

ROHIT GARG (2018A7PS0193G)

COMPUTER NETWORKS

CS F303

LAB5

On running ./server, the server will accept its port number as the input parameter.
(5 marks)



```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
```



```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening....
```

Upon running the client executable(./client), it asks for:
Server IP Address,
Server Port Number and then the
filename to be requested:

After taking the address and port, the client connects to the Server and
connection is established successfully.
Here, Port number used = 1234
Server IP Address used = 127.0.0.1

```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening...
Handling Client @ 127.0.0.1

rohit@rohit:~/3-2/#CN/Labs/Lab5/Client
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:

```

Note: Use Port Number > 1024 because port numbers below those are reserved for OS usages and may result in error

When a filename is entered that exists in the server's directory, the server retrieves the first 10 bytes from the file and returns the same to the client, which then creates a file (with the given filename) with the received message in it:

Entering filename = networkingtrends.txt

```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening...
Handling Client @ 127.0.0.1

rohit@rohit:~/3-2/#CN/Labs/Lab5/Client
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
Networkingtrends.txt

```

```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening...
Handling Client @ 127.0.0.1
Requested data sent to client successfully

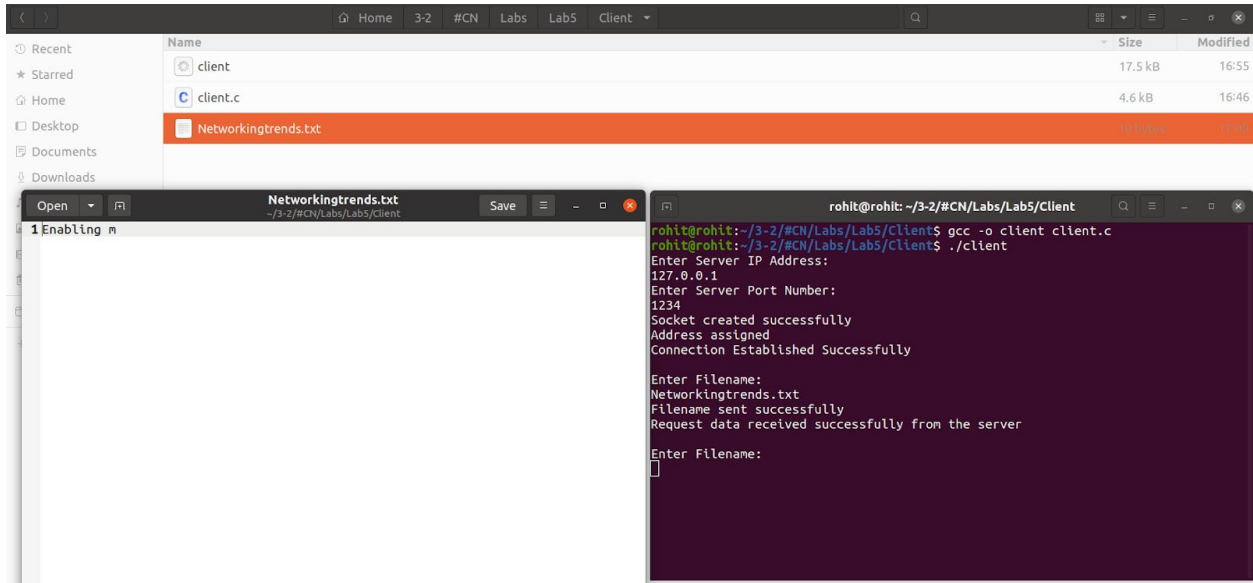
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
Networkingtrends.txt
Filename sent successfully
Request data received successfully from the server

Enter Filename:

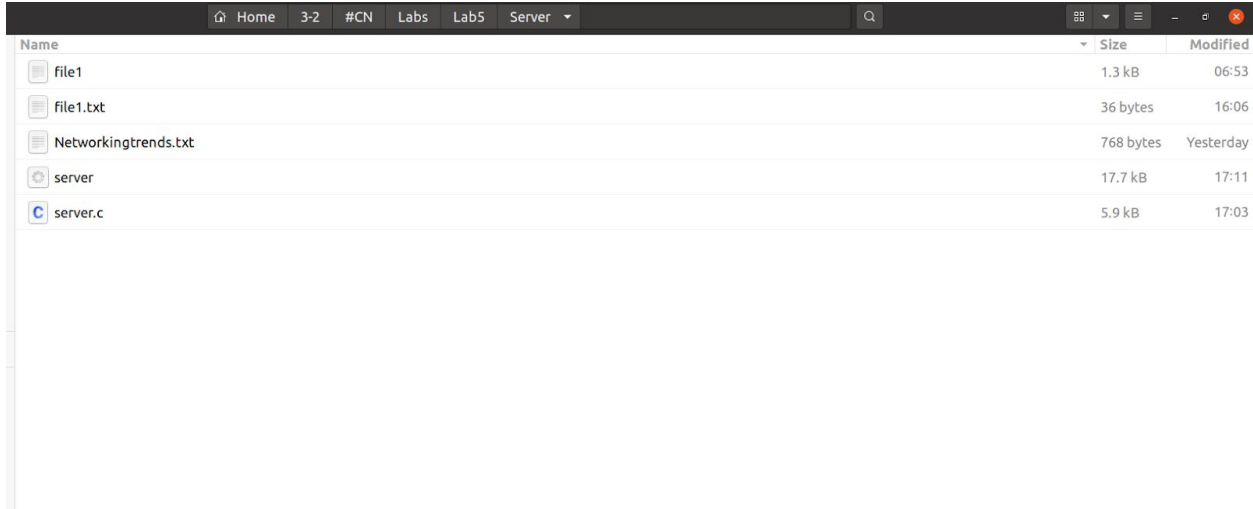
```

