

# Programming Assignment 1: Socket Calculator

In this programming assignment, you will implement client-server socket communication as we discussed in the lecture. Both client and server will be run on **eustis.eecs.ucf.edu** (by now you should know how to login Eustis server). Therefore, you would need two processes, namely a client process and a server process, to realize the socket communication. You can use C/C++, Java, and Python to program this project. Eustis supports all these programming languages.

In this assignment, you will implement a basic calculator. The server will run first. The client program will repeatedly prompt the user to enter a basic math question such as  $3.5+5$ . This math question will be sent to the server. The server calculates the math question and sends the answer back to the client. The client program will display the answer on the screen. The interaction between the client and the server will look like this:

## Client Side:

Connected with server on [IP address]

$20 + 10.5$

Answer from server: 30.5

$13 / 4$

Answer from server: 3.25

$0 / 0$

User input ends; end the client program

## Server Side:

Connected by client on [IP address]

Received question " $20 + 10.5$ "; send back answer 30.5

Received question " $13 / 4$ "; send back answer 3.25

Server program ends (due to the client closes the connection)

If the math question has the format of " $X \text{ op } Y$ ",  $X$  and  $Y$  are positive real numbers. The operations ("op") that you need to support include "+", "-", "\*", "/". The user ends the input by entering "0/0" and then the client program ends the connection to server. If the user types in an invalid math question, the client program should show an error message "Invalid question! Please enter the math question again."

In your program, you would pick a server port, which should be bigger than 1024 (but smaller than 65535). I suggest you pick a port in the range of 5000~65535 to be safe.

## Submission:

1. You will submit your source codes
2. You will submit a report to describe your program codes, explain how to compile/run your codes and include screenshots of the running client and server screen. Your screens should at least show each of four types of math operations at least once.
3. Since you need to submit several files, please zip all your files into one zip file and submit through webcourses.

## Notes:

1. Make sure you close every socket that you use in your program before your program ends. If you abort your program, the socket may still hang around and the next time you try and bind a new socket to the same port number you previously used (but never closed), you may get an error "Address already in use". If this happens, you have to wait a minute or so to let the hanging socket time out, or use a different port number.
2. Be aware that port number, when bound to sockets, are system-wide values and thus other students may be using the port number you are trying to use. If you cannot create socket with a certain port number, try a different port number.