1. Consider two hosts, A and B, connected by a single link with a capacity of $R$ bps. Suppose the two hosts are separated by $m$ meters, the propagation speed along the link is $s$ meters/sec, and host A needs to send a packet of size $L$ bits to host B.
   a. Express the propagation delay (one-way delay for a bit)
   b. Express the transmission delay (time to send the packet)
   c. Express the end-to-end delay between when the packet is sent and it is completely received (assume no overhead or queuing delay)

2. Many companies transfer very large amounts of data (i.e., many terabytes) across the country by physically sending disk drives via UPS instead of transmitting it over a network. Why might this technique work well for these applications (e.g., sending a month of customer purchasing data at Walmart to a remote data center), but not be widely used in data networking?

3. Individual protocol layers generally encode their layer-specific state in protocol headers, prepended to the front of the data they are encapsulating. However, a common exception is the error detection code value (e.g., the CRC) which is commonly appended after the data. For example, an Ethernet frame is composed of a header – which provides length base sentinel, addressing and other meta-data, followed by the actual data that it is trying to send, then followed by a CRC calculated over the header and data. Why do you think these error detection fields are not directly incorporated into the protocol header?

4. If all the links in the Internet were to provide a reliable delivery service would a reliable transport layer service be completely redundant? Why or why not?

5. Consider the following byte stream: 0xff 0x00 0x14 0x15 0x00 0x67 0x88 0x92 Describe what the output stream looks like under the following conditions:
   a) Byte-level sentinel framing (where STX = 0x93, ETX = 0x92 and DLE=0x14)
   b) Consistent overhead byte stuffing (reminder, the first byte will indicate how many bytes to the first zero or 0xff if there is no zero in the first 254 bytes).