Home 3-2 #CN Labs Lab5 Client				
Name	Size	Modified		
client	17.5 kB	16:55		
client.c	4.6 kB	16:46		
Networkingtrends.txt	10 bytes	17:00		



When a filename is requested that does not exist in the server's directory, the server returns an empty string to the client and the client shows that no such file was present on the server and makes an empty file, with the entered filename:

We will try to request filename = budget since it does not exist at the server side.



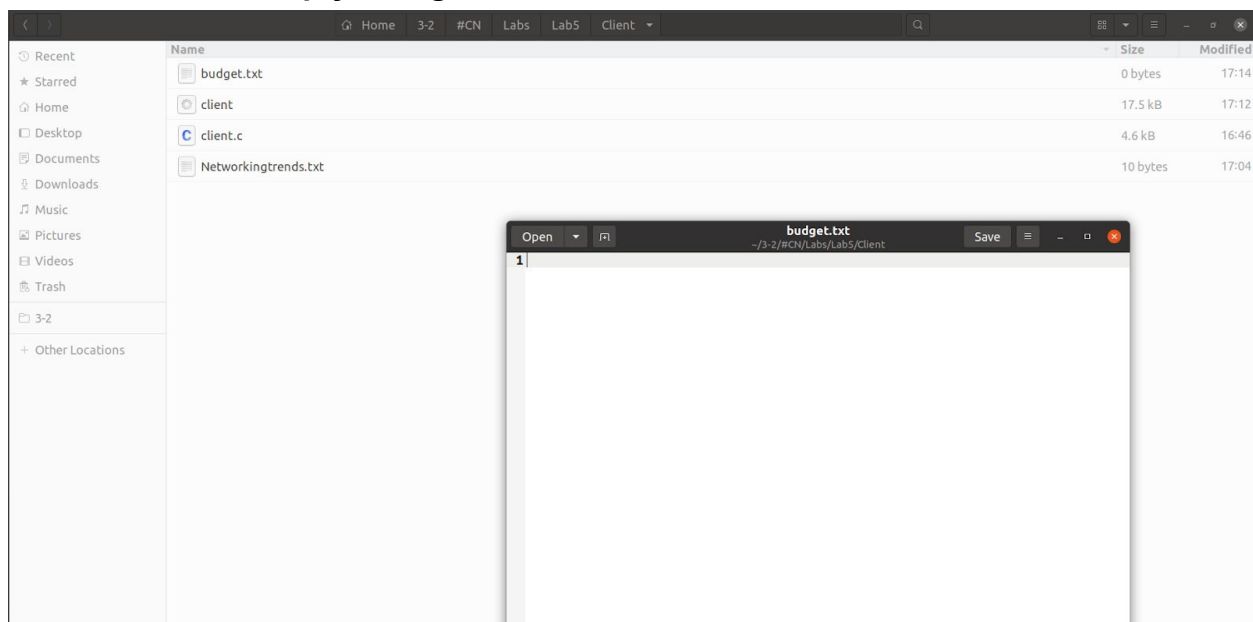
```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening....
Handling Client @ 127.0.0.1
Requested File does not exist at server side
Sending empty string to client...
Requested data sent to client successfully
█

rohit@rohit: ~/3-2/#CN/Labs/Lab5/Client
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
budget.txt
Filename sent successfully
File not found at server side
Server returned empty string
Request data received successfully from the server

Enter Filename:
█
```

server sends an empty string to client



The server repeatedly waits for a filename and a client can send multiple requests to the server.

