Blake Rusteberg

Programming Assignment #01: Basic client/server system

CS 447: Networking and Data Communication

Introduction

This project was intended to give us a better understanding on client and server communication. I started this project in Python and learned the basic functionality of python's socket programming library. This library was very straight forward in communicating how a server and a client interact with each other. The difficult part was understanding when bytes were being sent from the server to client and vice versa. I would say about 80-90 percent of project was error checking every possible outcome the client could check. For example, if the client typed SPHERE then they wouldn't be able to type AREA next because it is a CUBE command.

Objectives

- Establishing a connection between the server and client using socket programming
- Check to see if the server responds after the client types something
- Check to make sure the client says HELO to the server first before doing anything else
- Make a large switch statement in a while loop so when the user types a command it is either valid or invalid.
- Create an else if statement for each shape along with their corresponding commands
- Error check all these else if statements to make sure there is no way to break the program.
- Multithreading
- Clean up code
- Final Error checking tests
- Turn in

Design Choices

I stuck to a very simple design choice when coding this project. I wanted to make sure my code was understandable to someone who has never seen it before, and that the UI made sense as the client continued through the program. When the client interacted with one of the

different shapes, they had to complete a valid calculation with one of their calculation methods before being able to use other shapes. Example:

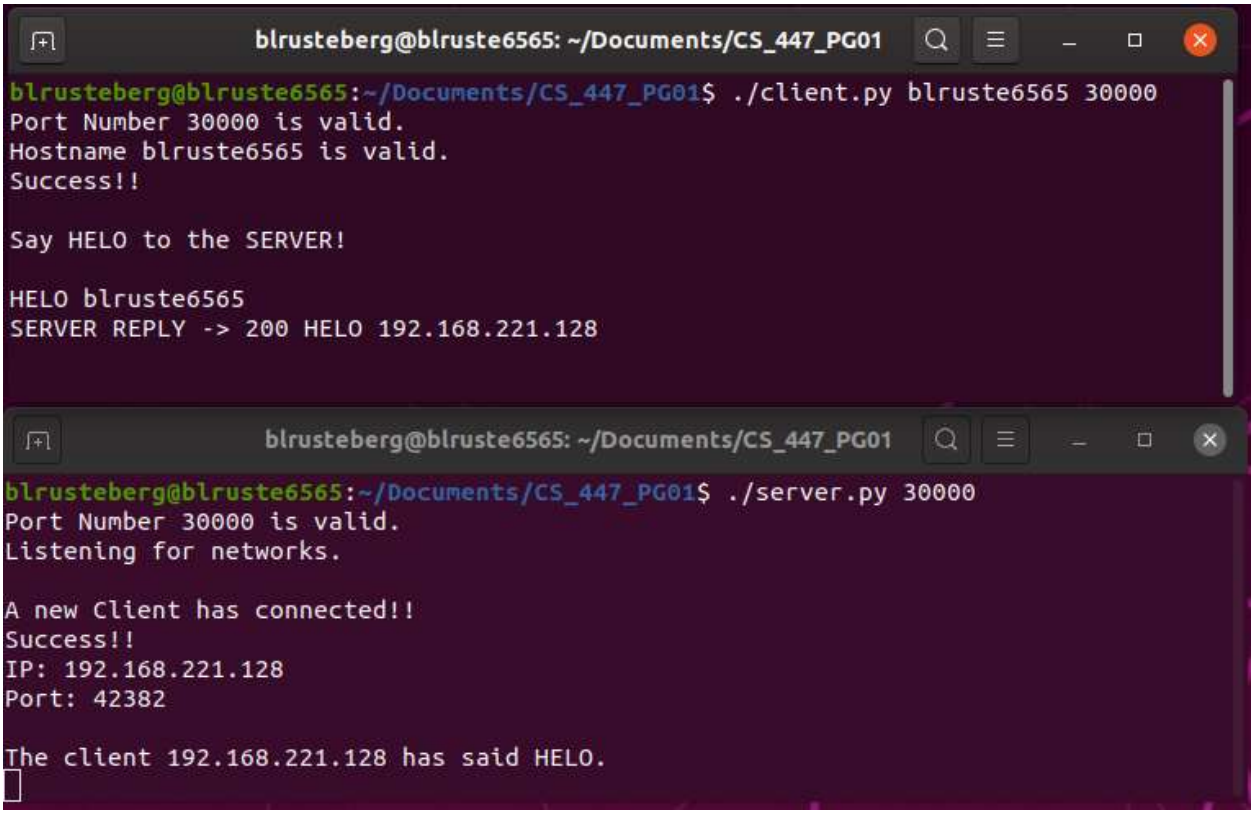
```
SPHERE
SERVER REPLY -> 220 SPHERE ready!

VOL 4 2
SERVER REPLY -> 501 Syntax Error in Parameters: VOL <r>

CUBE
SERVER REPLY -> 500 Syntax Error, Command 'CUBE' Unrecognized!
Use the 'HELP' Command for the list of Commands.

CONE
SERVER REPLY -> 500 Syntax Error, Command 'CONE' Unrecognized!
Use the 'HELP' Command for the list of Commands.

VOL 2
SERVER REPLY -> 250 33.510321638291124

CONE
SERVER REPLY -> 230 CONE ready!
```

