Roann Yanes (NetID: ryanes)
CSE 30264 Computer Networks
Professor Dong Wang
October 8, 2018

Homework 2

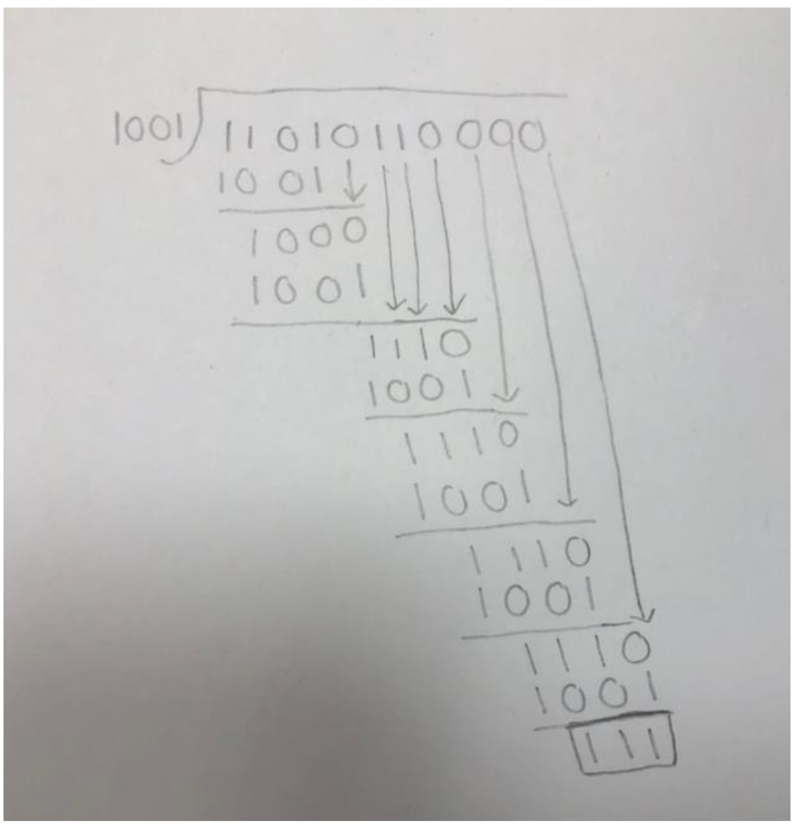
Problem 1:

(a) M(x) = 11010110

T(x) = 11010110000

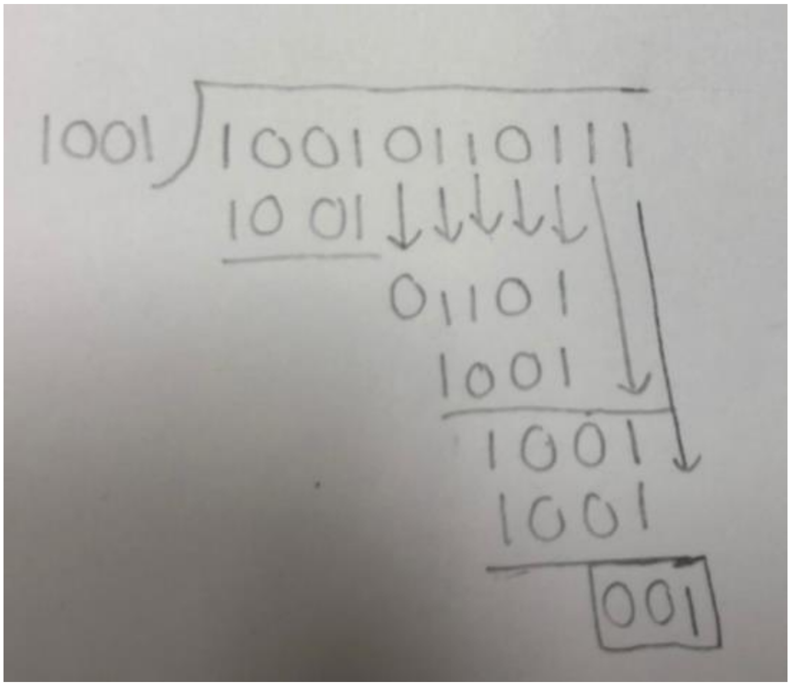
R(x) = 111

P(x) = T(x) – R(x) = 11010110111 (this is the message sent)



(b) The remainder is non-zero (R(x) = 001), so the receiver knows that the message is corrupt.

