1. (10 points) Suppose A and B are connected with a 40Mbps link. Suppose the round trip propagation delay on the link is 200ms. Segment lengths are 10,000 bits. Consider the maximum number of in-flight segments possible on this link.

   (a) (5 points) How many bits are needed in the sequence number field for SR?

   RTT + L/R = $0.20025 \approx 0.2$
   Delay bandwidth product $\approx 0.2 * 40 \cdot 10^6 = 8 \cdot 10^6$
   For a 10,000 bit segment or frame, that is about 800 frames. For SR you will need about 1,600 sequence numbers, which require 11 bits.

   (b) (5 points) How many bits are needed in the sequence number field for GBN?

   For GBN you will need about 801 sequence numbers, which require 10 bits.

2. Suppose two stations A and B are connected with a 10Mbps link. Suppose the round trip propagation delay on the link is 50ms. Packet lengths are 10,000 bits. There are no channel errors.

   (a) (5 points) What is the channel utilization for stop-and-wait ARQ if the block size is 1?

   $L/R = 0.001, RTT + L/R = 0.051$
   $U = 1/51$

   (b) (5 points) What must the block size be for the channel utilization for stop-and-wait to be greater than 90%?

   $\frac{x}{50+x} = 0.9$ Therefore $x = 450$.

(c) (10 points) Consider A newly established TCP connection that sends a message of 100,000 bytes

    i. (5 points) Will there ever be any outstanding segments with the same sequence number? No

    Explain: For TCP $2^{32} << 100,000$

    ii. (5 points) Will the sequence number ever roll over? Yes, it is possible.

    Explain: The sequence number does not necessarily start at 1. The value of the syn cookie can be any of the 2^{32} sequence number values.

(d) (10 points) UDP and TCP use 1s compliment for their checksums. Suppose you have the following three 8-bit bytes: 01010101, 01100110, 01011101.

    i. (5 points) What is the 1s compliment of the sum of these 3 8-bit bytes?

    11100110

    ii. (5 points) With the 1s compliment scheme, how does the receiver detect errors?

    The receiver adds the numbers and then adds the result to the checksum. If the result is not all 1's, there is an error.