LEBANESE AMERICAN UNIVERSITY

Electrical and Computer Engineering Dept

COE 431
Computer Networks

Spring 2013

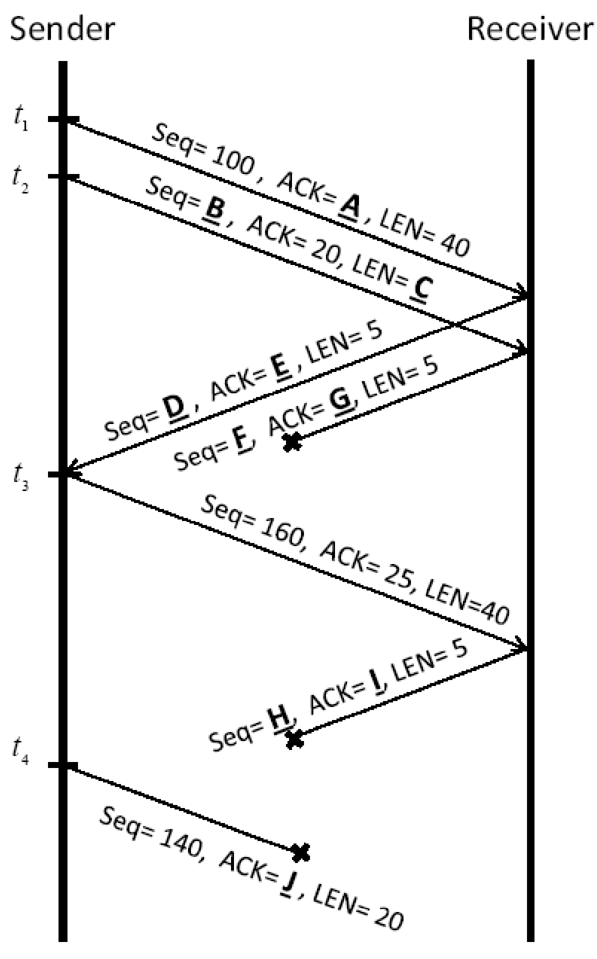
W. FAWAZ

Assignment II

Problem I

Complete the missing sequence numbers (Seq), Acknowledgment numbers (ACK) and segment length (LEN) in the following TCP connection. Assume:

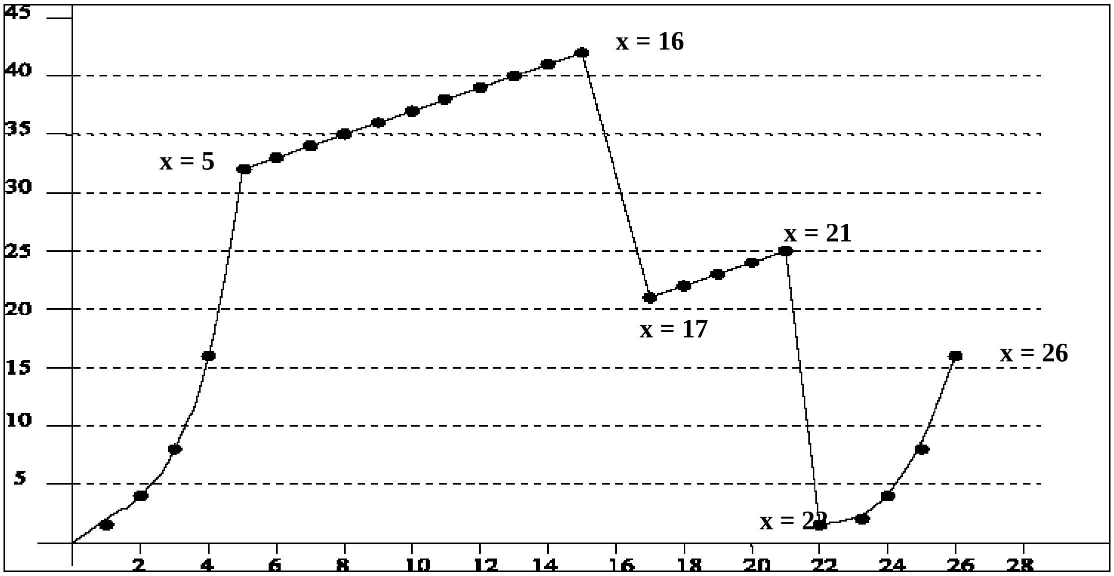
- No timeouts occur at the receiver
- The sender starts the timer at t1
- The sender and the receiver have always data to transmit
- There are no delayed acknowledgments at the sender or receiver



A	20
B	140
C	20
D	20
E	140
F	25
G	160
H	30
I	200
J	25

Problem II

Consider the following plot that depicts the evolution of TCP Reno's congestion window as a function of time.



