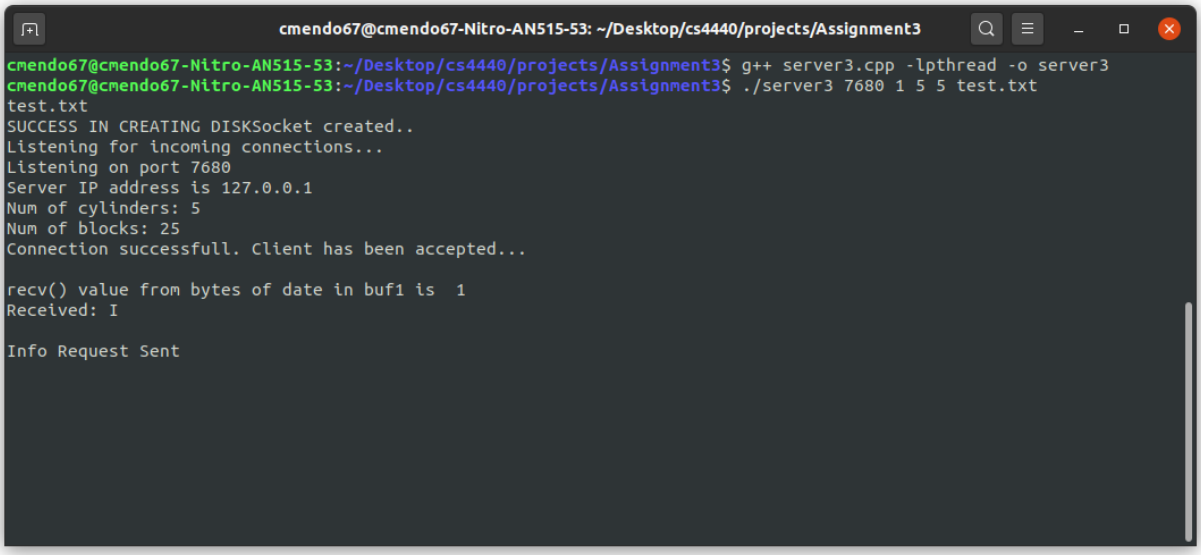
Then compile and run client1 to connect to server.



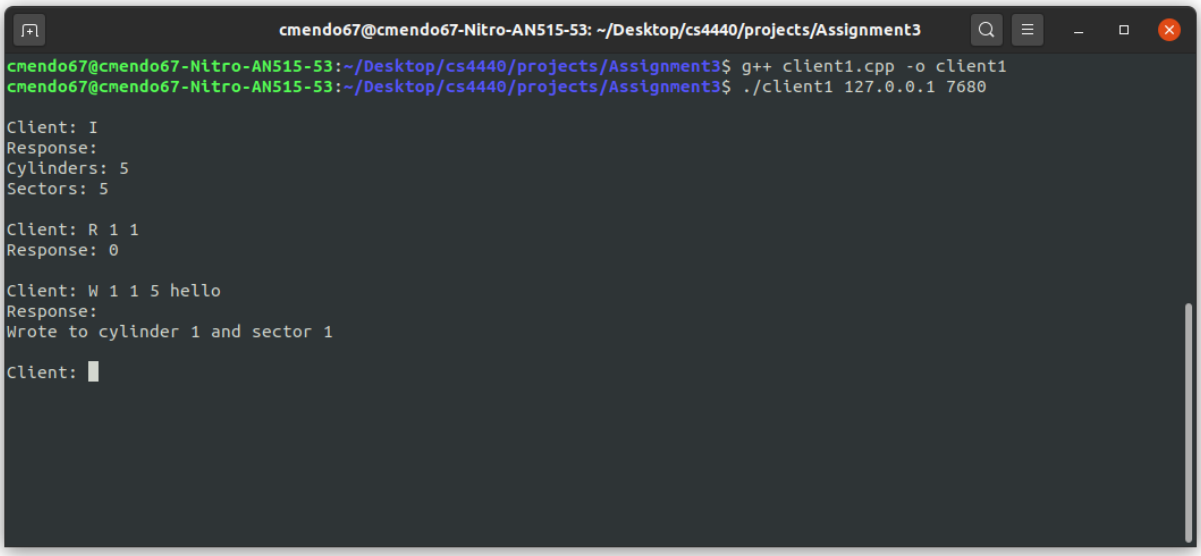


Now, if you enter I -

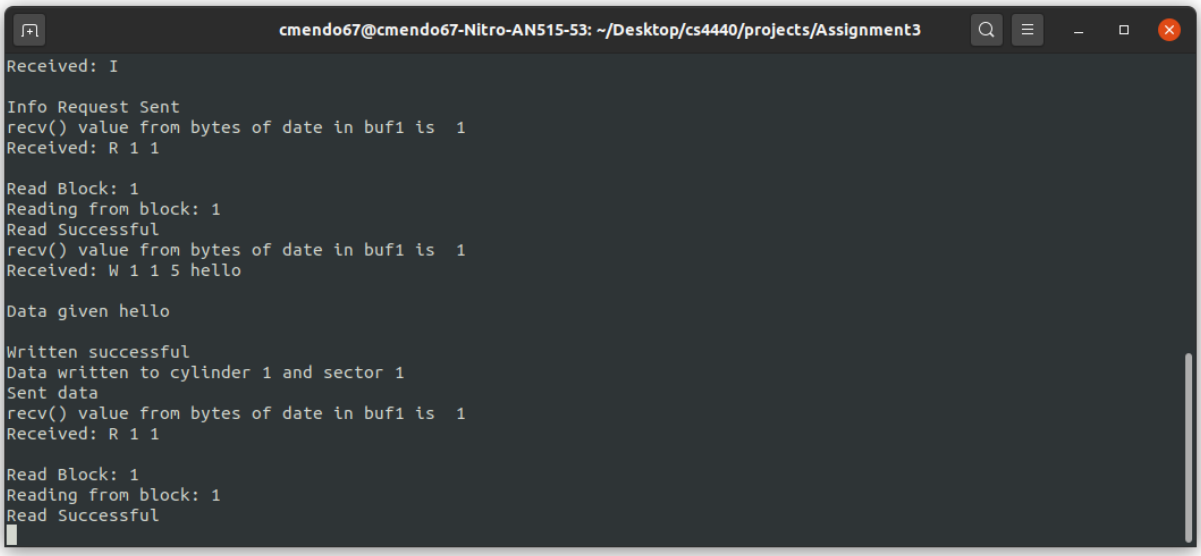
• I: information request. The disk returns two integers representing the disk geometry: the number of cylinders, and the number of sectors per cylinder.

