**CSE 489/589 Term Project Phase 1 grading details.**

1. Submitting the program and a small description (text file only) on how to run your program and things you “**HAVE/HAVE NOT**” implemented (preferably write it down in points) – 5 Marks

* A Small Text file on what your program does
* The commands implemented
* Special instructions to compile (if any)
* Your and your partners name.

1. Program Compiles without any errors and successfully displays the prompt – 5 Marks

* java Echoer <tcp-port> <udp-port>
* java Echoer abd uab
* java echoer 1234566666666666 44321 33445 ….
* So, basically you’ve got to compare the inputs for numbers and of limited size. (Up to 60000)

1. Info Command – 10 Marks

* Info -> No Loop back address
* To get the IP, do a connect on a UDP socket to a public DNS like 8.8.8.8 (Store this IP for future communication)

(Should display a valid IP address and not a Loopback address with correct Hostname and TCP port and UDP port)

1. Connect Command (Connect should have a minimum of 7 outgoing TCP connections) –10

* For eg. Connect 192.168.24.1 9999 and Connect timberlake.cse.buffalo.edu 8888 are both

Valid inputs.

* Both side should display that the connection is established.
* Connect P1 to P2

Connect P1 to P3

Connect P1 to P4

.

.

.

.

.

Connect P1 to P8

All Connections should be successful and should see 7 successfully connected messages on the P1 and 1 each successful connection message on P2 …. P8

1. Connect should block duplicate connections and self-connection – 5 Marks

* Connect P1 to P1
* Connect P1 to P2 (Sucessful Connection)

Connect P1 to P2 (Duplicate Connection)

(Suitable error message should display for attempting duplicate connections and self-connections)

1. Show Command –15 Marks

(Should display all the established TCP connections with valid data. Disconnected connections should not be displayed )

1. Send Command –20 Marks

(Message should be displayed correctly at the other end. Take care of the big/small endian which I went over during last recitation)

* Keep a floating buffer size for the message input and do not statically set the buffer size to some small fixed value.
* We won’t test it for huge data, not a file for sure.

1. Sendto Command – 20 Marks

(Message should be displayed correctly at the other end along with the sender’s connection id/ip address along with the protocol. Look at the project description for example.)

* Sendto <invalid ip> <invalid port> -> Handle errors

1. Disconnect Command –10 Marks

(When the connection is disconnected, the other side should display the message saying that the connection is lost/disconnected)

* Disconnect should work correctly
* Show after disconnect.

Total Marks – 100

cse\_submit589 echoer.tar

cse\_submit489 echoer.tar