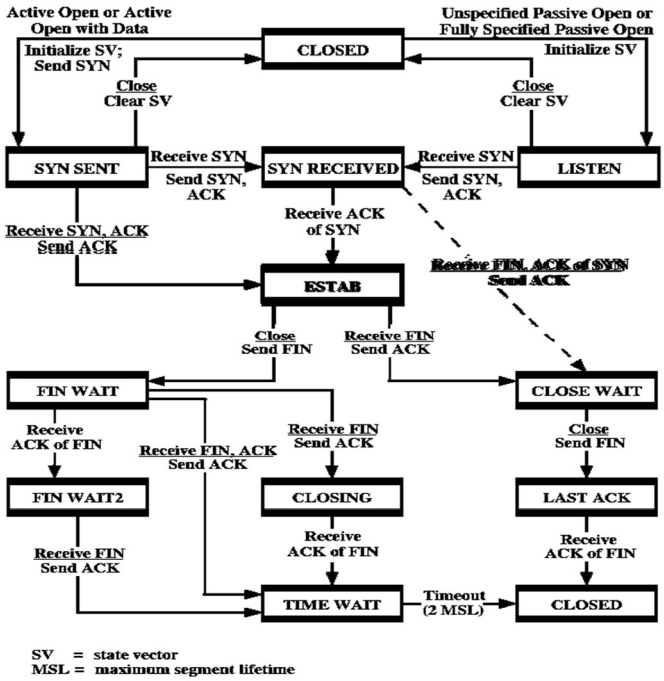
CMPT 371 Assignment 3

Junchen Li (301385486)
Wenqing Liu (301417132)