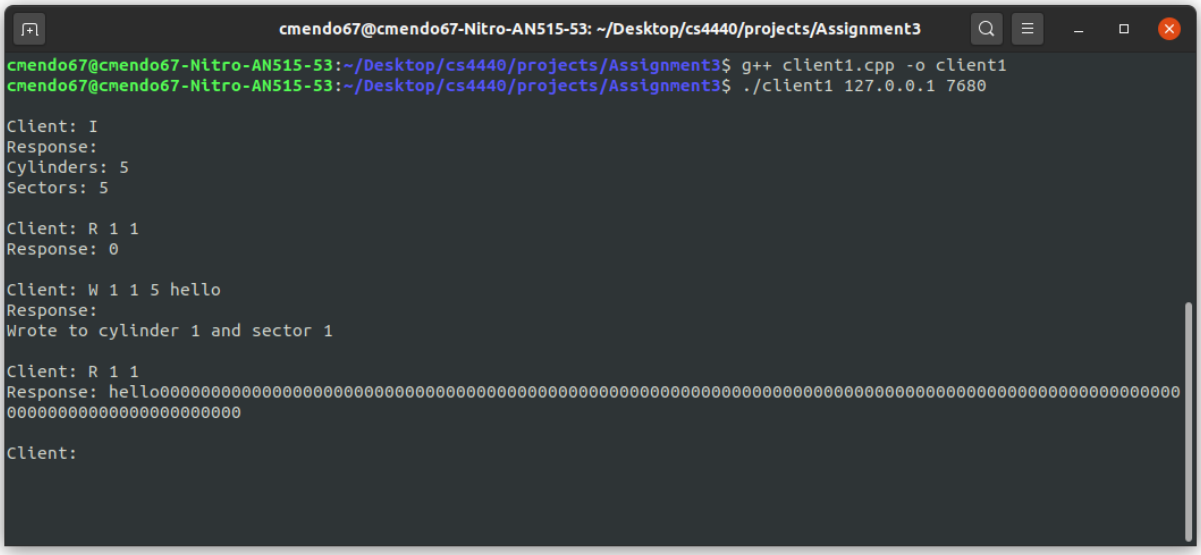


Now, if you run W c s data - : write request for cylinder c sector s. l is the number of bytes being provided, with a maximum of 128. The data is those l bytes of data. The disk returns ’1’ to the client if it is a valid write request (legal values of c, s and l), or returns a ’0’ otherwise.



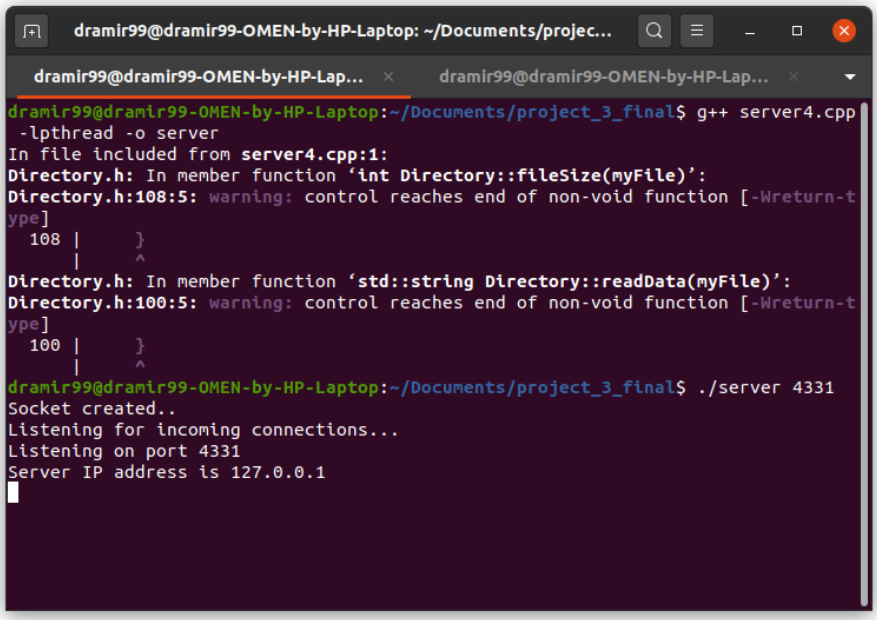
Now if you run R c s - This will return whatever data happens to be on the disk in a given sector, even if nothing has ever been explicitly written there before.)


cmendo67@cmendo67-Nitro-AN515-53: ~/Desktop/cs4440/projects/Assignment3
Received: I
Info Request Sent
recv() value from bytes of date in buf1 is 1
Received: R 1 1
Read Block: 1
Reading from block: 1
Read Successful
recv() value from bytes of date in buf1 is 1
Received: W 1 1 5 hello
Data given hello
Written successful
Data written to cylinder 1 and sector 1
Sent data
recv() value from bytes of date in buf1 is 1
Received: R 1 1
Read Block: 1
Reading from block: 1
Read Successful

