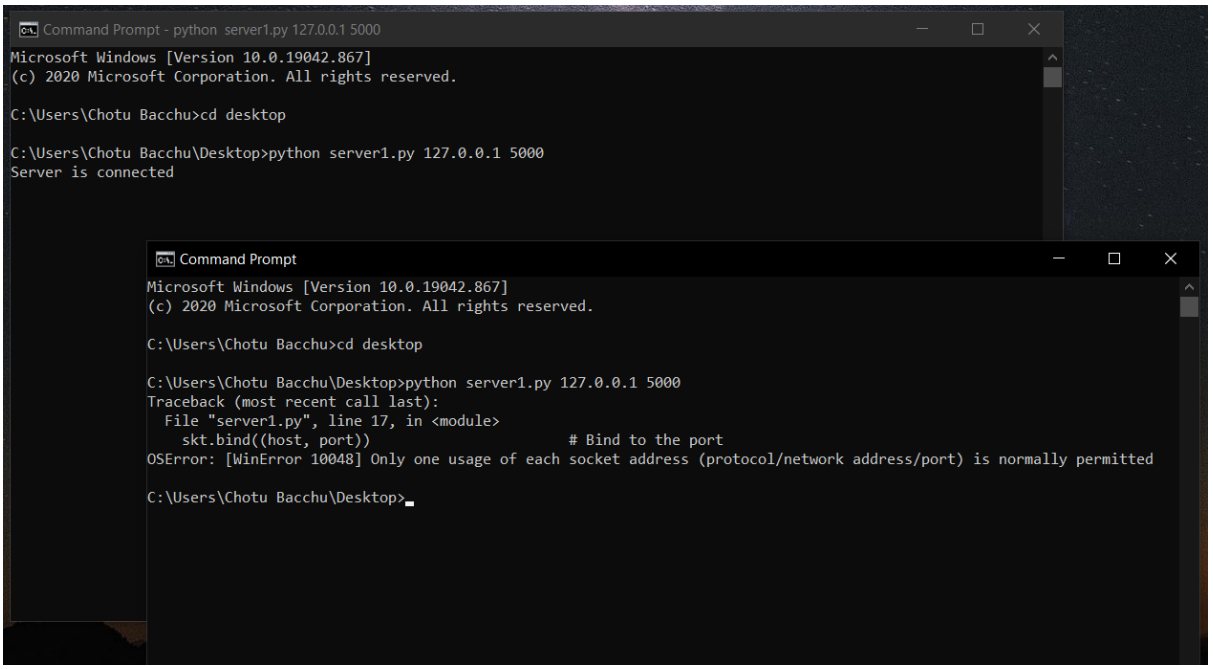
PRACHI KALANTRI
BT18CSE136

CN ASSIGNMENT 3

General instructions:

The server program is started on a custom port where server ip and port is provided on the command line. If the port is already occupied the server throws an error.



1. First start the server with command “python server[no].py [server ip] [port number]”.
2. Start the client with command “python client.py [server ip] [port number]” . The server ip and port number must be the same for both client and server.
3. For a multi threaded server you can start a maximum 5 clients at a time.

Server 1

After starting server1, a message will be displayed to ensure that server has started properly.

```
C:\Users\Chotu Bacchu\Desktop>python server1.py 127.0.0.1 5000
Server is connected
```

After this start the client program. As the client gets connected to the server , a message will be displayed .

```
C:\Users\Chotu Bacchu\Desktop>python client.py 127.0.0.1 5000
connected to server
enter the expression:
```

```
C:\Users\Chotu Bacchu\Desktop>python server1.py 127.0.0.1 5000
Server is connected
server is connected to  ('127.0.0.1', 50549)
```

When you enter the expression in the client side, the query will be sent to the server and the server will send back the appropriate response to the client.

Client:

```
C:\Users\Chotu Bacchu\Desktop>
C:\Users\Chotu Bacchu\Desktop>python client.py 127.0.0.1 5000
connected to server
enter the expression: 2+5
evaluating your answer...
answer recieved from the server is  7
you are still connected to the server

enter the expression: 5/0
evaluating your answer...
division by zero is not allowed
you are still connected to the server


enter the expression: 5+
evaluating your answer...
enter a valid expression
you are still connected to the server
```

Server:

```
C:\Users\Chotu Bacchu\Desktop>python server1.py 127.0.0.1 5000
Server is connected
server is connected to  ('127.0.0.1', 50584)
expression recieved from client :  2+5
evaluating the answer....
answer sent..!


expression recieved from client :  5/0
evaluating the answer....
expression recieved from client :  5+
evaluating the answer....
```

If we try to connect another client when the first client is already connected, the server will not accept any query given by the other client. Once the first client closes the connection, then the server accepts the connection with the other client.