2021/7/27

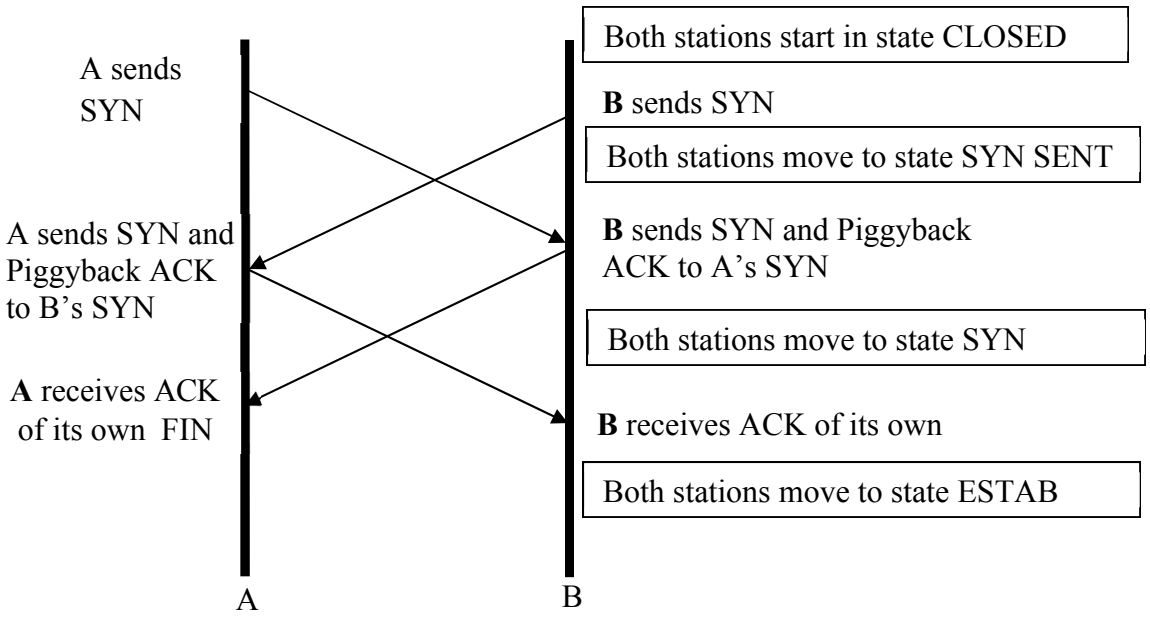
Assignment 3: CMPT 371 SOLUTION

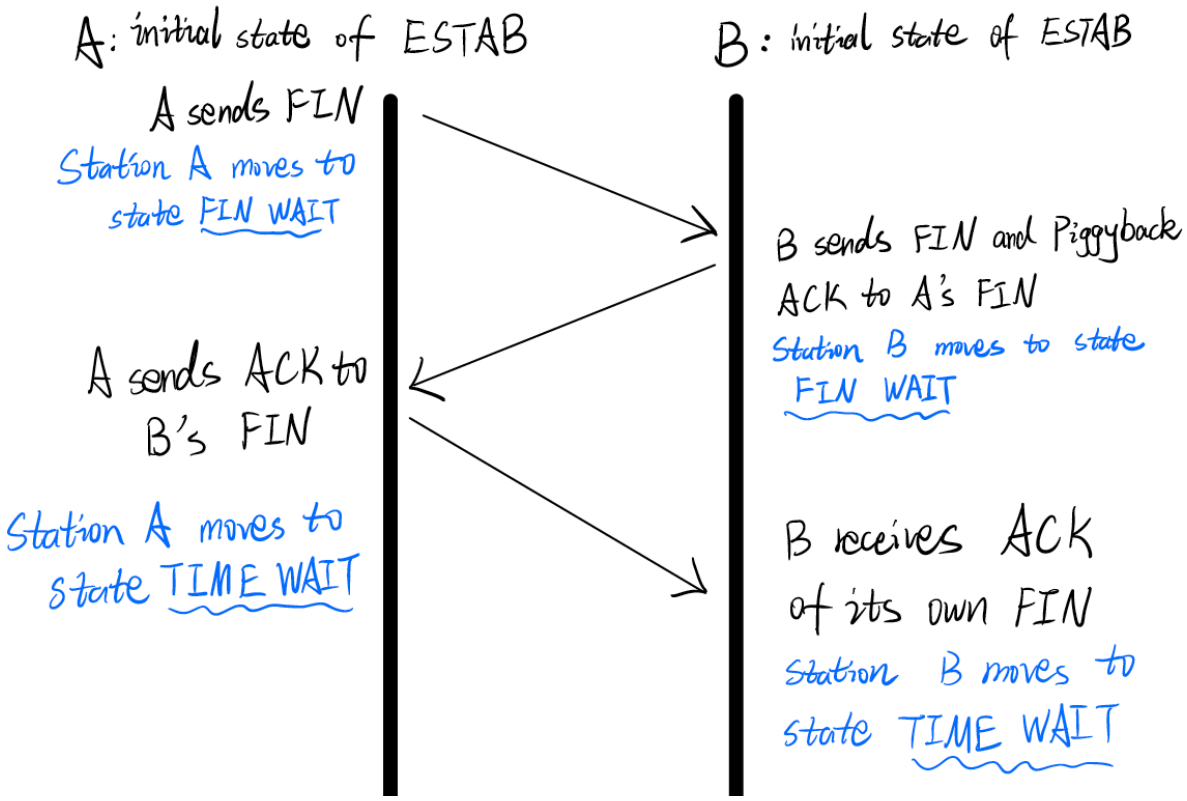
1) [24 points] Refer to the TCP state machine in the class notes (given below). Assume that a TCP connection between stations A(client) and B(server) has been in use. This connection was established using an active-passive open. Explain how the connection can be closed using an active close initiated by the client. To help you explain draw the series of segments exchanged during the active close for each of the three possible paths through the state machine (from stat ESTAB to state CLOSED).

