CST 415 Fall Term 2016
Computer Networks
Lab 4 Assignment
Due 11:00 PM Saturday, December 3rd

Overview

The purpose of this Lab 4 assignment is to create a TCP server that will receive request messages from **another** student's Lab 3 TCP client, create a response and return the response back to the client. The Lab 3 client is to run in "scenario 2" mode (asynchronous transaction processing mode) with the response delay field set to zero milliseconds.

The client student will need to create a method to control the pacing of request messages from the client. This method of pacing control may be at the student's choosing, such as a menu selectable setting, invocation argument, or perhaps data dependent. The Lab 3 client will need to submit request messages to the Lab 4 server at no slower than 5000 requests per minute (1 request every 12 milliseconds). For purposes of testing and debugging, the student will want to be able to ratchet this pacing from a low of a single transaction, up to at least 5000 per minute.

Request and response message formats are as detailed in the description for Lab 3. The server student may populate the request and response data fields with data of their choosing. The client student needs to include a sequence number, or some other form of transaction identification uniqueness, in the transaction-id (XID) field of the request message. The server student is to echo the XID field back in the response so the client student may easily confirm there is a mutually exclusive response associated with each request.

Scenario one:

To complete Lab 4 – Scenario 1, the server student is to submit **one client student's log file** organized as detailed in the Lab 3 assignment, with a log trailer record formatted as detailed later in this document. The log file is to include 10,000 completed transactions in two minutes or less.

Scenario two:

To complete Lab 4 – Scenario 2, the server student is to concatenate and submit **two client students' log files** disclosing the transmission of 10,000 request messages simultaneously from each client. with a log trailer record formatted as detailed later in this assignment. The client students are to turn over their client logs to the server student who will concatenate them for submission to the instructor.

Scoring

A server student's successful completion of scenario one at a pace of no less than 5000 per minute will earn 45 points.

A server student's successful completion of scenario two at a pace of no less than 5000 per minute simultaneously from two client students will earn 45 points.

A successful completion must match all response messages to its respective request messages.

It is critical that server students use a client other than their own client when making their submission to fulfill this lab assignment. The programming habits of others are always different than one's own and will test and stress a software system in ways not imagined by the original programmer. In the event a server student makes a submission where they choose to have used their own client, a 25% reduction in points will be assessed.

Each client student will earn 5 points for each instance they participate as a client for another student's server.

Should a client student have more than two participations where a total of ten points has been earned, then that student will earn 5 extra credit points per participation up to a maximum of 20 extra credit points.

Points for client student participations will be entered by the instructor when scoring the server student's submissions. Client students do not need to make a submission to earn these points.

Lab 4 Log Trailer Record

The log trailer record is to include the following lines of information, separated with a <CR><LF> character sequence:

```
Requests transmitted   = [xxxxx]
Responses received     = [xxxxx]
Req. run duration (ms) = [xxxxxxxxx]
Rsp. Run duration (ms) = [xxxxxxxxx]
Trans. Duration   (ms) = [xxxxxxxxx]
Actual req. pace  (ms) = [xxxx]
Actual rsp. Pace  (ms) = [xxxx]
Configured pace   (ms) = [xxxx]
Transaction avg.  (ms) = [xxxx]
Your name:
Name of student whose client was used:
```

Req. run duration: The total number of milliseconds from when the first request was transmitted until when the last request was transmitted

Rsp. Run duration: The total number of milliseconds from when the first response was received until the last response was received.

Trans. Duration: The total number of milliseconds from when the first request was transmitted until the last response was received.

Actual req. pace: Req. run duration divided by total number of requests messages transmitted.

Actual rsp. pace: Rsp run duration divided by total number of responses messages received.

Configured pace: Configured pace of request messages in terms of milliseconds per request.

Transaction avg.: Trans. Duration divided by total number of matched responses to a request.