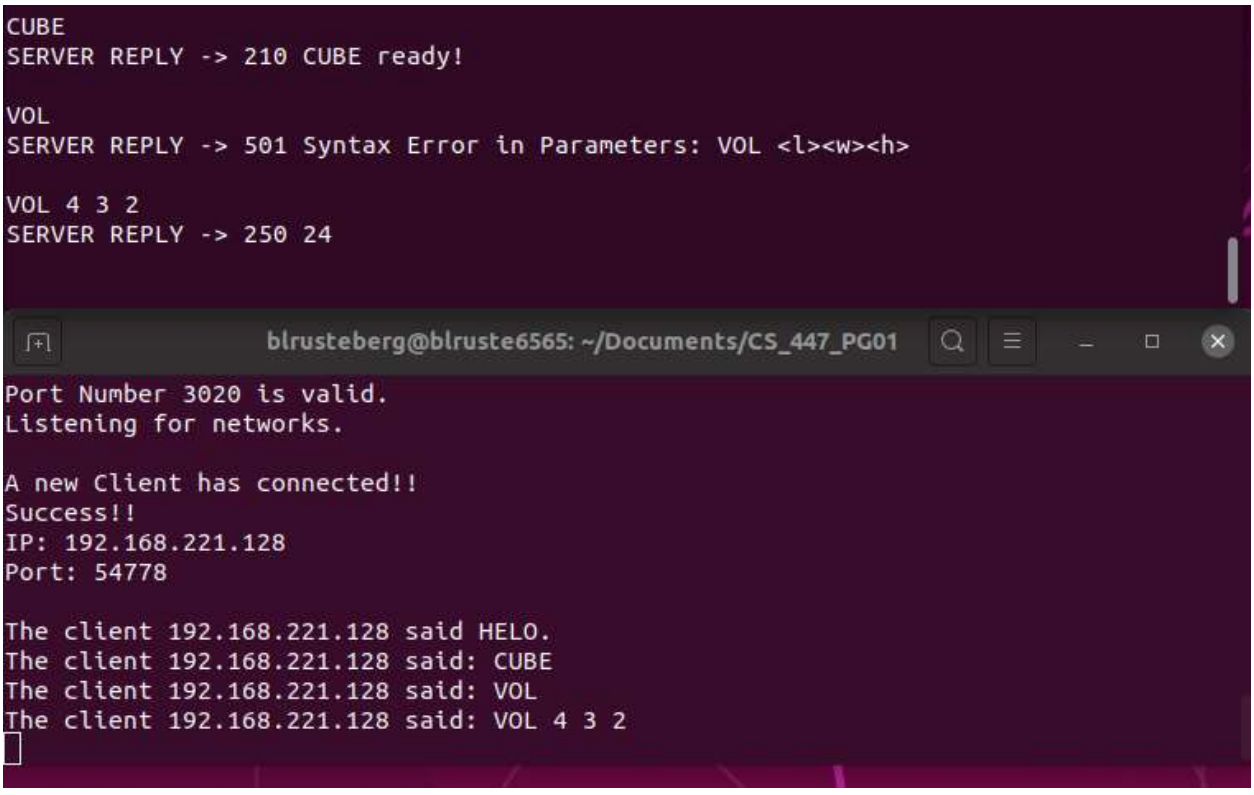
This took some time to code because there were many different error checks for each Command.

The HELP command in my code gives a good description of what each command does and will help the client if they get stuck.