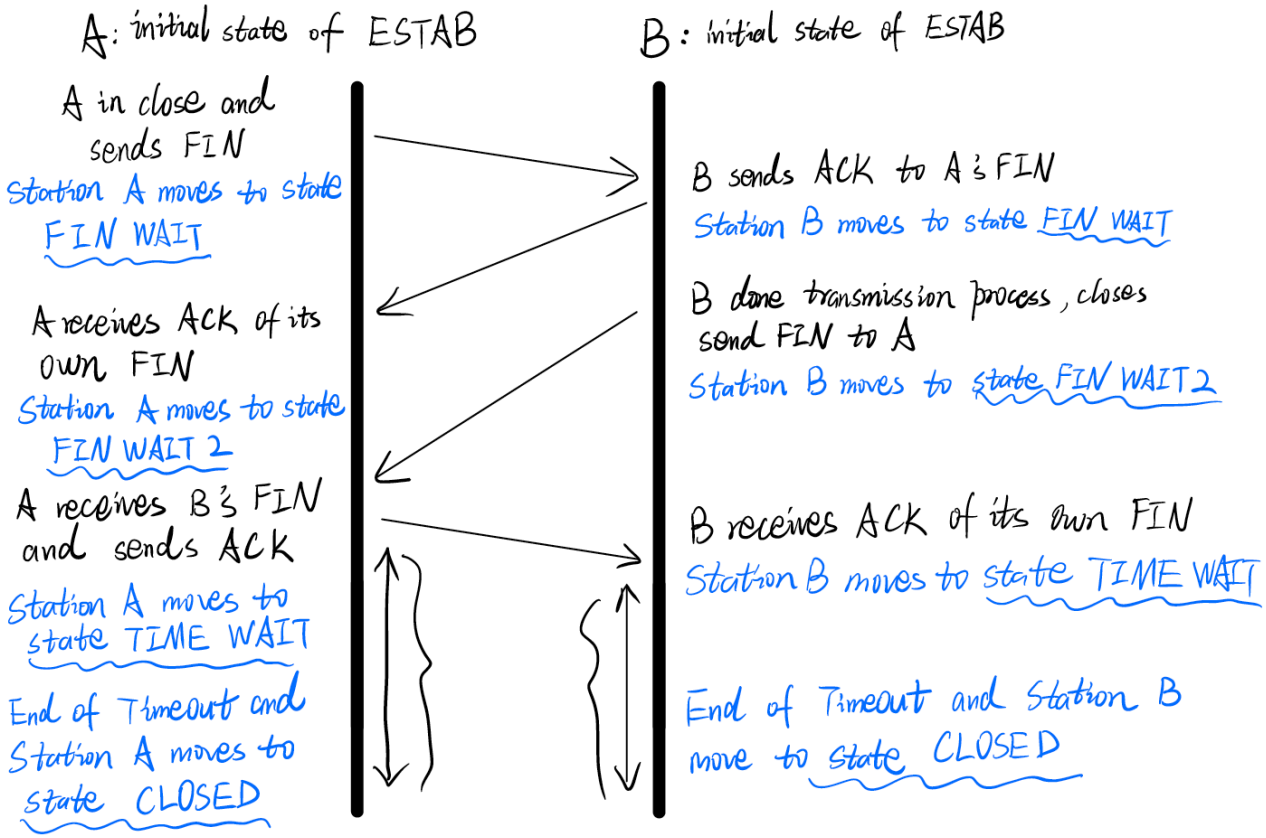


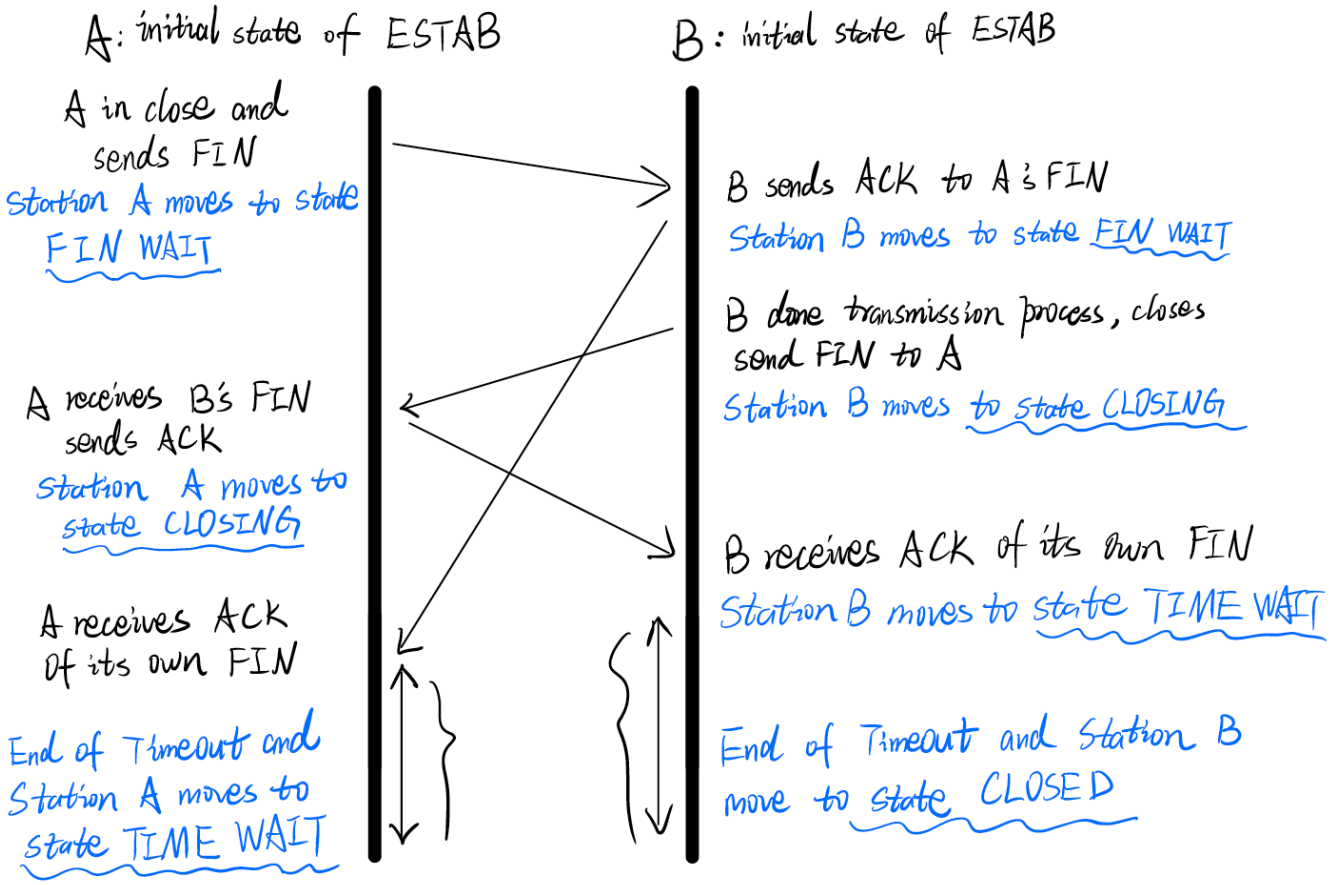
To indicate the type of diagram desired a SAMPLE DIAGRAM of the desired type is shown below the state machine. If you happen to be interested the sample diagram is for and active-active (peer-peer) open

