

### Part A

When the program runs and finishes, a file called *analysis.txt* is created and has information to the first 2 transactions after the 3-way handshake along with source and destination IP addresses and ports, SEQ, ACK, window size, and total sender throughput. The total sender throughput was calculated by subtracting the time between the receiver returning a FIN tag and the first transaction after the handshake and dividing the result (result in the denominator) with the total bytes sent by the sender. The total bytes were found by adding lengths of the TCP (ip.data in the code) for every sent data segment.

### Part B

In theory, the congestion window size would change based on retransmissions by triple duplicate ACKs and retransmission by timeout. The congestion window will start off slowly (aka the slow phase). Then, the cwnd will increase slowly until it reaches a steady-state. Then, from the steady-state, the cwnd will improve slowly (congestion avoidance phase). This is implemented in TCP Tahoe.

The idea is that initially, the window size will start at 1 and then cwnd will double for every RTT completed. If there is retransmission due to duplicates, the threshold value becomes half of the current window size, the current window size becomes halved and starts from here. Similarly, if there is retransmission due to timeout, then the threshold becomes half of cwnd, cwnd goes down to 1 and restarts from the slow phase. To avoid further congestion, once cwnd reaches the threshold every RTT will increase cwnd by 1 and for every ACK will increase cwnd by  $1/\text{cwnd}$ . The number of retransmissions can be found when there is a fast-retransmission present. Anything that is not a triple-duplicate ACK retransmission is treated as retransmission by timeout.