When exitClient as filename is passed on the client terminal, the client program exits while server does not exit and waits for a new client

```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening...
Handling Client @ 127.0.0.1
Requested data sent to client successfully

Requested data sent to client successfully
Requested File does not exist at server side
Sending empty string to client...
Requested data sent to client successfully

rohit@rohit:~/3-2/#CN/Labs/Lab5/Client
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
file1.txt
Filename sent successfully
Request data received successfully from the server

Enter Filename:
Networkingtrends.txt
Filename sent successfully
Request data received successfully from the server

Enter Filename:
file2
Filename sent successfully
File not found at server side
Server returned empty string
Request data received successfully from the server

Enter Filename:
exitClient
Filename sent successfully
Client Exiting...rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$
```

When the above client left, a new client joins the server.
A new client joins the server at same IP and port number. This is denoted by
Handling Client @.... statement

```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening...
Handling Client @ 127.0.0.1
Requested data sent to client successfully

Requested data sent to client successfully
Requested File does not exist at server side
Sending empty string to client...
Requested data sent to client successfully

Handling Client @ 127.0.0.1
Requested File does not exist at server side
Sending empty string to client...
Requested data sent to client successfully

rohit@rohit:~/3-2/#CN/Labs/Lab5/Client
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
file1.txt
Filename sent successfully
Request data received successfully from the server

Enter Filename:
Networkingtrends.txt
Filename sent successfully
Request data received successfully from the server

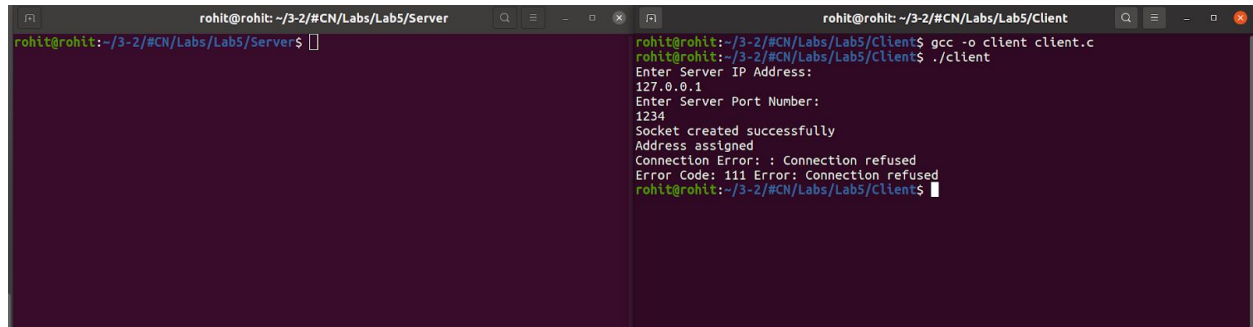
Enter Filename:
file2
Filename sent successfully
File not found at server side
Server returned empty string
Request data received successfully from the server

Enter Filename:
exitClient
Filename sent successfully
Client Exiting...rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
file3
Filename sent successfully
File not found at server side
Server returned empty string
Request data received successfully from the server

Enter Filename:
```