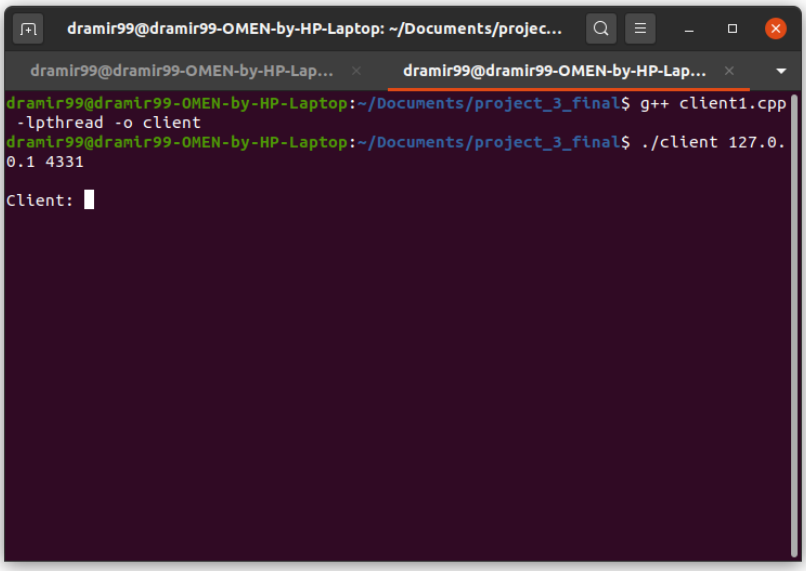

cmendo67@cmendo67-Nitro-AN515-53: ~/Desktop/cs4440/projects/Assignment3
cmendo67@cmendo67-Nitro-AN515-53:~/Desktop/cs4440/projects/Assignment3$ g++ client1.cpp -o client1
cmendo67@cmendo67-Nitro-AN515-53:~/Desktop/cs4440/projects/Assignment3$ ./client1 127.0.0.1 7680
Client: I
Response:
Cylinders: 5
Sectors: 5
Client: R 1 1
Response: 0
Client: W 1 1 5 hello
Response:
Wrote to cylinder 1 and sector 1
Client: R 1 1
Response: hello
Client:

4. **File System Server:**

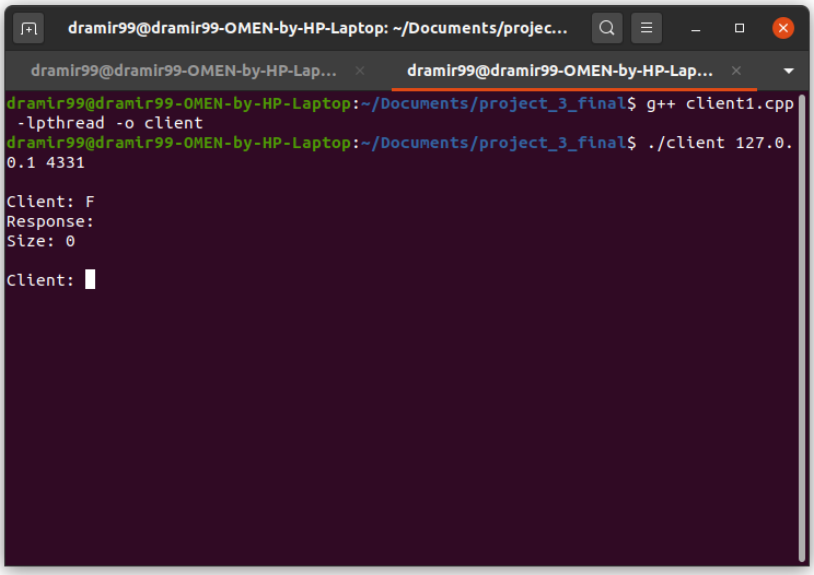
- First compile and launch server4



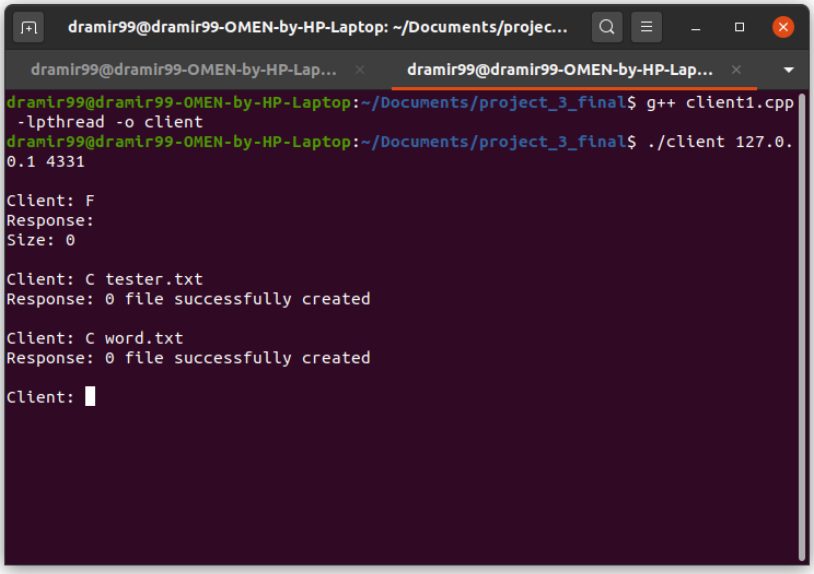
- Then compile and launch client1
  - In order to connect the client and server, in the client compile you must type in the argument from the server so for this example the port address i typed in is 4331 and ip address was set by default as 127.0.0.1