1. Identify the intervals of time during which TCP is in a slow start phase.

0 – 5, 22 – 26

2. Identify the intervals of time during which TCP is in a congestion avoidance phase.

5 – 22

3. After the 16th transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

Triple duplicate ACK

4. After the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

Timeout

Problem III

UDP uses 1s complement for the checksum. Consider the following three 8-bit code words: 01010101, 01110000, and 01001100.

1. What is the 1s complement of the sum of these 8-bit code words? Show all work.

Note, wrap around if overflow.

$$\begin{array}{ccccccccc} & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ + & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ \hline & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{array}$$

$$\begin{array}{ccccccccc} & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ + & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ \hline & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}$$

2. With the 1s complement, how does the receiver detect errors?

One's complement = 1 1 1 0 0 0 1 1.

To detect errors, the receiver adds the four words (the three original words and the checksum). If the sum contains a zero, the receiver knows there has been an error.

3. Is it possible that a 1-bit error will go undetected? Justify your answer.

All one-bit errors will be detected.

4. Is it possible that a 2-bit error will go undetected? Justify your answer.

Two-bit errors can be undetected (e.g., if the last digit of the first word is converted to a 0 and the last digit of the second word is converted to a 1).

Problem IV

Hosts A and B are communicating over a TCP connection, and Host B has already received from A all bytes up through byte 358. Suppose Host A then sends two segments to Host B back to back. The first and second segments contain 50 and 80 bytes of data, respectively. In the first segment, the sequence number is 359, the source port number is 1028, and the destination port number is 80. Host B sends an acknowledgement whenever it receives a segment from Host A.

1. In the second segment sent from Host A to B, what are the sequence number, source port number, and destination port number?

In the second segment from Host A to B, the sequence number is 409, source port number is 1028 and destination port number is 80.

2. If the first segment arrives before the second segment, in the acknowledgement of the first arriving segment, what are the acknowledgment number, the source port number, and the destination port number?

If the first segment arrives before the second, in the acknowledgement of the first arriving segment, the acknowledgement number is 409, the source port number is 80 and the destination port number is 1028.

3. If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, what is the acknowledgement number?

If the second segment arrives before the first segment, in the acknowledgement of the first arriving segment, the acknowledgement number is 359, indicating that it is still waiting for bytes 359 and onwards.

Problem V

Consider sending a large file from a host to another over a TCP connection that has no loss.

1. Suppose TCP uses AIMD for its congestion control without slow start. Assuming **CongWin** increases by 1 MSS every time a batch of ACKs is received and assuming approximately constant round-trip times RTT, how long does it take for **CongWin** to increase from 1 MSS to 6 MSS (assuming no loss events)?

It takes 1 RTT to increase CongWin to 6 MSS; 2 RTTs to increase to 7 MSS; 3 RTTs to increase to 8 MSS; 4 RTTs to increase to 9 MSS; 5 RTTs to increase to 10 MSS; and 6 RTTs to increase to 11 MSS.

2. What is the average throughput (in terms of MSS and RTT) for this connection up through time = 5 RTT?

In the first RTT 5 MSS was sent; in the second RTT 6 MSS was sent; in the third RTT 7 MSS was sent; in the fourth RTT 8 MSS was sent; in the fifth RTT, 9 MSS was sent; and in the sixth RTT, 10 MSS was sent. Thus, up to time 6 RTT, 5+6+7+8+9+10 = 45 MSS were sent (and acknowledged). Thus, we can say that the average throughput up to time 6 RTT was (45 MSS)/(6 RTT) = 7.5 MSS/RTT.