SAMPLE DIAGRAM



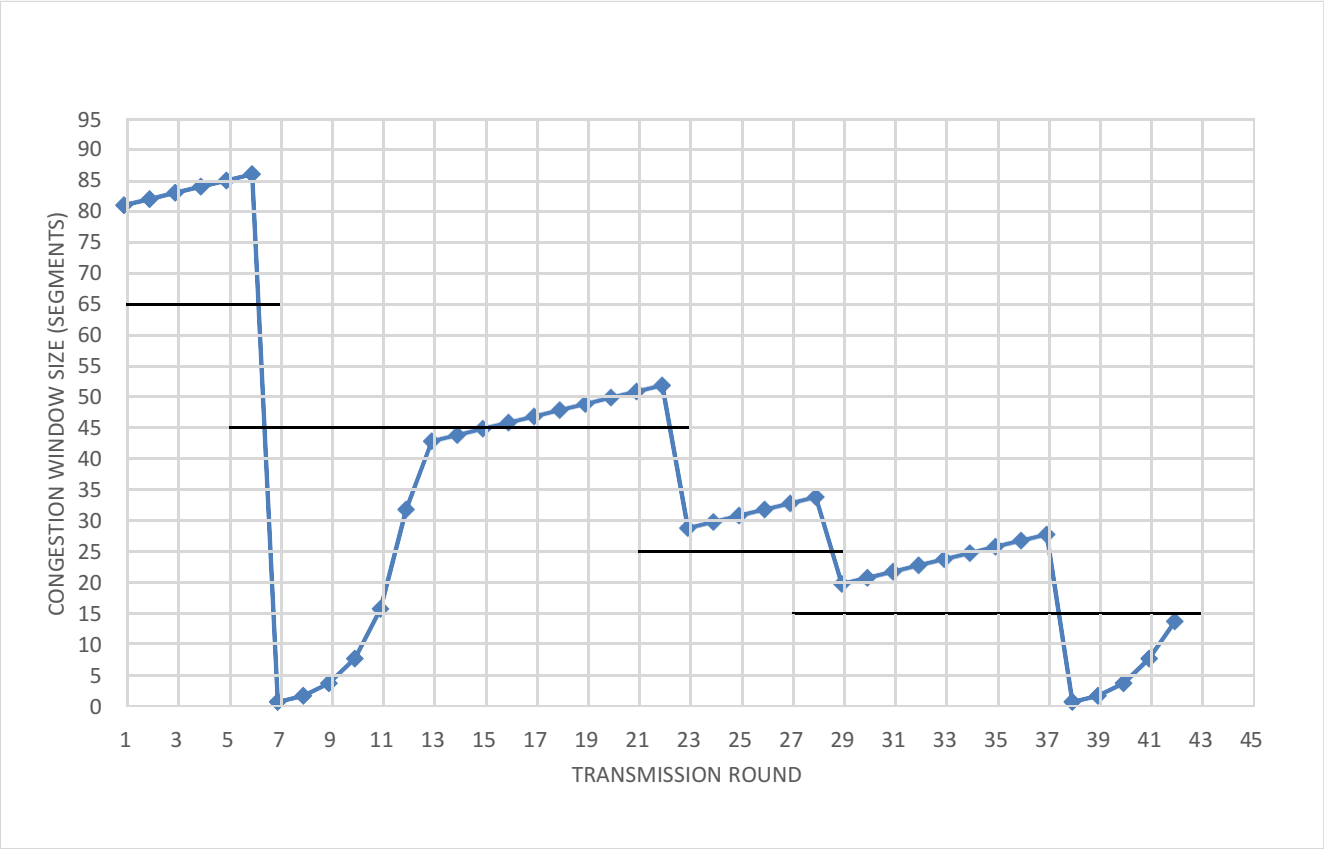

A: initial state of ESTAB
B: initial state of ESTAB
A sends FIN
Station A moves to state FIN WAIT
B sends FIN and Piggyback ACK to A's FIN
Station B moves to state FIN WAIT
A sends ACK to B's FIN
Station A moves to state TIME WAIT
B receives ACK of its own FIN
Station B moves to state TIME WAIT


A: initial state of ESTAB
B: initial state of ESTAB
A in close and sends FIN
Station A moves to state FIN WAIT
B sends ACK to A's FIN
Station B moves to state FIN WAIT
A receives ACK of its own FIN
Station A moves to state FIN WAIT 2
B done transmission process, closes send FIN to A
Station B moves to state FIN WAIT 2
A receives B's FIN and sends ACK
Station A moves to state TIME WAIT
B receives ACK of its own FIN
Station B moves to state TIME WAIT
End of Timeout and Station A moves to state CLOSED
End of Timeout and Station B move to state CLOSED