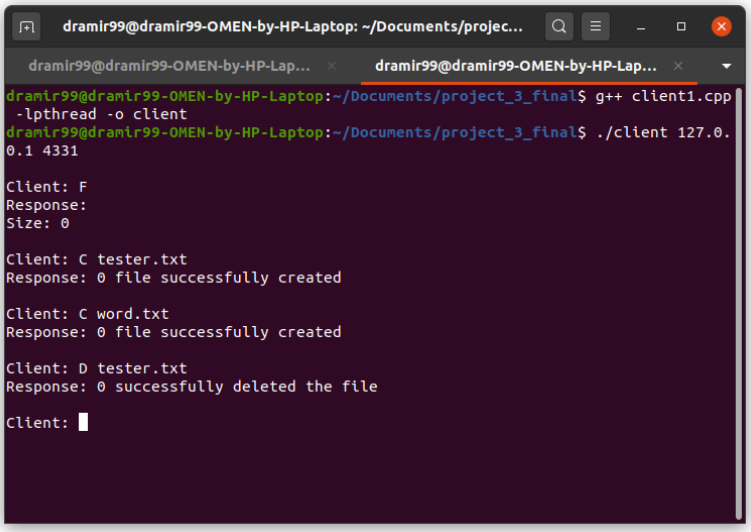
Now that you are in the client and server , in the client if you type ‘F’ you will get the size



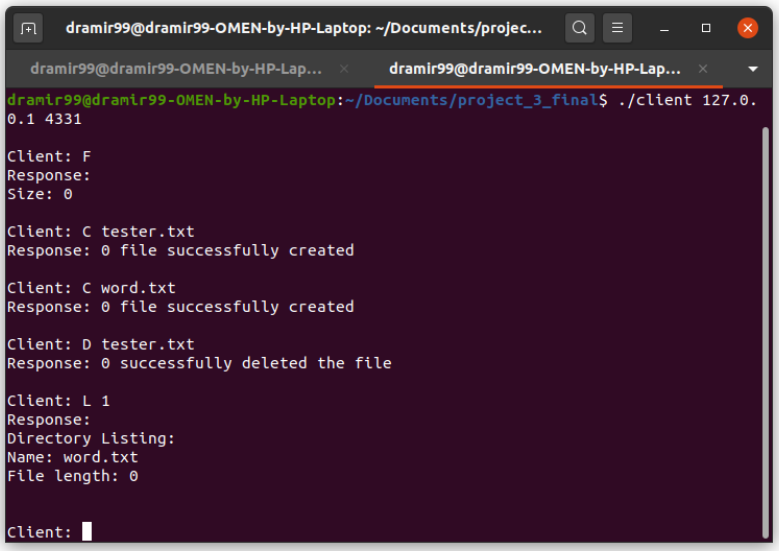
Then you can create a file in the fileSystem by typing ‘C <name of file>’ for this example i put in ‘C tester.txt’ and ‘C word.txt’



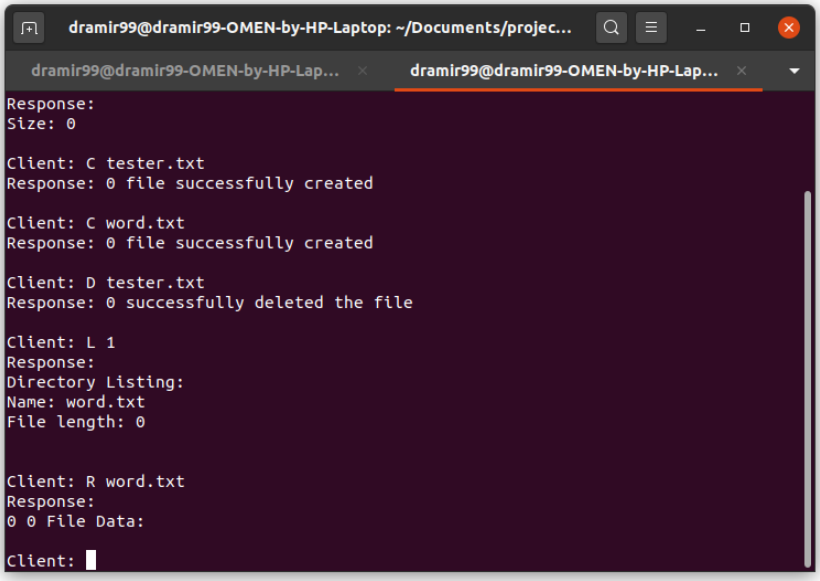
Then you can delete a file in the fileSystem by typing ‘D <name of file>’ for this example i put in ‘D tester.txt’



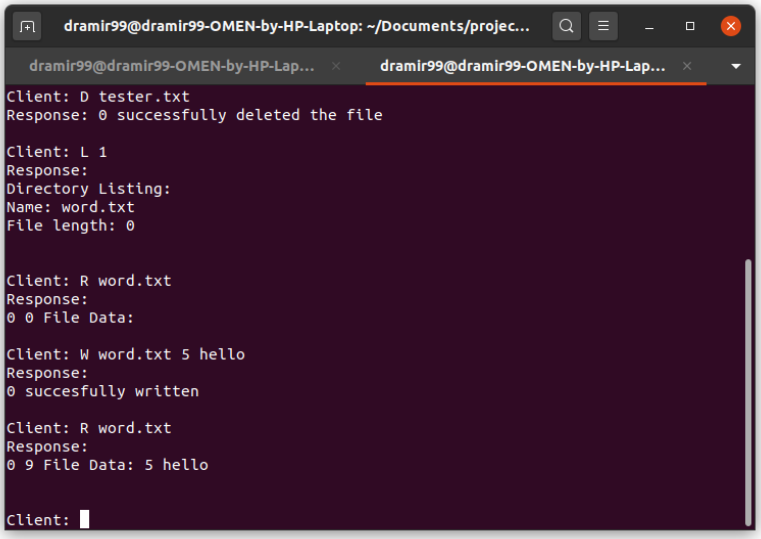
Then you can see the directory in the fileSystem by typing ‘L 1’ or ‘L 0’ depending if you want to see the name of the files or name of the files and the size of the file



Then you can read a file in the fileSystem by typing ‘R <name of file>’ for this example i put in ‘R word.txt’

