Stylianos Adamou -> 1038399

P40

a) 1:6 , 23:26 -> TCP slow start
b) 6:16, 17:23 -> TCP congestion avoidance
c) Triple duplicate ACK because the congestion window size have not been set to 1 but is equal to the value of threshold
d) Timeout because congestion window size was set to 1
e) 32 because is the window size that slow start stops and congestion avoidance begins
f) ssthresh = 21 because congestion window at the loss event at transmission round 18 is 42 and ssthresh value will be equal with the congestion window/2
g) ssthresh = 14 because congestion window at the loss event at transmission round 22 is 29 and ssthresh value will be equal with the congestion window/2 = 29/2 = 14.5 -> 14
h) 1st Transmission round -> segment 1, CongWin = 1 -> (slowstart)
   2nd Transmission round -> segment 2-3, CongWin = 2 -> (slowstart)
   3rd Transmission round -> segment 4-7, CongWin = 4 -> (slowstart)
   4th Transmission round -> segment 8-15, CongWin = 8 -> (slowstart)
   5th Transmission round -> segment 16-31, CongWin = 16 -> (slowstart)
   6th Transmission round -> segment 32-63, CongWin = 32 -> (slowstart)
   7th Transmission round -> segment 64-96, CongWing = 33 -> (avoidance)
   Segment 70th will be sent at 7th transmission round
i) Packet loss at 26th transmission round - triple duplicate ACK
   CongWing at loss event = 8
   Threshold = CongWin/2 = 8/2 = 4
   CongWin = Threshold = 4
   But Threshold will be added 3 MSS for each the triple duplicate ACK
   Threshold Value = 7
j) ssthresh = 21, CongWin = 1 because tahoe cannot understand from triple ACK it handles all losses event with Timeout

k) when timeout event occurs at 16th round threshold will be set to CongWin/2= 42/2 = 21
   17th Transmission round -> 1 segment, CongWin = 1 -> (slowstart)
   18th Transmission round -> 2 segment, CongWin = 2 -> (slowstart)
   19th Transmission round -> 4 segment, CongWin = 4 -> (slowstart)
   20th Transmission round -> 8 segment, CongWin = 8 -> (slowstart)
   21th Transmission round -> 16 segment, CongWin = 16 -> (slowstart)
   22th Transmission round -> 21 segment , CongWin = threshold = 21
   Total packets = 1 + 2 + 4 + 8 + 16 + 21 = 52 packets

Stylianos Adamou -> 1038399

P37

- ⓘ Large Timeout value in order that all ACKs can be received
- ⓘ Host A send 5 segments to B
- ⓘ 2nd segment is lost sent by A
- ⓘ All segments at the end will correctly received by host B

a) GBN
Segment 1 -5 will all be sent to B. B will respond with ACK1-Seq1 in all 4 segments as it only received the 1st segment and 2nd segment never arrives. When timeout will be occurred after A send all 5 segments then A will retransmit packet 2,3,4,5 and B will ACK all of them with sequence number 2,3,4,5.
B send 8 ACKs
A send 9 segments

Selective Repeat
Segment 1 -5 will all be sent to B. B will respond with ACK1-Seq 1,ACK3-Seq3,ACK4-seq4,ACK5-seq5 as segment 2 was lost. When timeout will be occurred after A send all 5 segments then A will retransmit packet 2 and B will ACK packet 2 with ACK2- Seq2
B send 5 ACKs
A send 6 segments

TCP
B send 5 ACKs -> 4 ACK1 with sequence number 2, ACK5 with sequence number 6
A send 6 segments -> Segments 1,2,3,4,5 and resent segment 2 after triple duplicate ACK

b) TCP is the faster because uses fast retransmit and doesn't wait until Timeout Occur

P36

Since TCP does not know whether a duplicate ACK is caused by a lost segment or just a reordering of segments(send ACK2 first then ACK0 and then ACK1), it waits for a small number of duplicate ACKs to be received.

It is assumed that if there is just a reordering of the segments, there will be only one or two duplicate ACKs before the reordered segment is processed, which will then generate a new ACK. If three or more duplicate ACKs are received in a row, it is a strong indication that a segment has been lost.

P46)

a) LinkRate = CongWin/RTT = Segments * MSS / RTT
(Maximum window size in segments) Segments = RTT * LinkRate / MSS = 10 * 10^6 * 0.15 / 1500 * 8 = 125

b) When a loss detected congestion window size will be set the value of threshold which is W/2 as slowstart phase is ignored. So congestion window size varies from W/2 to W until reach the new maximum window size which is W

Average Window Size = 0.75 * W = 0.75 * 125 = 93.75
Average Throughput = Average window Size * MSS /RTT = 94 * 1500 * 8 / 0,15 = 7.52 Mbps

c) In order to increase the window size from W/2 -> W a W/2 * RTT is required as window size increases by one in each RTT
125/2 * 0.15 = 9.45 seconds