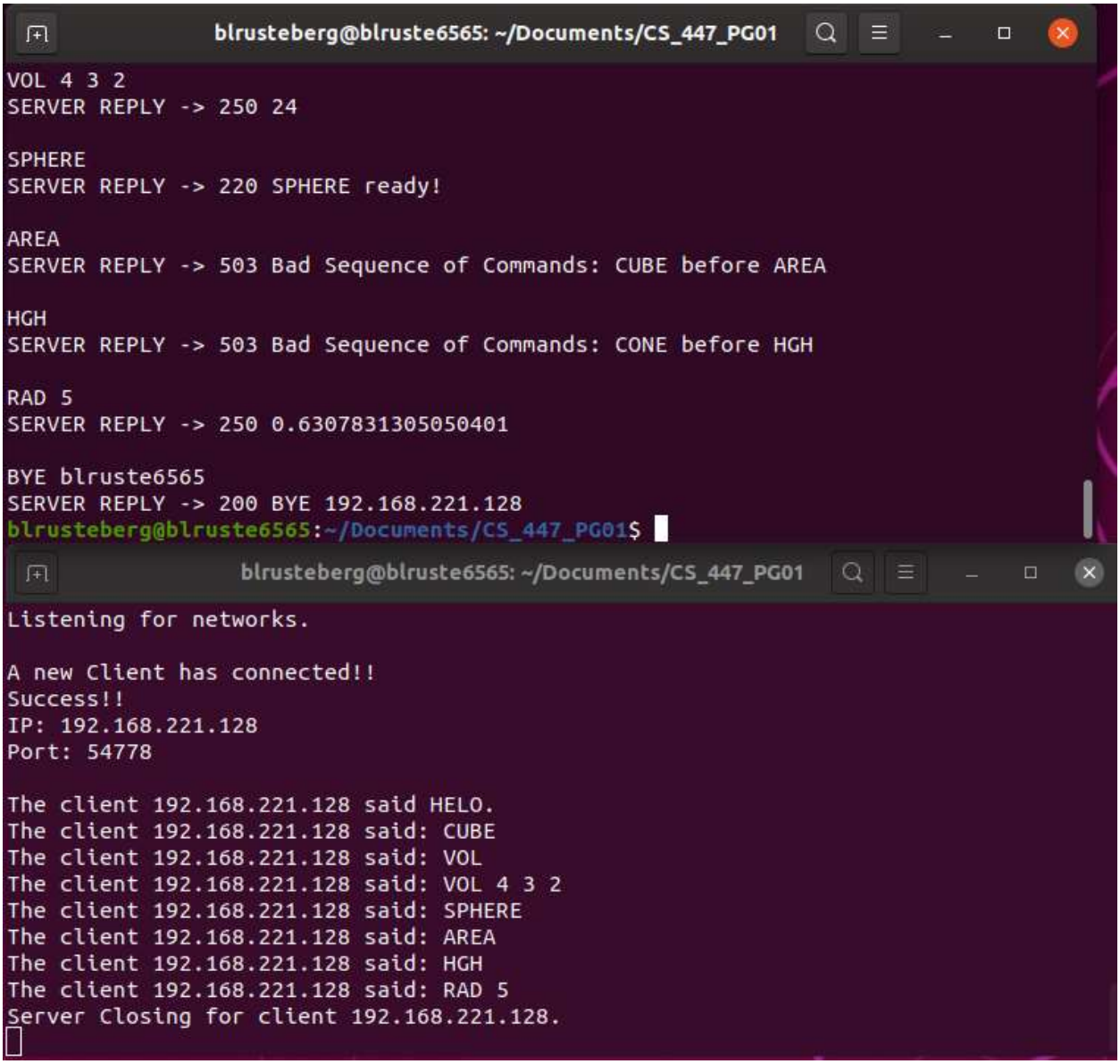
```
HELP
SERVER REPLY -> 200 These are the list of commands used on this server.
HELO   - The first command that is used to greet the server. Unrecognized after the first greeting. <
HELP   - A list of commands the server interacts with.
CUBE   - Used to calculate the area calulation AREA <l><w><h> and the volume calulation VOL <l><w><h>.
SPHERE - Used to calculate the volume calulation VOL <r> and the radius calulation RAD <A>.
CONE   - Used to calulate the height calulation HGH <A><r> and volume calulation VOL <r><h>.
AREA   - Can only be used after the CUBE command. Formula: 2(lw + hw + hl).
VOL    - Can only be used after the CUBE, SPHERE, or CONE command, CUBE: lwh, SPHERE: (4/3)πr^3, CONE: πr√(r^2 + h^2).
RAD    - Can only be used after the SPHERE command, Formula: (1/2)√(A)/(π).
HGH    - Can only be used after the CONE command. Formula: (3/A)/(πr^2).
BYE    - Exit the server gracefully... BYE <hostname>
Note - Client must enter a valid calulation command to get another shape ready. EX: CONE then HGH. <r>
```

