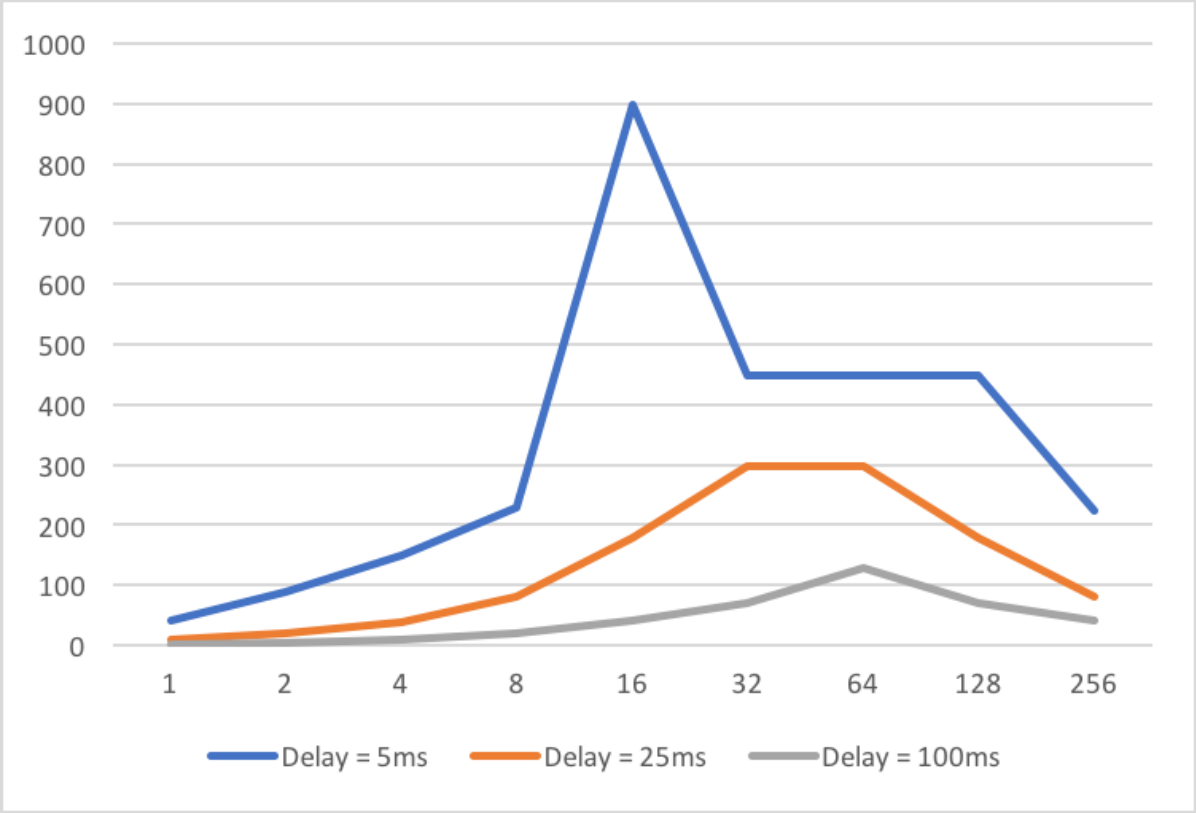
Computer Communications and Networks (COMN)
2016/17, Semester 2

Assignment Part 2 Results Sheet

Forename and Surname:	Deirdre Bringas
Matriculation Number:	S1368635

Question 1 – Experimentation with Go-Back-N:

	Throughput (Kilobytes per second)		
Window Size	**Delay = 5ms**	**Delay = 25ms**	**Delay = 100ms**
1	40	9	2
2	89	19	5
4	149	39	10
8	299	81	20
16	899	179	40
32	449	299	69
64	449	299	128
128	449	179	69
256	224	81	40



Graph of Throughput in Question 1.

Question 2 – Discuss your results from Question 1.

As seen from both the table and the graph, the optimal window size is N=16 and a delay of 5ms. Throughout the differently lengths of delays, N=16 proves to be the most consistent in both throughput and minimal packet loss. However, if the delays are 25ms or 100ms, N=32 are more optimal. Although the throughput and window size have a positive correlation, the received image isn't complete, though I am unsure if this is due to my implementation. Since the receiver for GBN has only a window of 1, it cannot accept out of order packets and so the sender must keep sending them. Therefore, having a larger window size for the sender creates congestion in the pipeline and the last few packets (and the packet containing the flag to end the file) are "lost". If the error rate is high, then GBN doesn't work well. I believe this is why my Receiver2a doesn't abort after Sender2a indicates that it has sent the entire file for N=64,128,256. Of course, the delay also affects the throughput, but it also affects how many packets are lost, as I had mentioned earlier. Since there is more of a delay in higher window sizes, there is less congestion in the pipeline, though even having a delay of 100ms doesn't seem to fix the loss of the final packets. Nonetheless, the throughput for lower window sizes are acceptable.

Question 3 – Experimentation with Selective Repeat

Window Size	Throughput (Kilobytes per second) Delay = 25ms
1	9
2	19
4	31
8	29
16	31
32	37

Question 4 - 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.

From the two tables, we can see that GBN has a higher throughput rate than Selective Repeat (SR) across all window sizes with delay = 25ms. This may be because the receiver for the SR protocol must individually acknowledge correctly received packets. The lack of the cumulative acknowledgement in SR lowers the throughput. Despite having a lower throughput, using SR is more efficient in higher window sizes. For instance, for N=64 (not shown in the SR experiment table), the throughput is roughly 69KB/s but the entire file is transferred unlike when we use GBN. Since the receiver for SR buffers out of order packets, there is less congestion in the pipes and packets are transferred efficiently.

Question 5 – Experimentation with *iperf*

Window Size (KB)	Throughput (Kilobytes per second) Delay = 25ms
1	14
2	17.9
4	39.1
8	76.7
16	120
32	102

Question 6 - 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.

The throughput using iperf is in between the throughputs of GBN and SR with a delay of 25ms. With higher window sizes iperf doesn't transfer the whole file, sometimes only transferring about half of the size of the original file. Iperf uses TCP, which is almost like a combination of the both of them as it uses cumulative acknowledgement (like GBN) and also stores out-of-order packets (like SR). Therefore, it would make sense that the throughput is in between.