A: initial state of ESTAB
B: initial state of ESTAB
A in close and sends FIN
Station A moves to state FIN WAIT
B sends ACK to A's FIN
Station B moves to state FIN WAIT
B done transmission process, closes send FIN to A
Station B moves to state CLOSING
A receives B's FIN sends ACK
Station A moves to state CLOSING
B receives ACK of its own FIN
Station B moves to state TIME WAIT
A receives ACK Of its own FIN
End of Timeout and Station A moves to state TIME WAIT
End of Timeout and Station B move to state CLOSED

2) Consider the figure shown below. Assuming TCP (Reno) is using congestion control (slow start and congestion avoidance modes as discussed in your text and in class) answer the following questions. Assume the system was running before the sample window length data shown in the plot below was collected. Assume the first value of ssthresh shown on the diagram is 64.
Point 1 is assumed to be at the beginning of interval 1. If a failure that causes transition between modes occurs between two points the first point is considered to be part of the pre-failure mode, the second point to be part of the post failure mode. For other transitions between modes (slow start, congestion avoidance) one point is a part of both modes. It is both the end of one mode and the beginning of the other. For example, point 13 below would be part of both modes.

For each question you should provide a short discussion justifying your answer.



a) **[4 points]** Identify the periods of time when TCP collision avoidance is operating
When collision avoidance is operating, the length of the congestion window size will increase by one segment after each round. So, from the plot we can see collision avoidance is operating in rounds 1-6, rounds 13-22, rounds 23-28, rounds 29-37.

b) **[4 points]** Identify the periods of time when TCP slow start is operating
When slow start is operating, the congestion window will have two options. Either congestion window will increase by a factor of 2 or increase to the value of ssthresh. Therefore, slow start is operating in rounds 7-12 and rounds 38-41.

c) **[3 points]** During the 6^{th} transmission round, is segment loss detected by a triple duplicate ACK or by a timeout? What is the value of ssthresh
In my opinion, the loss is detected by a timeout. In collision avoidance mode, when the timeout happened, the slow start mode is restart and the congestion window size back to 1. The ssthresh is set to half the value of the congestion window when segment loss is detected. When loss is detected during transmission round 6, the value of ssthresh is 86/2 = 43.

d) **[3 points]** During the 22nd transmission round, is segment loss detected by a triple duplicate ACK or by a timeout?

In my opinion, the loss is detected by a triple duplicate ACK. In collision avoidance mode, If there were a timeout, the congestion window size would have dropped to 1.

e) **[10 points]** What is the value of ssthresh and the size of the congestion window during the 4th transmission round? During the 16th transmission round? During the 27th round? During the 35th round? During the 39nd round? NOTE: The value of the size of the congestion window during the 4th transmission round is the value at the beginning of the 4th transmission round

Explain how the values of ssthresh and the congestion window size are determined at each change in the value of ssthresh. The first value of ssthresh (at point 1) is given and need not be discussed.

round	Ssthresh	Congestion window size
4	64	84
16	43	46
27	29	34
35	20	26
39	14	2

(f) [2 points] During what transmission round is the 270th segment sent? The 1300th segment? Include the segments sent in transmission round 1.

81 + 82 + 83 (+ 84) = (246 < 270) 330 > 270. So segment 270 is sent in round 4.
81 + 82 + 83 + 84 + 85 + 86 + 1 + 2 + 4 + 8 + 16 + 32 + 43 + 44 + 45 + 46 + 47 + 48 + 49 + 50 + 51 + 52 + 29 + 30 + 31 + 32 + 33 + 34 + 20 + 21 + 22 + 23 + 24 = 1317 > 1300.
So segment 1300 is sent in round 33.

3) Consider two hosts transferring data using a TCP connection. Assume the connection between hosts A and B has already been made. The establishment of the TCP connection is not a part of this problem. Host A is sending a stream of application data to host B. The first octet of data A is sending to B in the transfer of data illustrated below is octet 3453. Each packet sent by A contains 550 octets of data. Host B is sending a different stream of application data to host A. The first octet of data B is sending to A in the transfer of data illustrated below is 7777. Each packet B is sending to A contains 400 octets of data.

a) [15 points] Fill in the sequence numbers and acknowledgement number on the diagram below.

b) [5 points] What are the two TCP error control mechanisms shown below? These are the mechanisms that help recover from loss of packets or ACKs.

