Due Thurs, Dec. 3, 2015 at the beginning of class.
Submit code via Blackboard and the graphs via printouts brought to class.
(email graphs to the grader if late)

Implement a UDP client and UDP server that support simple timeout-based retransmission with AIMD congestion control:

- The client should send as many packets as are in its CURWIN (current window) and wait for each one to be ACKd. If any are not ACKed within the indicated timeout, those packets should be resent. The client should update its AIMD window each round trip as indicated below. Remember that the window should never move forward further than MINAUNACK+CURWIN (where MINUNACK is the smallest index of a packet that has not yet been ACK'd).

- The server receives and ACKs packets that arrive in-order only. Any packets except the next one expected should be silently dropped.

- Use two UDP packet formats:

  - Data: a 1 byte packet number followed by data (use dummy data)

  - ACK: a 1 byte packet number

- Start the server first. The client should report the number of packets and bytes that need to be retransmitted, as well as the time from the first packet sent until the last ACK is received.

- Use the following parameters:

  - Timeout = 50 ms (milliseconds)

  - Set the receive and send socket buffers to 64KB (using setsockopt)

  - Set MAXWIN as indicated in the experiment below

- Start with a window size of 1. Each round trip time:

  - If no packets are lost, increase the window by 1 up to MAXWIN

  - If packets are lost, drop the size of MAXWIN by at least 50% (don't forget to account for odd numbers).

- Use 10ms as the round trip time (don't worry that this isn't the actual round trip time).

- Run the experiment between two _different_ machines ON THE SAME LAN (e.g., on the same WIFI network or on Ethernet in the same room).

  - Measure 5 runs with the same configuration, where each run successfully transfers as much data as possible in 1 minute (of real time); plot points as average +/- std. dev.

  - Vary MAXWIN from 1 to 256 packets in steps of 2x.

  - Measure the time needed, number of bytes lost, and number of UDP packets lost.

Submit the code and the following plots:

- Throughput, lost bytes, and lost packets for UDP packet sizes of 32B, 512B, 1400B, and 8192B

In total, you should turn in 4 plots (you can put up to two on each page). Each plot is worth 1 point, and the code is worth one point.

For each plot, the X axis is the window size (logarithmic), and Y is the measured performance (throughput, lost bytes, lost packets). Plot throughput, lost bytes, and lost packets on the same graph.