Computer Networking

Assignment 8

**## Problems of Chapter 3:**

P40. Consider Figure 3.61. Assuming TCP Reno is the protocol experiencing the behavior shown above, answer the following questions. In all cases, you should provide a short discussion justifying your answer.

a. Identify the intervals of time when TCP slow start is operating. **The intervals of time is 1-6 and 23-26.**

b. Identify the intervals of time when TCP congestion avoidance is operating. **The interval of time congestion avoidance is 6-23.**

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

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

e. What is the initial value of ssthresh at the first transmission round? **The initial value of ssthresh at the first transmission round is 32.**

f. What is the value of ssthresh at the 18th transmission round? **The value of ssthresh at the 18th transmission round is 21.**

g. What is the value of ssthresh at the 24th transmission round? **The value of ssthresh at the 24th transmission round is 13.**

h. During what transmission round is the 70th segment sent? **The transmission round to which the 70th segment is sent is 7.**

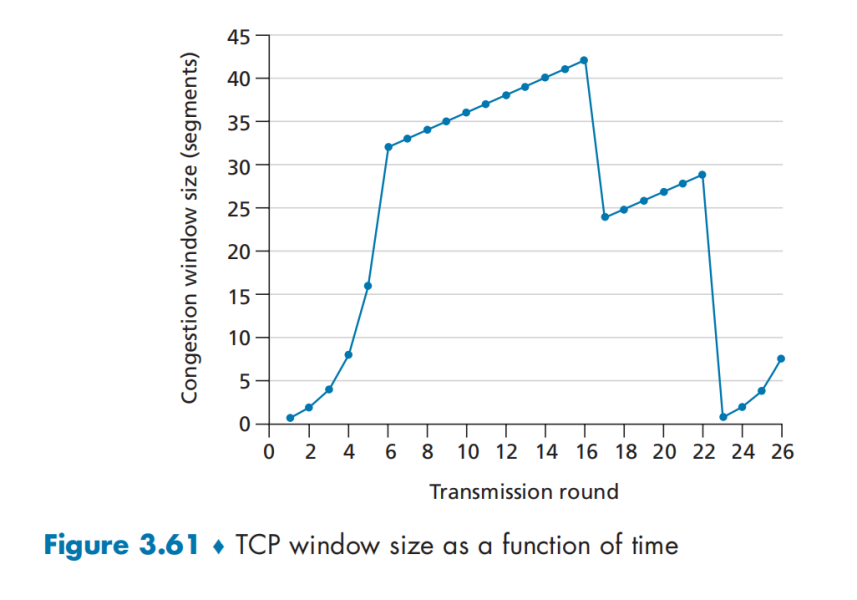
1. Assuming a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, what will be the values of the congestion window size and of ssthresh? **If a packet loss is detected after the 26th round by the receipt of a triple duplicate ACK, then the values of the congestion window size and ssthresh would be 4.**

j. Suppose TCP Tahoe is used (instead of TCP Reno), and assume that triple duplicate ACKs are received at the 16th round. What are the ssthresh and the congestion window size at the 19th round?

**The ssthresh and congestion window size the 19th round would be 1 and transmission round is 21.**

k. Again suppose TCP Tahoe is used, and there is a timeout event at 22nd round. How many packets have been sent out from 17th round till 22nd round, inclusive?

**52 packets are sent out from the 17th round till the 22nd round.**



P44. Consider sending a large file from a host to another over a TCP connection that has no loss.

a. Suppose TCP uses AIMD for its congestion control without slow start. Assuming cwnd increases by 1 MSS every time a batch of ACKs is received and assuming approximately constant round-trip times, how long does it take for cwnd increase from 6 MSS to 12 MSS (assuming no loss events)?

**Since we start with a CWND = 6,**

**After 1 RTT the CWND = 7.**

**After 2 RTTs, the CWND = 8**

**After 3 RTTs, the CWND = 9**

**After 4 RTTs, the CWND = 10**

**After 5 RTTs, the CWND = 11**

**After 6 RTTs, the CWND = 12**

**Thus, increasing CWND = 6 to CWND =12 using AIMD takes 6 RTTs.**

b. What is the average throughput (in terms of MSS and RTT) for this connection up through time = 6 RTT?