Cumulative acknowledgements and fast retransmission.

c) [10 points] Give a step-by-step description of how the first mechanism you identified in b) operates using the diagram as an example to help in your explanation.

Cumulative acknowledgement:
A sends packet with SEQ 3453.
B receives and sends ACK 4003.
A sends SEQ 4003 about the same time.
B receives and sends ACK 4553.
A receives ACK 4003 and sends SEQ 4553 about the same time.
B receives the packet and sends ACK 5103.
A sends SEQ 5103 at the same time, but does not receive ACK 4553.
B receives the packet.

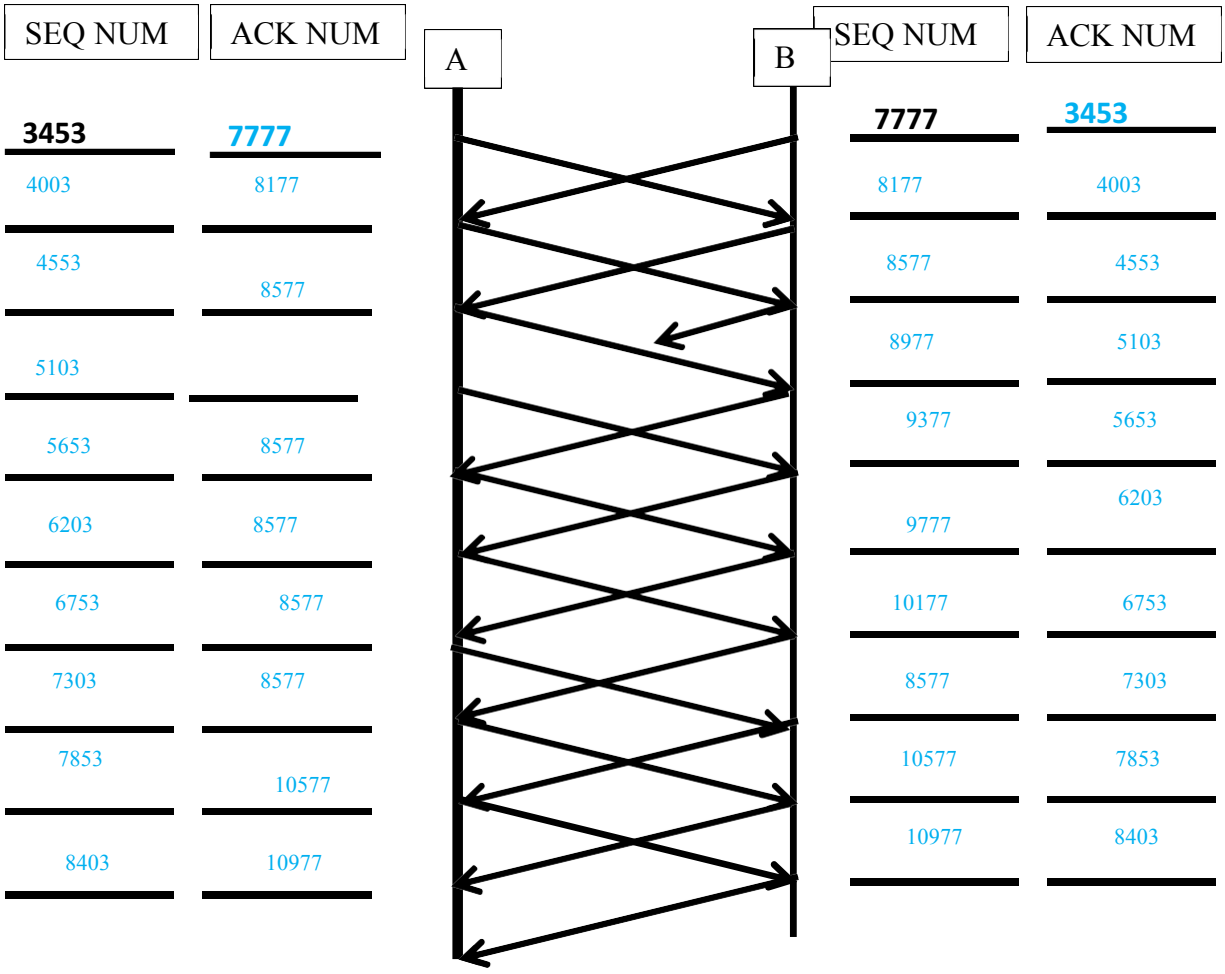
A receives the ACK 5103.
Then A knows that all has been received up to 5103 octets, and the ACK 4553 is lost.

