SOFE 4790U Distributed Systems (Fall 2019)

Lab#1 – Client / Server Communications

Objective

In this lab you will modify and extend the sample programs of client/server communication attached with this lab (TCPServer.java, TCPClient.java). You will demonstrate your running programs to the TA (50% of the mark), and you will write and submit a lab report (50% of the mark).

Note: The Lab tasks must be completed in the lab. If you don't show up for a lab, you will receive a zero and there is no need for you to submit a lab report.

Tasks

Task #1: TCP Server (10 Mark)

Use the provided TCPServer.java and TCPClient.java program. Modify them so that the client prompts the user to enter a number through standard input (keyboard) and sends it to the server. The server reads message from the client, parses it, finds the summation from 0 to that number and sends the average of summation to the client. The client displays the result.

Presentation to the TA:
Demonstrate to one of the TAs the running program.

Task #2: Caesar Cipher (10 Mark)

The Caesar cipher is one of the earliest known ciphers. It is a type of substitution cipher in which each letter in the plaintext is 'shifted' a certain number of places down the alphabet. For example if the text is abcd and shift value is 3 then the cipher text will be defg. Use the provided code and modify it so that the client can send a text and shift value to the server (the information is entered through the keyboard). The server should find the cipher text and send it to client. Client displays the result.

Presentation to the TA:
Demonstrate to one of the TAs the running program.

Task#3: Repeated Inputs from Client (10 Marks)

Similar to the Echo client/server we discussed in class, the client should allow the user to keep entering messages, sending them to the server and receiving responses. Modify the client/server code in Task#2 to accomplish this.

Presentation to the TA:
Demonstrate to one of the TAs the running program.

Task#4: Multi-threaded Server (20 Mark)

In this task you need to fulfill the same requirement for task#1, but also:

1. The client should send multiple numbers like task#3.
2. The server should be able to process requests from multiple clients simultaneously.
3. The server should handle a fixed number of clients. If the number of connected client reaches the maximum (you define it) and a new client tries to connect with server, the server should send a "busy" message to the client and close the connection to that client.

Presentation to the TA:

Demonstrate to one of the TAs the running program.

Lab Report: 1 to 2 pages max (50 Marks)

In your lab report, for each task:

1- Explain how you accomplished the task.
2- Describe any challenges you faced with the tasks and how you solved them.
3- What did you learn?

Submit the lab report (in Word or PDF) on Blackboard by **11:59pm on Thursday, September 19. No late submissions will be accepted no matter what is the reason.**