**Total segments sent from 6 RTTs:**

**6+7+8+9+10+11 = 51**

**51 MSS/ 6 RTT = 8.5 MSS/RTT**

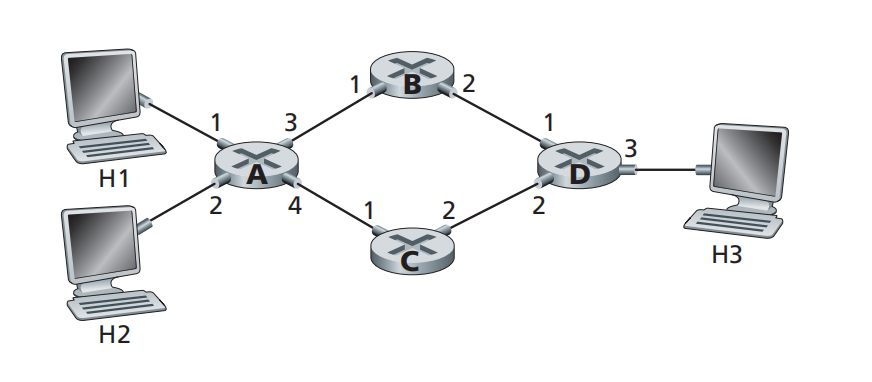
**## Problems of Chapter 4:**

**P1. Consider the network below.**

1. Show the forwarding table in router A, such that all traffic destined to host H3 is forwarded through interface 3.

|  |  |
| --- | --- |
| **Destination Address** | **Interface** |
| **H3** | **3** |

b. Can you write down a forwarding table in router A, such that all traffic from H1 destined to host H3 is forwarded through interface 3, while all traffic from H2 destined to host H3 is forwarded through interface 4? (*Hint:* This is a trick question.)



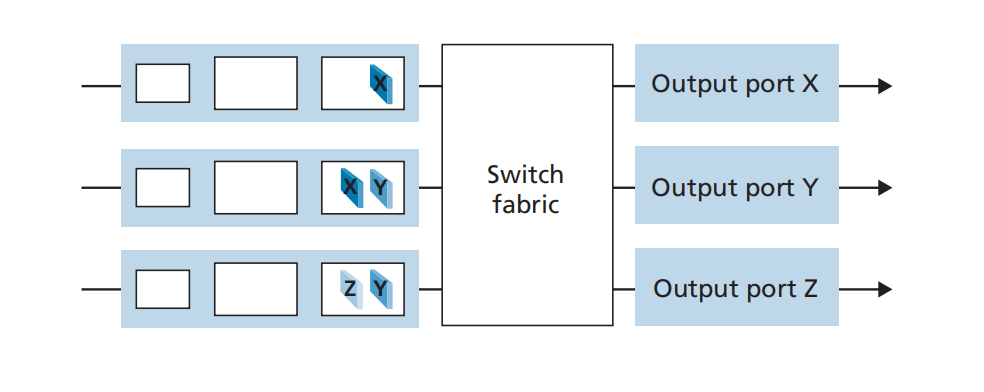
**There is no such forwarding table since forwarding rule is based on destination address only.**

**P4. Consider the switch shown below.**

Suppose that all datagrams have the same fixed length, that the switch operates in a slotted, synchronous manner, and that in one time slot a datagram can be transferred from an input port to an output port. The switch fabric is a crossbar so that at most one datagram can be transferred to a given output port in a time slot, but different output ports can receive datagrams from different input ports in a single time slot.

What is the minimal number of time slots needed to transfer the packets shown from input ports to their output ports, assuming any input queue scheduling order you want (i.e., it need not have HOL blocking)?

What is the largest number of slots needed, assuming the worst-case scheduling order you can devise, assuming that a non-empty input queue is never idle?



**The minimal number of time slots needed is 3:**

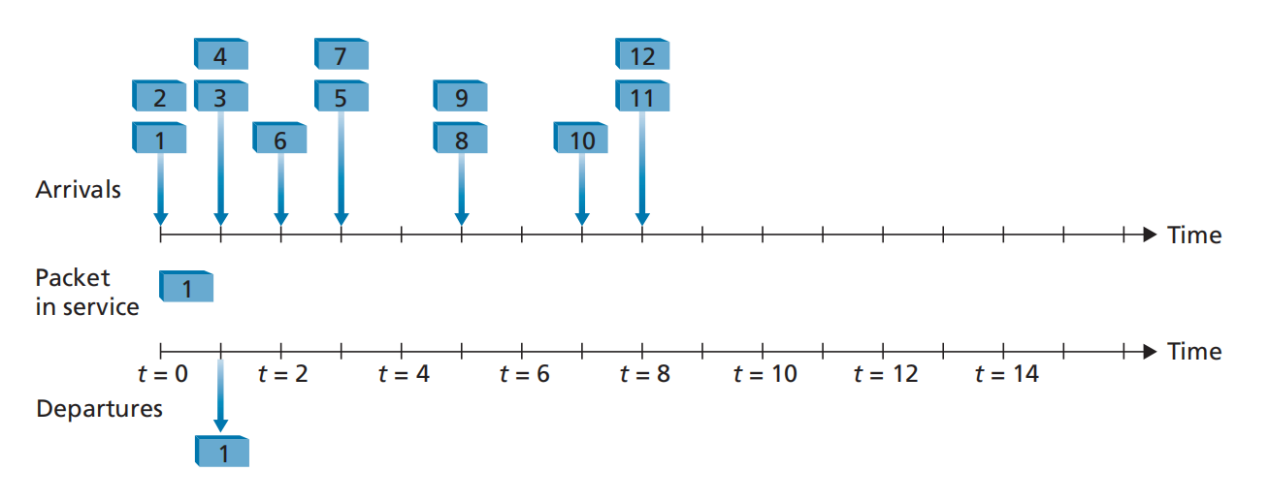
**Slot 1- send X in top input queue, send Y in middle input queue.**