A sends the first three packet. All of the three are received by B. Then, B sends the ACK for each of the three packets, but the second ACK is lost. However, when the third ACK is received by A. The third ACK tells that all octets up to the end of the third packet have been received, since the ACK is cumulative. So, A knows that the second packet has been received even without receiving the ACK.

d) [10 points] Give a step-by-step description of how the second mechanism you identified in b) operates using the diagram as an example to help in your explanation.

Fast retransmission: recover the loss of a packet. The SEQ 8577 from B to A is lost. When 3 duplicate ACKs have been received, the fast retransmission is applied.

B sends SEQ 8177.
A receives the packet and send ACK 8577.
B sends SEQ 8577.
And the packet is lost, no ACK is sent by A.
B sends SEQ 8977.
B receives ACK 8577 about the same time.
B sends SEQ 9377.
B receives ACK 8577 about the same time.
B sends SEQ 9777.
B receives ACK 8577 about the same time.
B sends SEQ 10177.
A receives the packet and send the ACK.
B has received 3 retransmissions of the ACK 8577, so retransmit SEQ 8577.
B receives ACK 10177.


SEQ NUM
ACK NUM
A
B
SEQ NUM
ACK NUM
3453
7777
4003
8177
4553
8577
5103
5653
8577
6203
8577
6753
8577
7303
8577
7853
10577
8403
10977
7777
3453
8177
4003
8577
4553
8977
5103
9377
5653
6203
9777
10177
6753
8577
7303
10577
7853
10977
8403

4) Consider the CIDR routing table shown below.

Destination	*Gateway*	*Mask*	*Interface*	
192.168.48.0	*****	***255.255.240.0***	***eth1***	***Line 1***
192.168.4.0	*****	***255.255.254.0***	***eth2***	***Line 2***
192.168.0.0	*****	***255.248.0.0***	***eth3***	***Line 3***
120.124.160.0	***192.168.0.2***	***255.255.224.0***	***eth3***	***Line 4***
192.156.32.0	***128.168.48.1***	***255.255.255.0***	***eth1***	***Line 5***
0.0.0.0	***192.168.200.12***	***0.0.0.0***	***eth4***	***Line 6***

a) **[5 points]** Is the forwarding table (routing table) above optimized so that the first match found is the "best" match? Explain why or why not

The forwarding table above is not optimized. The best match one is the longest prefix matching, which means to find the longest prefix matching with incoming packet's destination IP, and forward the packet to corresponding next hop. So, these lines need to be arranged in descending order of the mask.

b) **[15 points]** For each address in the table below which line in the routing table on the previous page is used, which interface is the packet sent through, and what the IP address of the host the packet will be sent to in the Ethernet layer? Fill in the table below

Destination address	*Line in forwarding table*	*Interface*	*Next hop IP address*
192.168.5.55	Line 2	Eth2	192.168.5.55 (directed delivery)
192.168.6.3	Line 3	Eth3	192.168.6.3 (directed delivery)
192.168.55.12	Line 1	Eth1	192.168.55.12 (directed delivery)
192.156.33.1	Line 6	Eth4	192.168.200.12
120.124.160.12	Line 4	Eth3	192.168.0.2