Program Run through

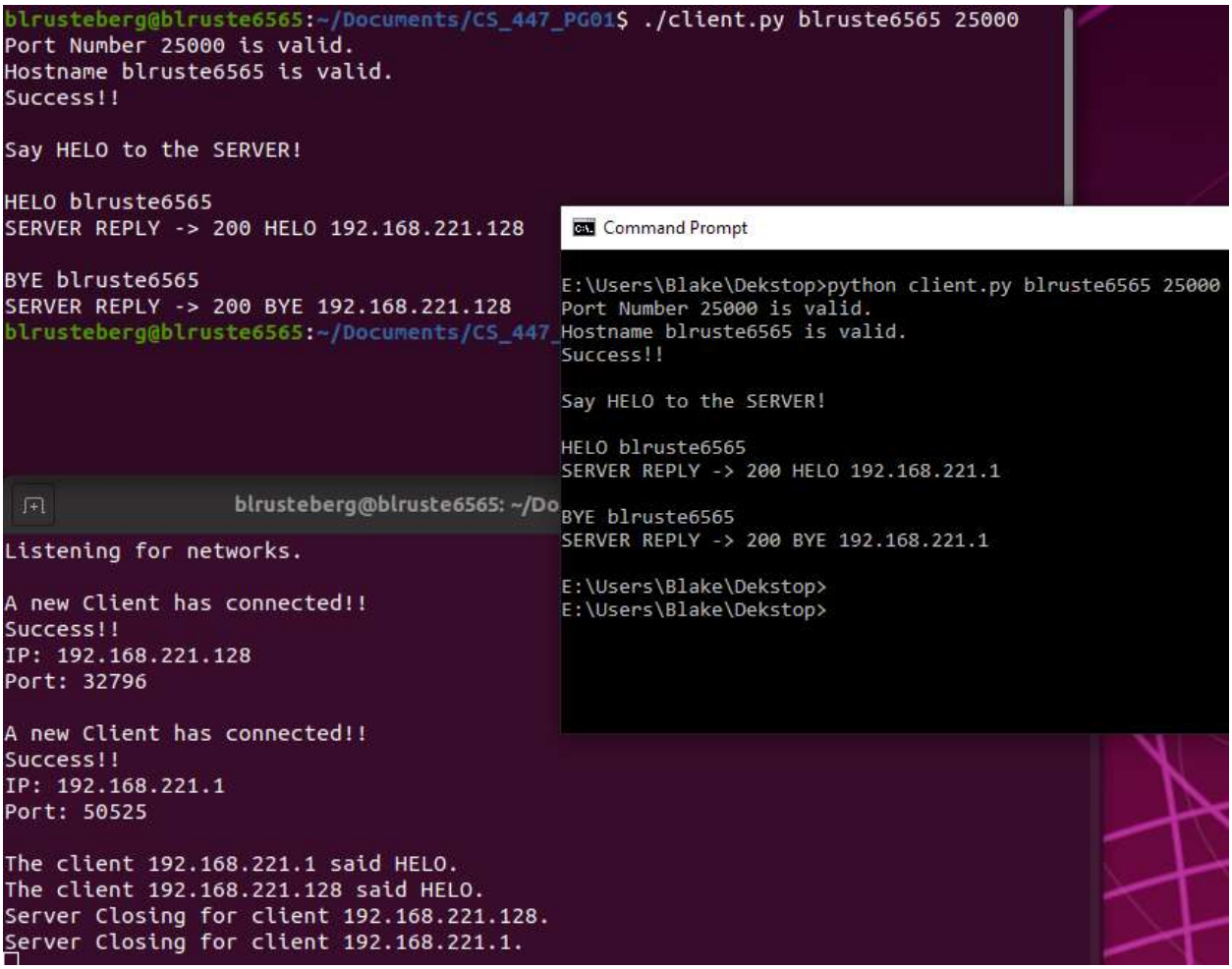




Error checks and BYE message



Multiple clients



Summary

This project took no time at all to set up in terms of having the client and server communicate. The difficult part was understanding what the client is going to type in and making sure the server has an answer for it. I encountered many issues the first time running the project. If the client typed a command sometimes other error message would come up not related to that command. This made me realize I had to completely erase my while loop that connected the server and client together and start from scratch. So, I proceeded to draw it out on a piece of paper. When doing these projects having a good plan from the beginning is key, because in the end you will have to restart all your code. Overall this project increased my knowledge of programming and help me become a more efficient programmer.