Computer Communications and Networks (COMN)
2020/21, Semester 2

Assignment 2 Results Sheet

Forename and Surname:	Orges Skura
Matriculation Number:	S1813106

Question 1 – Number of retransmissions and throughput with different retransmission timeout values with stop-and-wait protocol. For each value of retransmission timeout, run the experiments for **5 times** and write down **average number of retransmissions** and **average throughput**.

Retransmission timeout (ms)	Average number of re-transmissions	Average throughput (Kilobytes per second)
5	2755.8	69.2
10	1633.5	66.3
15	132.4	72.2
20	96	71.2
25	109.8	67
30	107.8	64.5
40	111.6	58.6
50	97.4	57.1
75	113.8	46.9
100	102.7	41.4

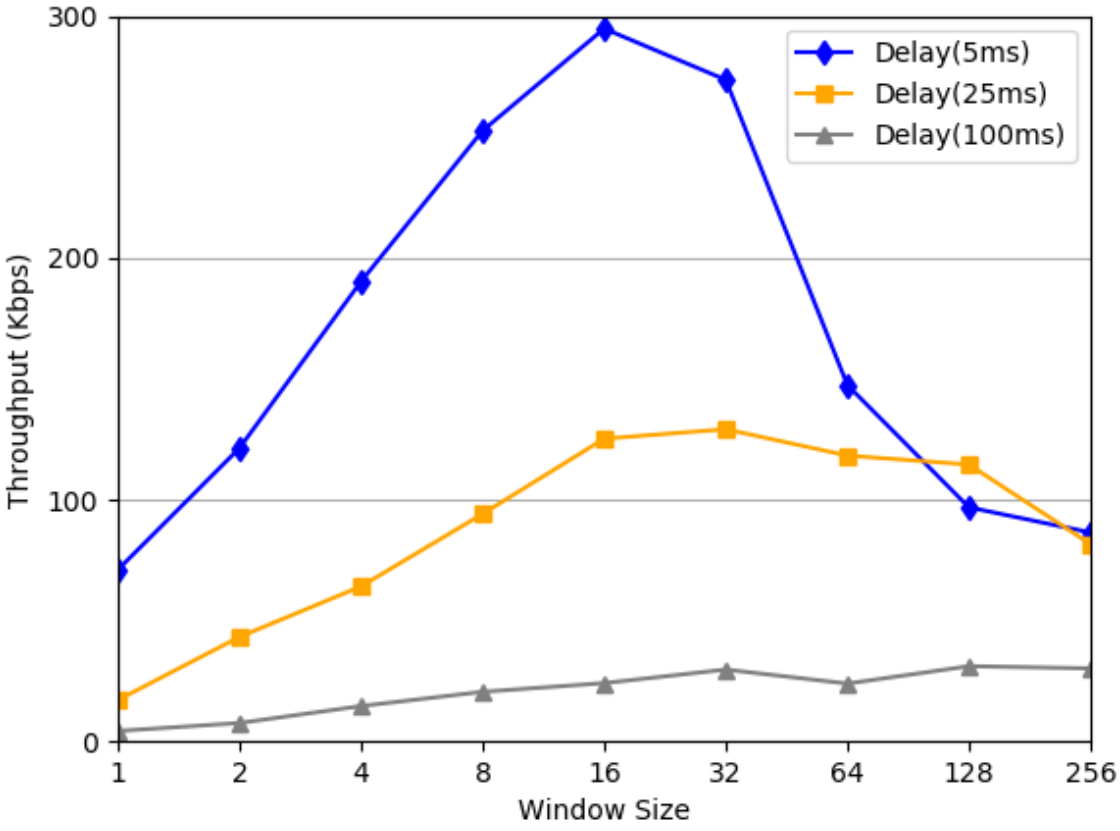
Question 2 – Discuss the impact of retransmission timeout value on the number of retransmissions and throughput. Indicate the optimal timeout value from a communication efficiency viewpoint (i.e., the timeout that minimizes the number of retransmissions while ensuring a high throughput).