P'(x) = 10010110111



Problem 2:

(a) The one-way propagation delay for this link:

$$\text{Propagation Delay: } \frac{30*10^3\ \text{m}}{2*10^8\frac{\text{m}}{\text{s}}} = 1.5 * 10^{-4}\ \text{sec} = 150\ \mu s$$

(b) A suitable timeout value for the Stop-and-Wait algorithm to use is twice the roundtrip time (according to the book). In this case, the roundtrip time is $150\ \mu s * 2 = 300\ \mu s$. Therefore, $\boldsymbol{600\ \mu s}$ is and appropriate roundtrip time for this algorithm.

(c) Given a timeout value of $\boldsymbol{600\ \mu s}$, it is possible for the Stop-and-Wait algorithm to timeout and retransmit a frame because using only the propagation delay to calculate a timeout value does not take into consideration slow processing of the frame that might occur on the receiving end; there may be delayed acknowledgment in this case, as the receiver could take a while to generate an acknowledgment. Similarly, there may be slow processing of the acknowledgment by the sender as well, which could very well result in timeout and retransmission.

Problem 3:

The bandwidth × delay (roundtrip) product for the link is:

$187.5\ \text{KBps} * (2 * 3\ \text{seconds}) = 1{,}125\ \text{KB}$ $\quad$ $(1.5\ \text{Mbps} = 187.5\ \text{KBps})$

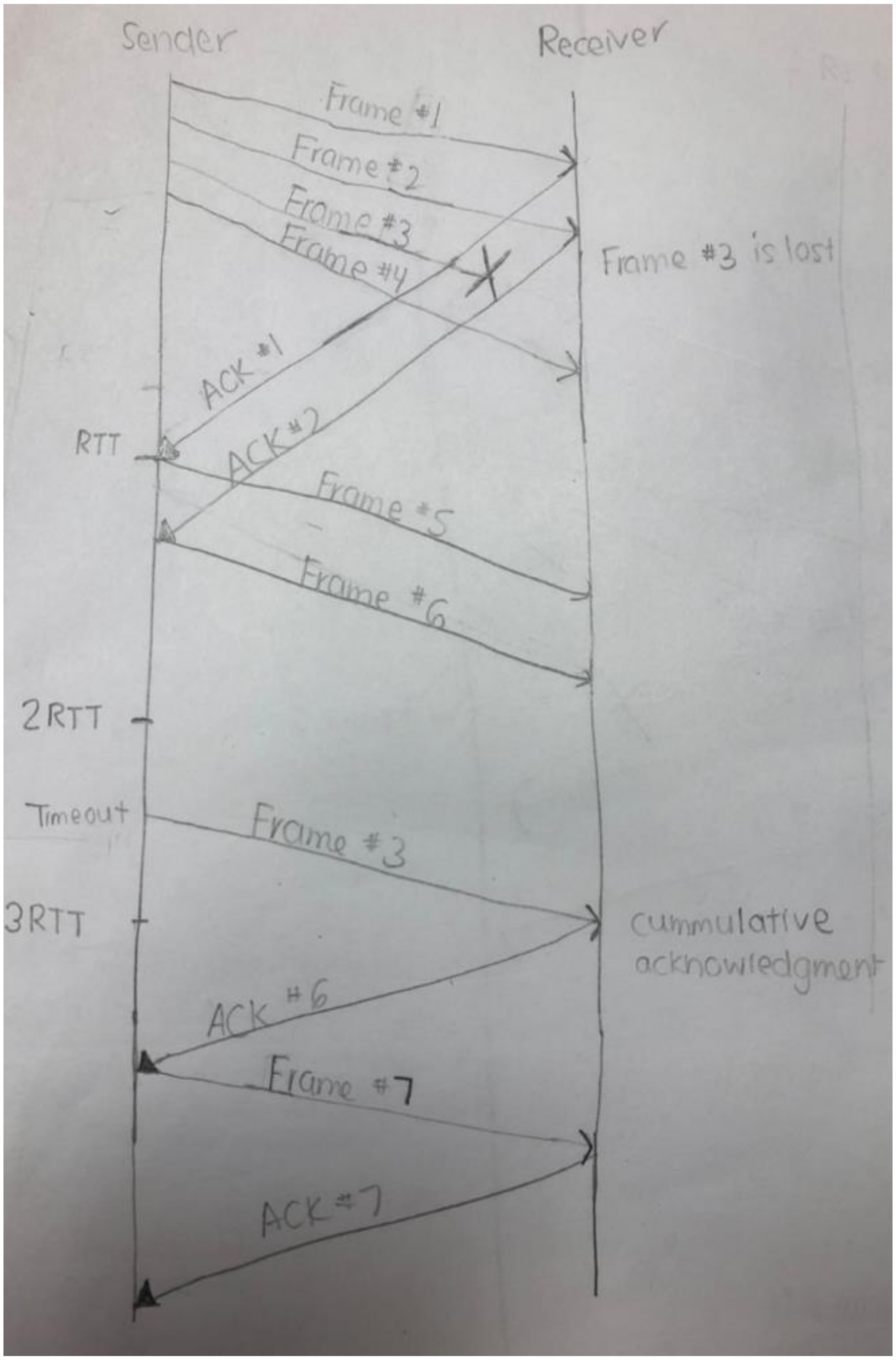
$$1{,}125\ \text{KBps} * (2 * 3\ \text{seconds}) = 1{,}125\ \text{KB}$$