**Slot 2- send X in middle input queue, send Y in bottom input queue.**

**Slot 3- send Z in bottom input queue.**

**The largest number of slots needed based on the two assumptions would be still be 3 slots. This is due to the fact that we can see that the first time slot is always made up of transmitting X in the top input queue and Y in either the middle or bottom input queue. The second time slot is always available for sending two further datagrams, and the third time slot is available for sending the last datagram. The worst situation would need 4 time slots if X was the first datagram in the bottom input queue.**

**P6. Consider the figure below. Answer the following questions:**



a. Assuming FIFO service, indicate the time at which packets 2 through 12 each leave the queue. For each packet, what is the delay between its arrival and the beginning of the slot in which it is transmitted? What is the average of this delay over all 12 packets?

**In the FIFO (First In First Out) Service, the packets enter the queue first (arrival time) and leaves the queue first (leaving time).**

**The following table is used to calculate the delay:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Packet** | **Arrival time** | **Leaving time** | **Delay time (leaving time - arrival time)** |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 1 |
| 3 | 1 | 2 | 1 |
| 4 | 1 | 3 | 2 |
| 5 | 3 | 5 | 2 |
| 6 | 2 | 4 | 2 |
| 7 | 3 | 6 | 3 |
| 8 | 5 | 7 | 2 |
| 9 | 5 | 8 | 3 |
| 10 | 7 | 9 | 2 |
| 11 | 8 | 10 | 2 |
| 12 | 8 | 11 | 3 |

Calculation:

Average delay for all 12 packets

= Total number of delays/ Total number of packets

=0+1+1+2+2+2+3+2+3+2/12

=1.92

**Thus, the average delay for all 12 packets using FIFO service is 1.92.**

1. Now assume a priority service, and assume that odd-numbered packets are high priority, and even-numbered packets are low priority. Indicate the time at which packets 2 through 12 each leave the queue. For each packet, what is the delay between its arrival and the beginning of the slot in which it is transmitted? What is the average of this delay over all 12 packets?

**In the priority queue service, we can assume that the odd-numbered (1,3,5,7,9) packets are high priority and even numbered (2,4,5,8,10, and 12) packets are low priority. Thus, the packets with higher priority are transmitted before the packets with lower priority in the priority queue service.**

**The following table data is used to calculate the delay:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Packet** | **Arrival time** | **Leaving time** | **Delay time (leaving time - arrival time)** |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 2 | 2 |
| 3 | 1 | 1 | 0 |
| 4 | 1 | 6 | 5 |
| 5 | 3 | 4 | 1 |
| 6 | 2 | 7 | 5 |
| 7 | 3 | 3 | 0 |
| 8 | 5 | 9 | 4 |
| 9 | 5 | 5 | 0 |
| 10 | 7 | 10 | 3 |
| 11 | 8 | 8 | 0 |
| 12 | 8 | 11 | 3 |

Calculation:

Average delay for all 12 packets

=Total number of delays / Total number of packets

=0+2+0+5+1+5+0+4+0+3+0+3/12

=1.92

**Thus, the average delay for all 12 packets using priority queue service is 1.92.**

c. Now assume round robin service. Assume that packets 1, 2, 3, 6, 11, and 12 are from class 1, and packets 4, 5, 7, 8, 9, and 10 are from class 2. Indicate the time at which packets 2 through 12 each leave the queue. For each packet, what is the delay between its arrival and its departure? What is the average delay over all 12 packets?

