

SPEEDTEST ASSIGNMENT (OPTIONAL)

In this fun assignment, you will race your project 3 programs with each other. We have setup a test box on **snoopy.csuglab.cornell.edu** on port **7777**. This is basically the same code as test box for project 3 except for the fact that it will not send you any UPDATE/DELETE etc requests, and it will send you the network topology through the syntax used by project 3 IP box. There is going to be a lot more nodes and subnets than project 3, so you might need to modify your program to handle that. To make things easier, the server will wait for you to send “OK” message before it starts the test. When you are done configuring the forwarding tables, you should send an “OK” message. Then, it will start sending packets to be routed as FAST as you can route them. For a fixed number of packets, it will measure the time your program took. It will then average this time over 10 runs and return you the result.

You have to change your code a little bit for this assignment. Everything is the same as described in the project 3 handout except for the following points:

1. You have to send a test mode number to the server instead of the test-type string. This number can be 0, 1, 2 or 3. These are there for the sake of the test machine since it has to generate many random packets for each test you do. 0 represents 100 packets, as a result of this test you will receive a message saying: “100 packets in 0.165802 s”. This means you were able to route 100 packets in 0.16 seconds. Same goes for 1, 2 and 3 where number of packets will increase to 500, 1000 and finally 5000. When you are initially testing and improving your code, use a small number of packets since you will be running tests quite often. When you are done however, run it with 5000 and see how good you are!
2. You have to make your box understand the final string that is sent. The format is given above. “X packets in Y s” where X is integer and Y is a double. You have to make sure you handle the single “OK” message protocol.
3. For debugging purposes, if there is a packet whose reply the server did not receive the test will end immediately and you will receive the following message: “Did not receive: %d” where %d will be the packet number. The packets are numbered from 1 to NUM_PKTS. The NUM_PKTS is defined by the parameter you send which is explained in (1). It can be 100, 500, 1000 and 5000. So you can easily keep track of packets.
4. The results will be posted on the web periodically so that you can see how other students are doing.
5. We have tested a pure echoer which does not do any routing and the fastest times for messages to go back are the following:

100 pkts	→ 0.014 s	500 pkts	→ 0.071 s
1000 pkts	→ 0.142 s	5000 pkts	→ 0.709 s

This test was done when test box ran on csug01 and IP box on csug02. Of course localhost tests will be different. If you need to run the IP box on the same machine you are developing here are the times for echoer:

100 pkts	→ 0.0035 s	500 pkts	→ 0.017 s
1000 pkts	→ 0.036 s	5000 pkts	→ 0.194 s