2a.

The sequence number, and ack number are randomly generated and the information is located in the tcp section. The receive window size is determined by the window section of the packet offset by the number obtained in the SYN packet. The sequence number currently indicates that the sender is currently sending packets for the first 2 transactions in the ports.

Port 43498 Sequence number: 705669103 705669127 Ack number: 1921750144 1921750144 Receive Window size: 49152 49152	Port 43500 Sequence number: 3636173852 3636173876 Ack number: 2335809728 2335809728 Receive Window size: 49152 49152
Port 43502 Sequence number: 2558634630 2558634654 Ack number: 3429921723 3429921723 Receive Window size: 49152 49152	

(b)The throughput was determined by the total number of bytes added together and divided by the difference between the start time and end time of the flow.
(c) The loss rate for each flow was calculated by the number of duplicated packets that were repeatedly recieved divided by total number of packets

(1)
The congestion window was approximated by the number of windows able to be sent before all the acks returned. All of the packets sent were counted until the first ack returned. When the ack returned this started counting for the next window. When the final ack for the first send window arrives, the second congestion window closes and starts counting for the third. The windows are then closed when the last ack from the previous window arrives.
The congestion window seems to double for the first five windows. The doubling is off by some numbers, but it seems close. The initial window seems to be 10.
Congestion Window for Port 43498
[10, 30, 74, 173, 360]
Congestion Window for Port 43500
[10, 30, 69, 136, 346]
Congestion Window for Port 43502
[10, 30, 53, 125, 298]

(2) Compute the number of times a retransmission occurred due to triple duplicate ack and the number of time a retransmission occurred due to timeout.

Triple ack was determined by the number of packets sent that had the same sequence as the ack before that and was repeated 3 times or more. Timeout was more difficult and less accurate due to the RTO calculated(RTO=(1-alpha)old_rtt+alpha(new rtt)). The RTO got huge with the number of delays. Timeout was implemented on the number of packets sent that had the same sequence as the ack before to see if it was a timeout retransmission.