**In the round robin service, we can assume that packets 1,2,3,6,11, and 12 are from one group, and packets 4,5,7,8, 9, and 10 are from another group. A packet from each group leaves the queue alternatively in a round robin service.**

**The following table data is used to calculate the delay:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Packet** | **Arrival time** | **Leaving time** | **Delay time (leaving time - arrival time)** |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 2 | 2 |
| 3 | 1 | 4 | 3 |
| 4 | 1 | 1 | 0 |
| 5 | 3 | 3 | 0 |
| 6 | 2 | 6 | 4 |
| 7 | 3 | 5 | 2 |
| 8 | 5 | 7 | 2 |
| 9 | 5 | 9 | 4 |
| 10 | 7 | 11 | 4 |
| 11 | 8 | 8 | 0 |
| 12 | 8 | 10 | 2 |

Calculation:

Average delay for all 12 packets

=Total number of delays / Total number of packets

=0+2+3+0+0+4+2+2+4+4+0+2/12

=1.92

**Thus, the average delay for all 12 packets using round robin service is 1.92.**

1. Now assume weighted fair queuing (WFQ) service. Assume that odd numbered packets are from class 1, and even-numbered packets are from class 2. Class 1 has a WFQ weight of 2, while class 2 has a WFQ weight of 1. Note that it may not be possible to achieve an idealized WFQ schedule as described in the text, so indicate why you have chosen the particular packet to go into service at each time slot. For each packet what is the delay between its arrival and its departure? What is the average delay over all 12 packets?

**In the weighted fair queuing (WFQ) service, we can assume that odd-numbered packets 1,3,5,7,9, and 11 are from class 1 and even-numbered packets 2, 4, 6, 8, 10, and 12 are from class 2. During WFQ service, 2 packets of class 1 leaves in a turn and then then the turn comes to class 2 and only one packet may leave in one turn.**

**The following table data is used to calculate the delay between its arrival and the beginning of the slot in which it is transmitted by using the given figure:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Packet** | **Arrival time** | **Leaving time** | **Delay time (leaving time - arrival time)** |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 2 | 2 |
| 3 | 1 | 1 | 0 |
| 4 | 1 | 5 | 4 |
| 5 | 3 | 3 | 0 |
| 6 | 2 | 7 | 5 |
| 7 | 3 | 4 | 1 |
| 8 | 5 | 9 | 4 |
| 9 | 5 | 6 | 1 |
| 10 | 7 | 10 | 3 |
| 11 | 8 | 8 | 0 |
| 12 | 8 | 11 | 3 |

Calculation:

Average delay for all 12 packets

=Total number of delays / Total number of packets

=0+2+0+4+0+5+1+4+1+3+0+3/12

=1.92

**Thus, the average delay for all 12 packets using the weighted fair queuing service is 1.92.**

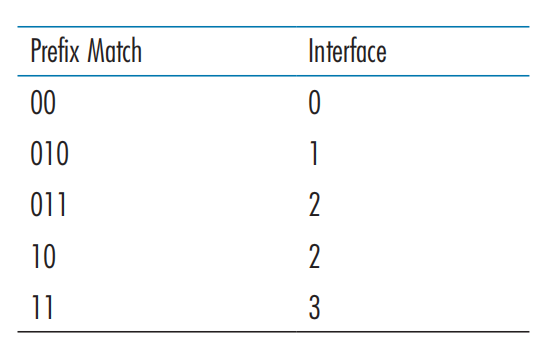
**Note: 1.92 is rounded from the real result of the correct average value which is 1.9166.**

e. What do you notice about the average delay in all four cases (FIFO, RR, priority, and WFQ)?

**The average delay for all 12 packets in all the services are the same but they all have different delay of packet values.**

**P9. Consider a datagram network using 8-bit host addresses. Suppose a router**

**uses longest prefix matching and has the following forwarding table:**

****

For each of the four interfaces, give the associated range of destination host addresses and the number of addresses in the range.

|  |  |
| --- | --- |
| **Destination Address Range** | **Link Interface** |
| 00**000000 --** 00**111111** | 0 |
| 010**00000 --** 010**11111** | 1 |
| 011**00000 --** 011**11111** | 2 |
| 10**000000 --** 10**111111** | 2 |
| 11**000000 --** 11**111111** | 3 |

**Number of addresses for interface 0 = 2^6 = 64**

**Number of addresses for interface 1 = 2^5 = 32**

**Number of addresses for interface 2 = 2^5 + 2^6= 32 + 64 = 96**

**Number of addresses for interface 3 = 2^6 = 64**

**## Lab**

`https://www.github.com/network-whu/lab/`

- 6.Wireshark\_TCP.docx