Therefore, we have 1,125 packets and the window size should be 1,125 KB. The space for the sequence number must be twice as large as the window size, so 2,250 KB.

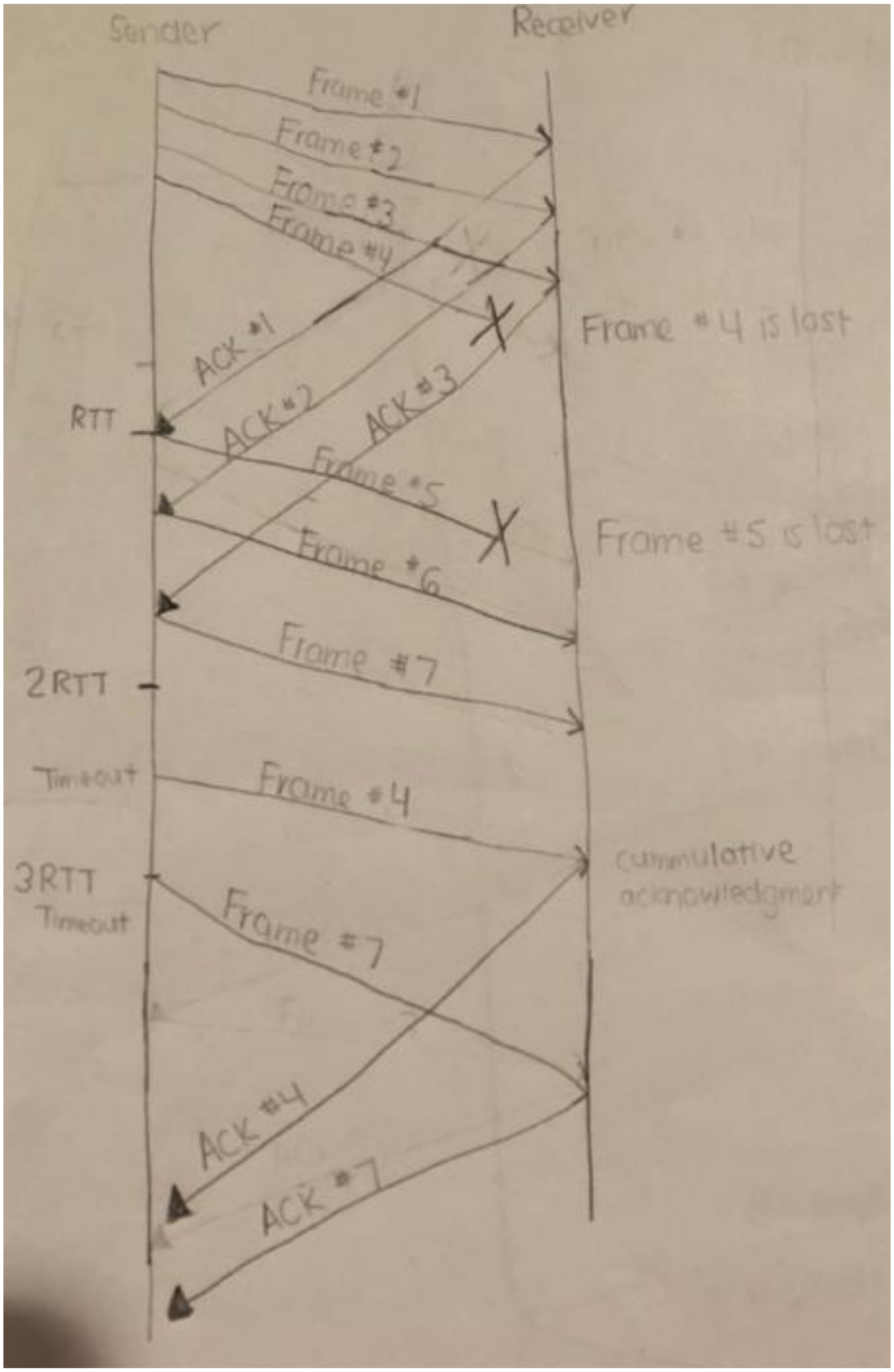
Therefore, $\log_2(2{,}250) = \mathbf{12\ bits}$.