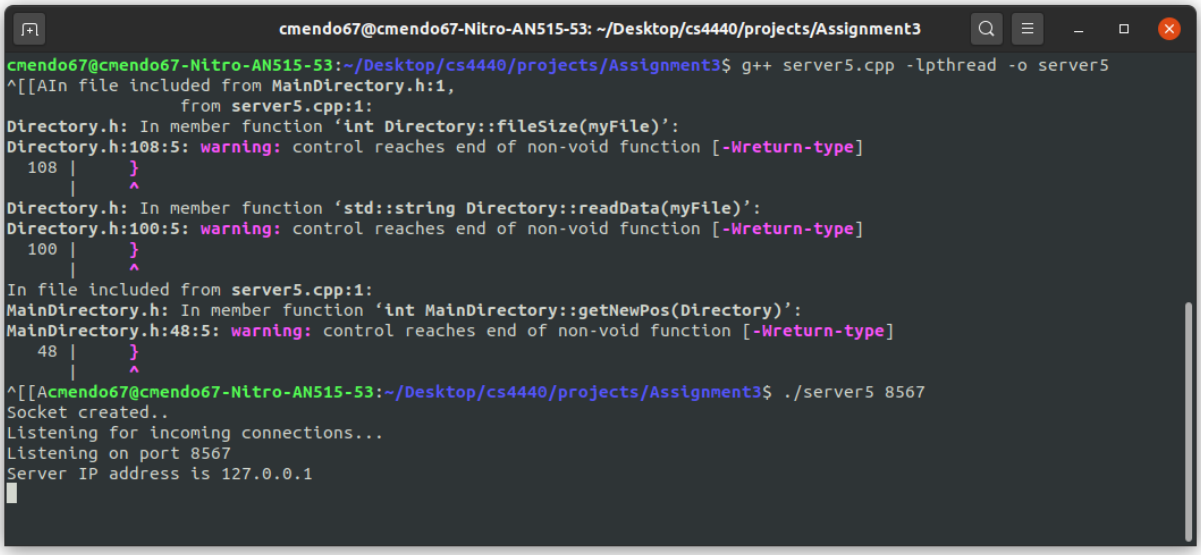


Then you can write to a file in the fileSystem by typing ‘W <name of file> <size of the text> <text>’ for this example i put in ‘W word.txt 5 hello’

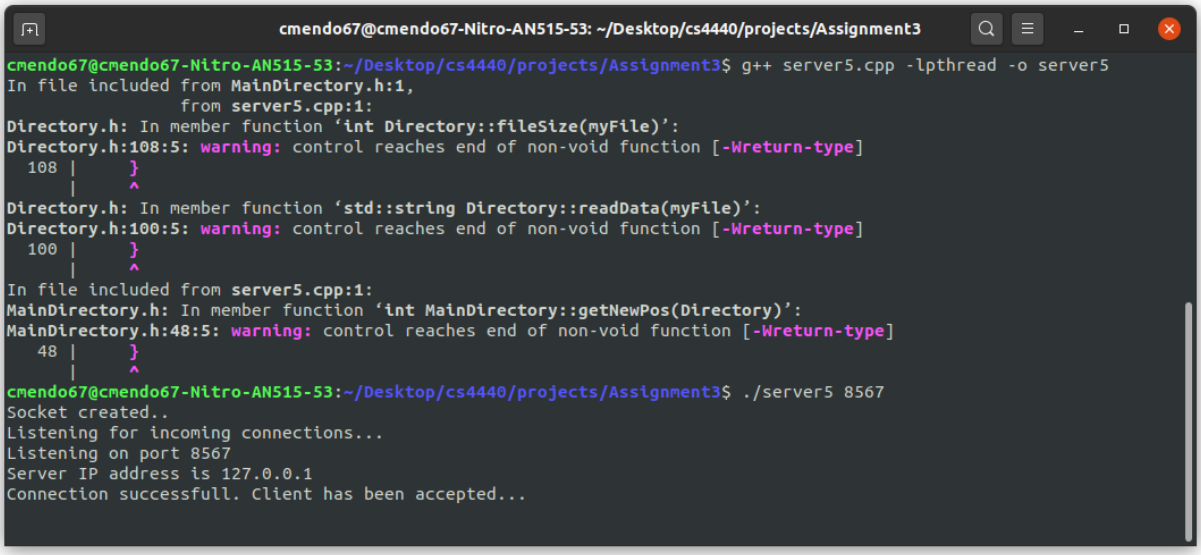


5. **Directory Structure**

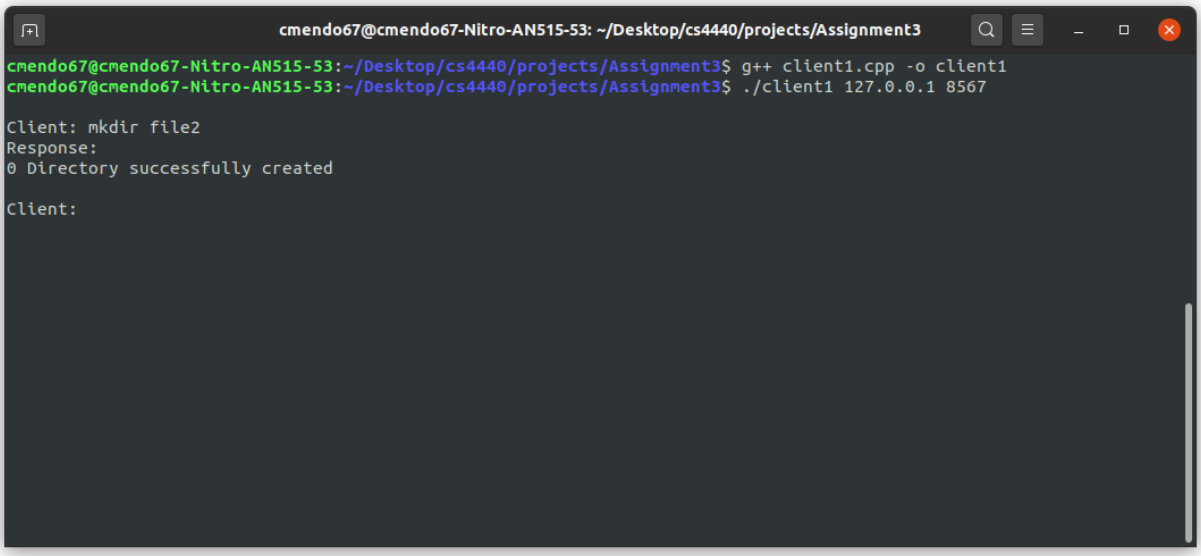
First run and compile server 5.