Server 2

This server is a multi threaded server. For the given code, it can simultaneously handle maximum 5 clients.
When server 2 starts properly, it will display a message.

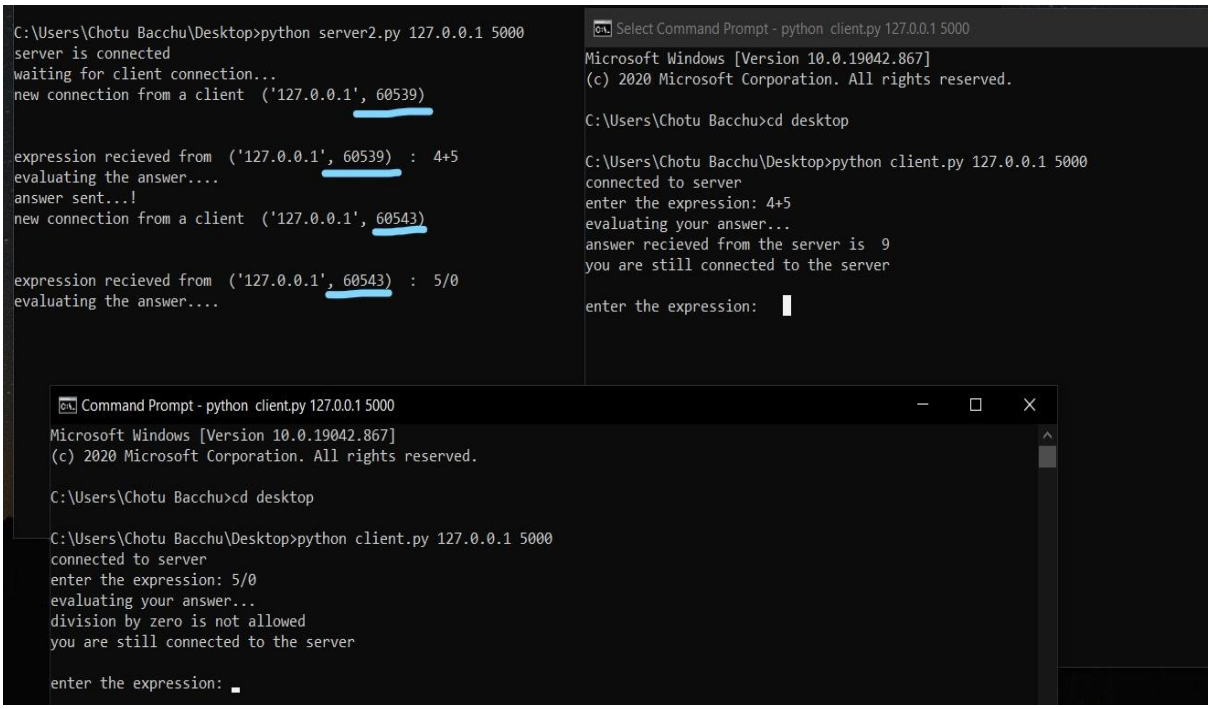
```
C:\Users\Chotu Bacchu\Desktop>python server2.py 127.0.0.1 5000
server is connected
waiting for client connection...
```

Whenever a new client is connected, the server will display a message with the address of the client.

```
C:\Users\Chotu Bacchu\Desktop>python client.py 127.0.0.1 5000
connected to server
enter the expression:

C:\Users\Chotu Bacchu\Desktop>python server2.py 127.0.0.1 5000
server is connected
waiting for client connection...
new connection from a client  ('127.0.0.1', 52385)
```

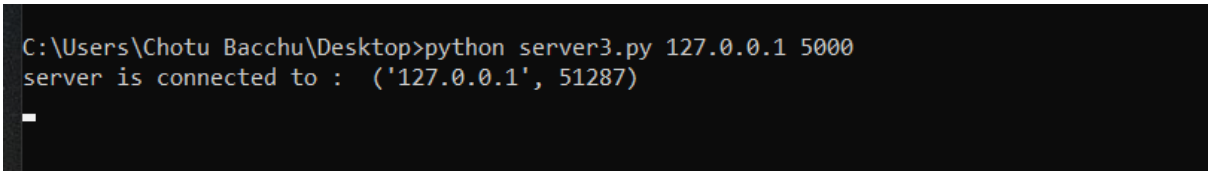
When a new client is connected, it will display its address. The server while handling 2 or more different clients will look like this.