Problem 4:

(a)



(b)



Problem 5:

The MAC protocol of Ethernet is CSMA/CD (Carrier Sense Multiple Access and Collision Detect, respectively). CSMA/CD ensures that all nodes are able to differentiate between channels that are in use and channels that are idle ("carrier sense") and that the senders listen to the channel while transmitting to detect when the frame that is in transmission has collided with another frame that is already in transmission by another host ("collision detection"); nodes on the Ethernet receive every other node's transmission, and nodes are able to transmit and listen at the same time. The Ethernet makes use of the p-persistent algorithm (transmitting with probability $0 \leq p \leq 1$ after a line becomes idle and transmitting immediately after noticing an Ethernet adaptor becomes idle), meaning that whenever a node has a frame to send and it senses an idle slot, it transmits with probability p and waits until the next slot with probability $1 - p$. When a node detects its frame colliding with another frame, it transmits a jamming sequence and stops the transmission. Wireless MAC protocol does not use "collision detection," rather, it uses "collision avoidance." Wireless nodes may be too far away or blocked by an obstacle to be able to receive every other node's transmission, and wireless nodes cannot transmit and receive at the same time because the power generated by the transmitter is much higher than any received signal is likely to be and has the effect of swamping the receiving circuitry; therefore, "collision detection" cannot be supported by wireless nodes.

(Source: Computer Networks: A Systems Approach)

Problem 6:

The hidden terminal problem is when two nodes that are within range of another node (B) but not within range of each other try to transmit to node B at the same time, causing their frames to collide (as they were not aware of each other) and the collision to go undetected. The hidden terminal problem is addressed in 802.11 networks by the RTS-CTS (Ready to Send-Clear to Send), where the sender transmits an RTS packet to the receiver and waits for the CTS packet from the receiver upon successfully sending the RTS packet; this signals to the nodes within range of the receiver that they should not transmit for the amount of time specified in the RTS and CTS packets. A collision is detected when the CTS packet has not been received, and the sender will wait a random period of time before transmitting again. An ACK packet is received upon a successful RTS-CTS exchange.

(Source: Computer Networks: A Systems Approach)