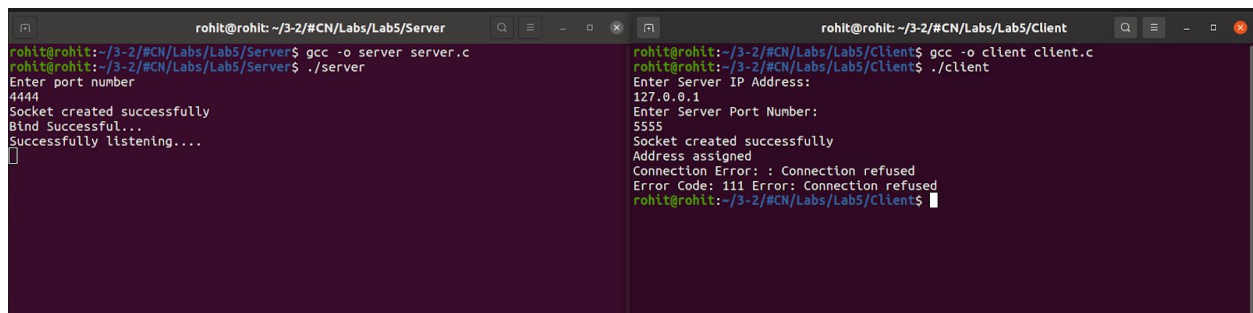
If the server is not running/unreachable or the client enters an address or port the server is not listening on, an error code given by errno along with statement is displayed showing that connection to server failed/refused:



The screenshot shows two terminal windows. The left window is the server terminal, and the right window is the client terminal. The client terminal shows the following output:

```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Error: : Connection refused
Error Code: 111 Error: Connection refused
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$
```

If different port numbers at client and server terminals are given then also connection will be refused along with error code displayed.



The screenshot shows two terminal windows. The left window is the server terminal, and the right window is the client terminal. The server terminal shows the following output:

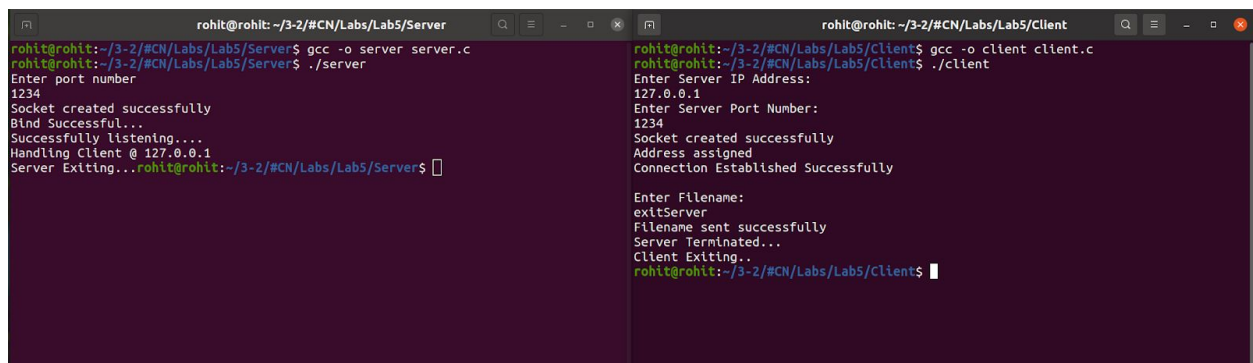
```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
4444
Socket created successfully
Bind Successful...
Successfully listening...

```

The client terminal shows the following output:

```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
5555
Socket created successfully
Address assigned
Connection Error: : Connection refused
Error Code: 111 Error: Connection refused
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$
```

For exiting the server, pass exitServer as filename at client terminal.



The screenshot shows two terminal windows. The left window is the server terminal, and the right window is the client terminal. The server terminal shows the following output:

```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening...
Handling Client @ 127.0.0.1
Server Exiting...rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$
```

The client terminal shows the following output:

```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ gcc -o client client.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
exitServer
Filename sent successfully
Server Terminated...
Client Exiting...
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$
```


Note: server can also be exited using Ctrl + c. In this case, client will be notified when it queries the server next time for any file data.

```
rohit@rohit: ~/3-2/#CN/Labs/Lab5/Server
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening....
Handling Client @ 127.0.0.1
^C
Server Terminated
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$
```

When a client tries to send data to the server, it is notified that the server exited so it also exits along with it.

```
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server  rohit@rohit:~/3-2/#CN/Labs/Lab5/Client
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ gcc -o server server.c
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$ ./server
Enter port number
1234
Socket created successfully
Bind Successful...
Successfully listening....
Handling Client @ 127.0.0.1
^C
Server Terminated
rohit@rohit:~/3-2/#CN/Labs/Lab5/Server$

rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$ ./client
Enter Server IP Address:
127.0.0.1
Enter Server Port Number:
1234
Socket created successfully
Address assigned
Connection Established Successfully

Enter Filename:
hello
Filename sent successfully
Server Terminated...
Client Exiting...
rohit@rohit:~/3-2/#CN/Labs/Lab5/Client$
```