Server 3

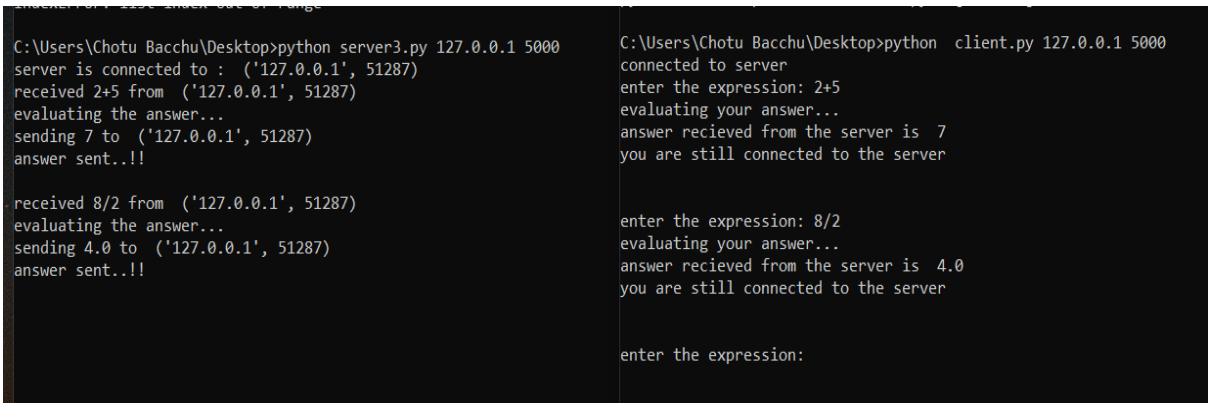
Server 3 is a single process server that can handle, for this code, 3 clients concurrently.

After successfully establishing the connection with any client, a message will be displayed.

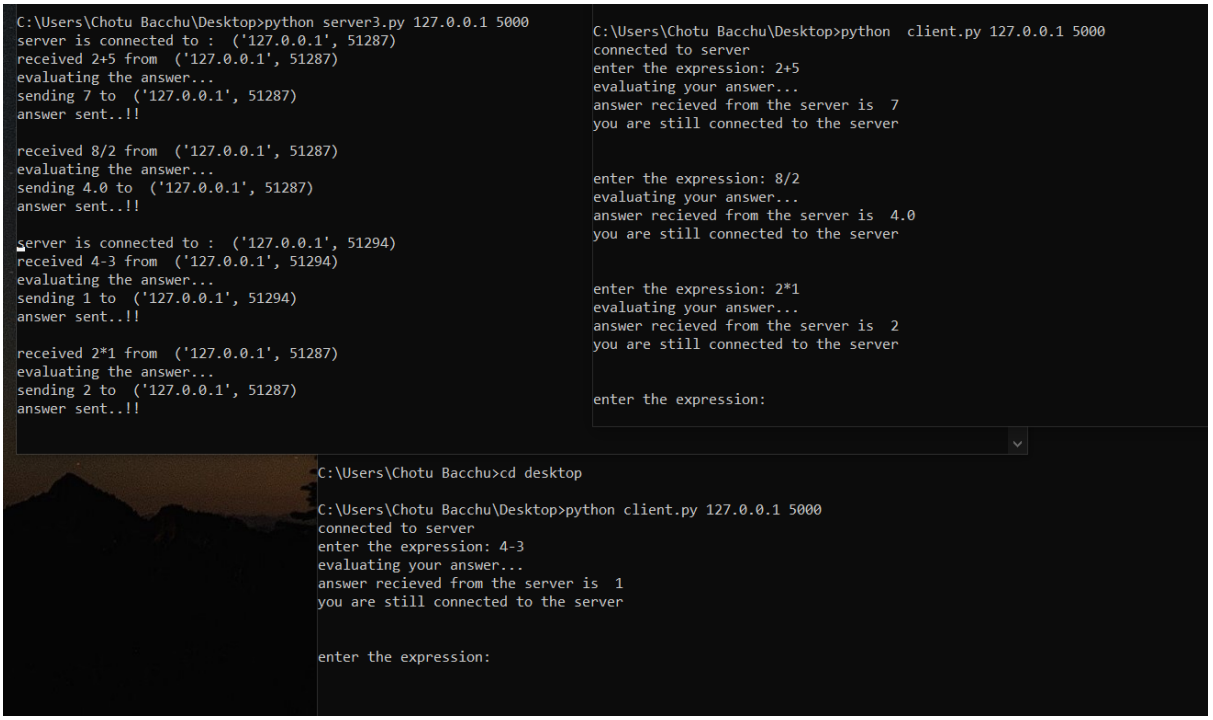


When the client passes any expression, the server will evaluate the answer and send it to the client. On the server side we will be able to see the expression received and the answer sent.

Execution for a single client is shown in the screenshot given below.



For multiple clients, the server will store the address of the client and send the answer to the same client. On the server side we can see all the queries received from all the addresses.
Execution of multiple (2 in this case) clients can be seen in the below screenshot.



Server 4

This is an echo server i.e it will return the same message that the client has sent. Working is the same as server 3.
The execution is shown in the below screenshot.