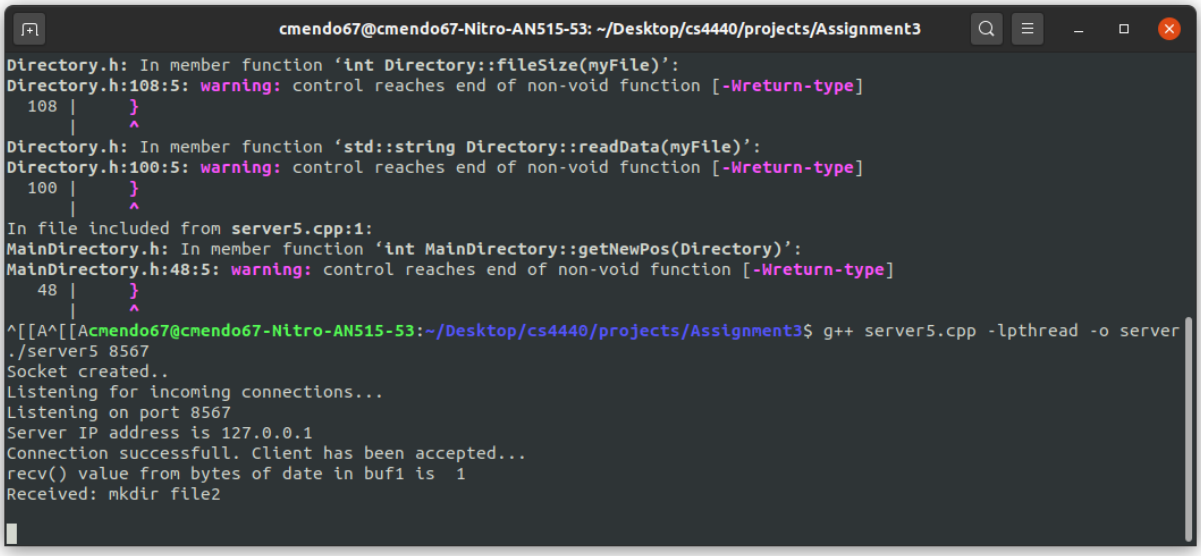


Now, compile and run client1 to connect to server.