Increasing the retransmission timeout value makes the number of transmissions decrease dramatically up until 15ms where after that the re-transmissions remain mostly the same. So increasing retransmission timeout would decrease the number of retransmissions for small retransmission timeout. When it gets bigger than 20, increasing the retransmission timeout does not affect the number of retransmissions. This happens as we have set the delay to 5 ms and having a bigger retransmission timeout would count for these delays and help decrease re-transmissions.

The average throughput remains on similar values till retransmissions timeout of 25 and after that we see a decline in throughput with the increase in retransmission timeout. This happens because we set a large timeout and if a packet is lost we wait a lot before re transmitting it.
The optimal timeout value for my chart would be 20ms as it keeps a very high throughput while having a small number of retransmissions.

Question 3 – Experimentation with Go-Back-N. For each value of window size, run the experiments for **5 times** and write down **average throughput**.

1	71	17.1	4.3
2	121.3	43.2	7.6
4	190.2	64.1	14.6
8	252.6	94.1	20.5
16	294.8	125.2	24.1
32	273.6	129.1	29.7
64	147.4	118.2	23.9
128	96.8	114.5	31.1
256	86.4	81.6	30.2



Question 4 – Discuss your results from Question 3.

I used 20 ms RetryTimeout for 5ms, 60ms for 25 ms propagation delay and 210 ms for 100ns propagation delay. The reason I chose 60 ms and 210 ms is because I want to have a timeout that is larger than the delay of sending a packet and receiving the ACK. For 25ms propagation delay it needs to be bigger than 50 and for 100 ms delay it needs to be bigger than 200.

I observed that by increasing window size the throughput gets bigger as there are more packets on each grouping to be sent. A too large window size, however, can actually cause the throughput to drop as the receiver becomes overwhelmed with incoming packets, being unable to ack packets quickly enough for the sender to move the window forwards at a reasonable rate. Whether or not a window size is 'too large' depends on the propagation delay in the channel. The larger the propagation delay is, the higher the window size at which the throughput peaks, as the receiver is less overwhelmed by a large number of packets due to their slower incoming speed.

This can be seen in the graph where for 5ms, the peak is on window size 16. For delay 25ms the peak is at 32 while for 100ms delay, peak is at 128.

Question 5 – Experimentation with Selective Repeat. For each value of window size, run the experiments for **5 times** and write down **average throughput**.

1	18.6
2	40.1
4	69.5
8	119.2
16	170.1
32	42.3

Question 6 - Compare the throughput obtained when using "Selective Repeat" with the corresponding results you got from the "Go Back N" experiment and explain the reasons behind any differences.

The results I obtained from using Selective Repeat were slightly faster than Go Back N. This makes sense, as the main problem with Go Back N that Selective Repeat fixes is the re-sending of out-of-order packets that have already been sent and received successfully by the Receiver. That is, if packets 1, 2, 3 and 4 are sent to the Receiver in Selective Repeat and only 3 and 4 are received, then the Sender will only have to re-send packets 1 and 2 (assuming the Receiver acked packets 3 and 4). In Go Back N theSender would have had to resend all four packets.

But for window size 32, the results on Go Back N are a lot better than Selective Repeat. This could be attributed to different methods I used for those protocols. For Go Back N I used non-blocking sockets while for Selective Repeat I used multi-threading. As I have a thread for timeouts, having to manage a larger window size might substantially decrease the performance of the algorithm and overall throughput.

Question 7 – Experimentation with *iperf*. For each value of window size, run the experiments for **5 times** and write down **average throughput**.

1	
2	
4	
8	
16	
32	

Question 8 - Compare the throughput obtained when using "Selective Repeat" and "Go Back N" with the corresponding results you got from the *iperf* experiment and explain the reasons behind any differences.