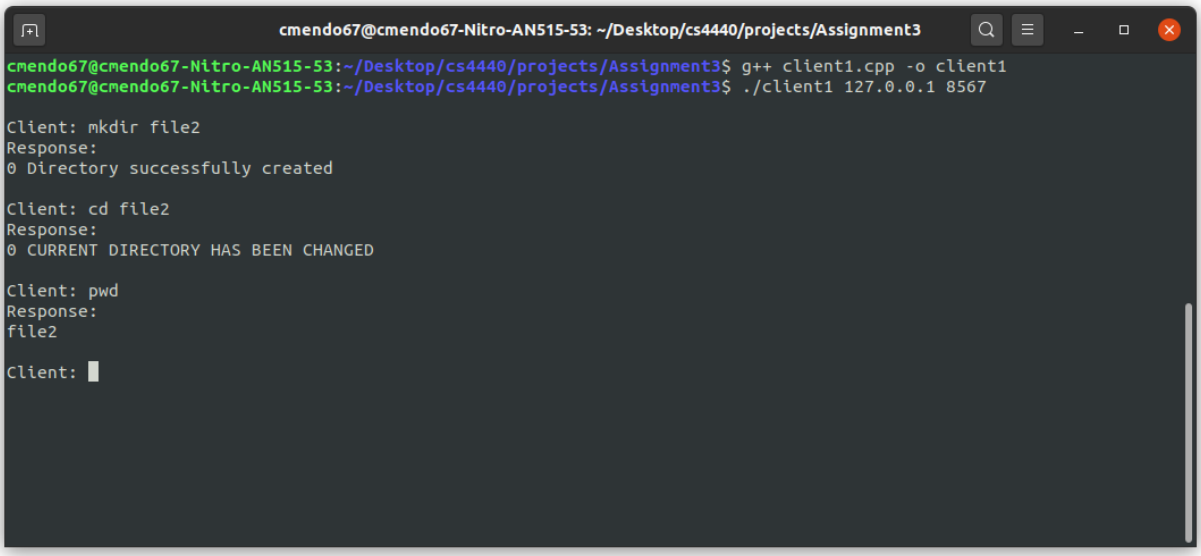


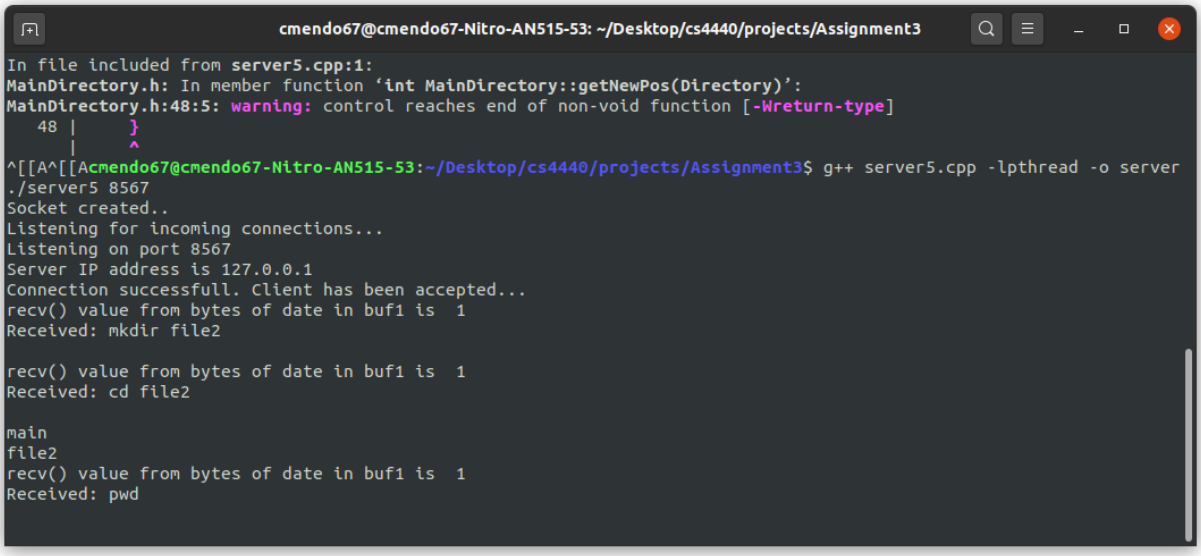
Now, if you enter mdir - dirname : create a directory of name dirname.



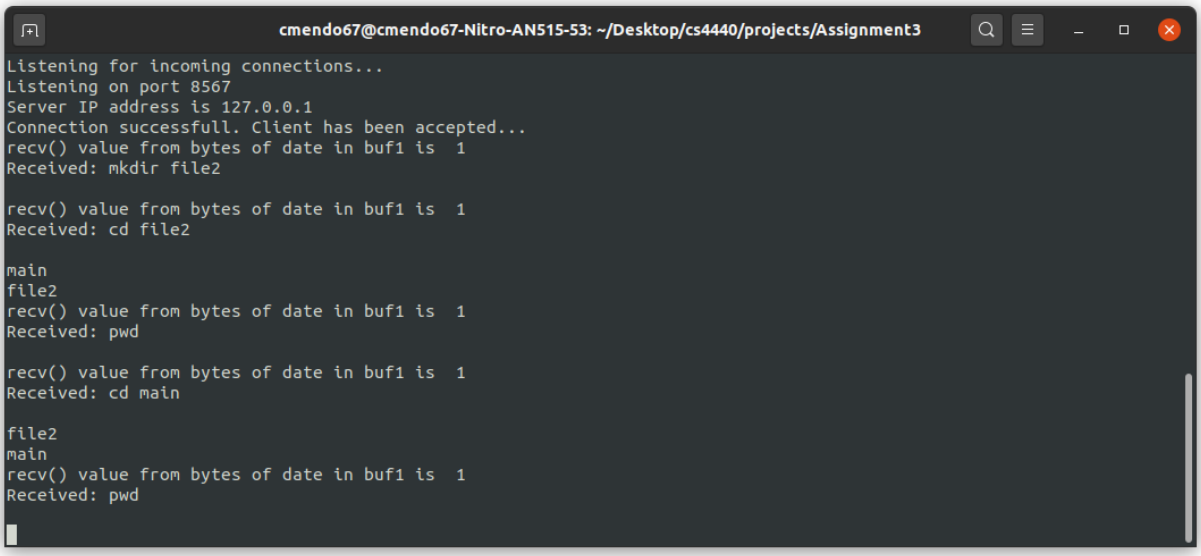


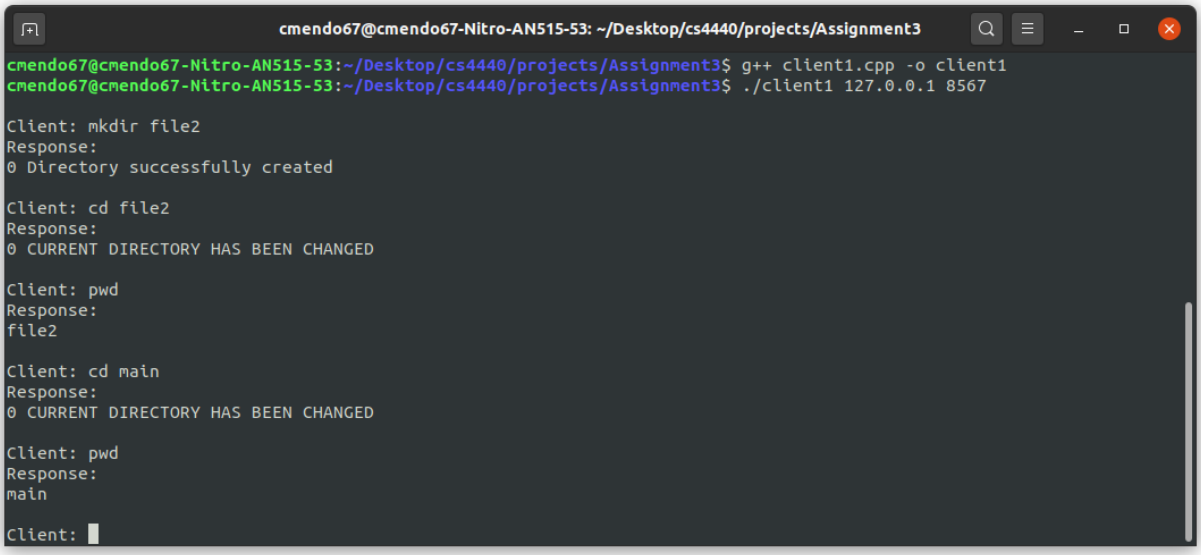
Now, if you enter cd - dirname : change current working directory to dirname



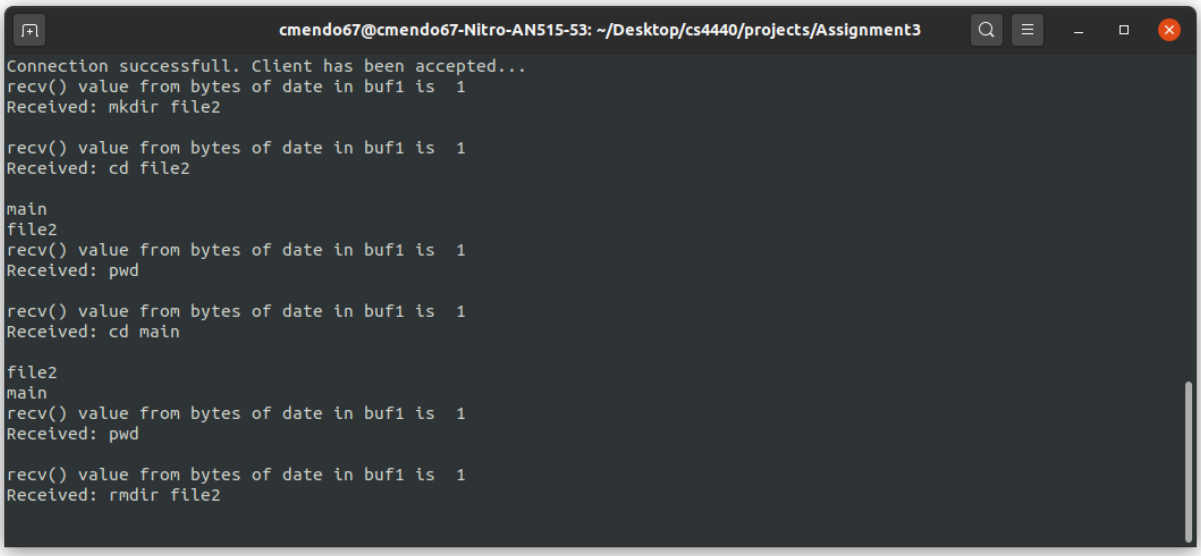


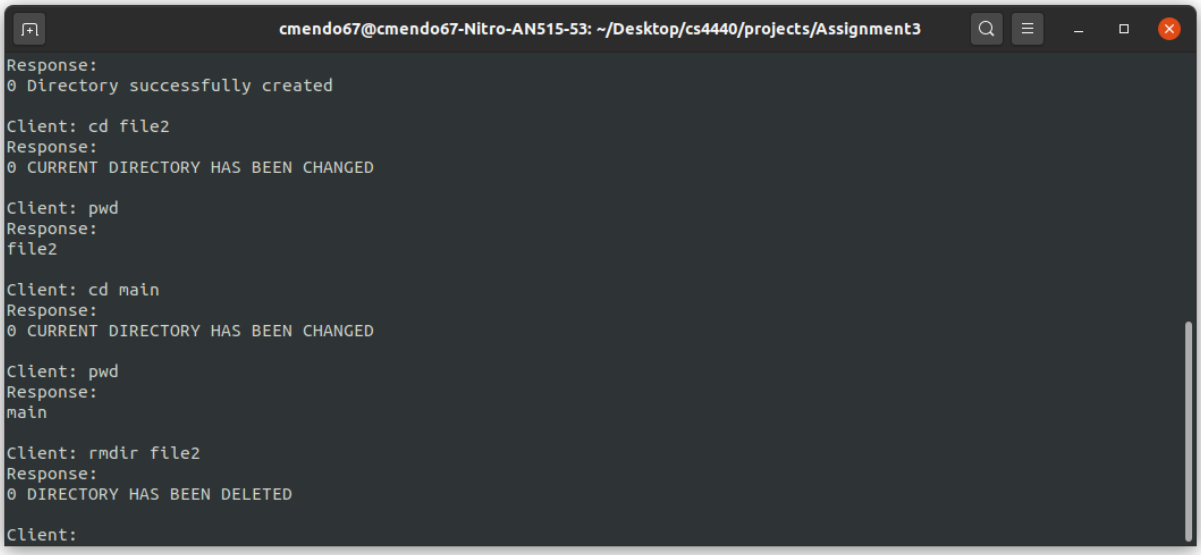
Now if you enter pwd, : print the working directory name.





Now, if you enter rmdir - dirname: remove the directory given. Throw an error if it is not present.


cmendo67@cmendo67-Nitro-AN515-53: ~/Desktop/cs4440/projects/Assignment3
Connection successfull. Client has been accepted...
recv() value from bytes of date in buf1 is 1
Received: mkdir file2
recv() value from bytes of date in buf1 is 1
Received: cd file2
main
file2
recv() value from bytes of date in buf1 is 1
Received: pwd
recv() value from bytes of date in buf1 is 1
Received: cd main
file2
main
recv() value from bytes of date in buf1 is 1
Received: pwd
recv() value from bytes of date in buf1 is 1
Received: rmdir file2


cmendo67@cmendo67-Nitro-AN515-53: ~/Desktop/cs4440/projects/Assignment3
Response:
0 Directory successfully created
Client: cd file2
Response:
0 CURRENT DIRECTORY HAS BEEN CHANGED
Client: pwd
Response:
file2
Client: cd main
Response:
0 CURRENT DIRECTORY HAS BEEN CHANGED
Client: pwd
Response:
main
Client: rmdir file2
Response:
0 DIRECTORY HAS BEEN DELETED
Client:

Technical Design

3.

For the design disk we need the foundation of question 1 since we need to talk to the server and client, the design disk was designed using the fopen in order to read and write to the file then using the given formula to create the size of the cylinder and sector. We initilize the char with 128 bytes and the user will be able to manipulate the data depending on what the user does

4 and 5

For the algorithm we used was a vector in order to hold the list of files. We created 3 classes which were DIrectory and Main Directory and myFile each was used for the foundation in order to visualize the linux system.myFile was used to replicate a file. The Director was used to hold multiple files. The Main Directory was used to hold multiple Directories. We used the vector algorithm since we need to be able to get to each element.