ECE 654: Computer Networks
Final Exam

8 December 2004.

- The exam consists of two parts. Answer **all** questions of **PART I** and 4 out of 6 questions of **PART II**
- Always be sure to answer each question **concisely** but **precisely**!
- All questions of **PART I** have 5 points each and the questions of **PART II** 25 points.

PART I

1. What is the Address Resolution Protocol (ARP) and how does it work?

   The ARP is a protocol for finding the MAC address that corresponds to a specific IP address. Suppose that a sender wants to send a packet to IP address 195.42.12.5. It broadcasts a message requesting the MAC address of that corresponds to 195.42.12.5. If 195.42.12.5 is on the same local area network, when it sees the message, it responds back with its MAC address. If 195.42.12.5 is not on the same local area network, then either there is a default router (with a known MAC address), or the router responds with its own MAC address.

2. Suppose you are defining a new version of the TCP header that accommodates a round-trip time (RTT) of 200ms, a network bandwidth of 100Mbps, and a maximum segment lifetime of 150 seconds. How many bits will you need for the advertised window field and for the sequence number field? Give the exact number of bits these two fields need.

   *In order to keep the pipe full, the window should be able to open to Bandwidth*RTT/8= 100Mbps * 200msec/8 = 2.5MBytes. Therefore the window should have log$_2$2.5M= 22 bits. During a 150 second interval the source can transmit up to 150sec *100Mbps/8= 1.875 GBytes, which requires 31 bits.*

3. Why does the maximum packet lifetime, *T*, have to be large enough to ensure that not only the packet but also its acknowledgements have vanished?

   Suppose that we only allow for the time that a packet needs to vanish. Suppose that just before it vanishes, packet p reaches the destination which responds with the acknowledgement. Now, suppose that the source fills safe that it can reuse p and does so. After a while the p acknowledgment arrives and the sender thinks that p has been safely received.

4. What services does the User Datagram Protocol (UDP) offer to justify its existence?

   UDP provides an interface for the applications as well as multiplexing through ports.

5. A computer on a 6-Mbps network is regulated by a token bucket. The token bucket is filled with a rate 1 Mbps. The bucket is initially filled to capacity with 1 Mb. How long can the computer transmit at the full 6 Mbps?

   The net outflow from the token bucket is 5Mbps. As a result, the time it takes for the full bucket to empty is 1Mb/5Mbps=0.2sec. Thus, during the first 0.2 seconds the computer transmits at the maximum 6-Mbps rate and then it switches to 1-Mbps.

6. In general the round trip time (RTT) is a random variable that depends on a number of parameters (propagation delay, congestion, etc). Often, RTT also has high variability making

the selection of the TCP TimeOut a difficult task. What is a mechanism (that has also been implemented is some versions of TCP) for addressing this problem?

The mechanism is the fast retransmit which works as follows. When the receiver receives a packet that is not expecting, it responds with an acknowledgement of the last correctly received packet. When the transmitter receives the same ACK three consecutive times, it interprets that the next packet has been lost and thus resends the corresponding packet. In this case the TimeOut is set to a long period of time to avoid premature retransmissions.

7. What are the problems that the TCP may face with the emergence of high speed networks and how can these be addressed?

(a) The window size of 64KB is very small to keep the pipe full therefore TCP will underutilize the link.

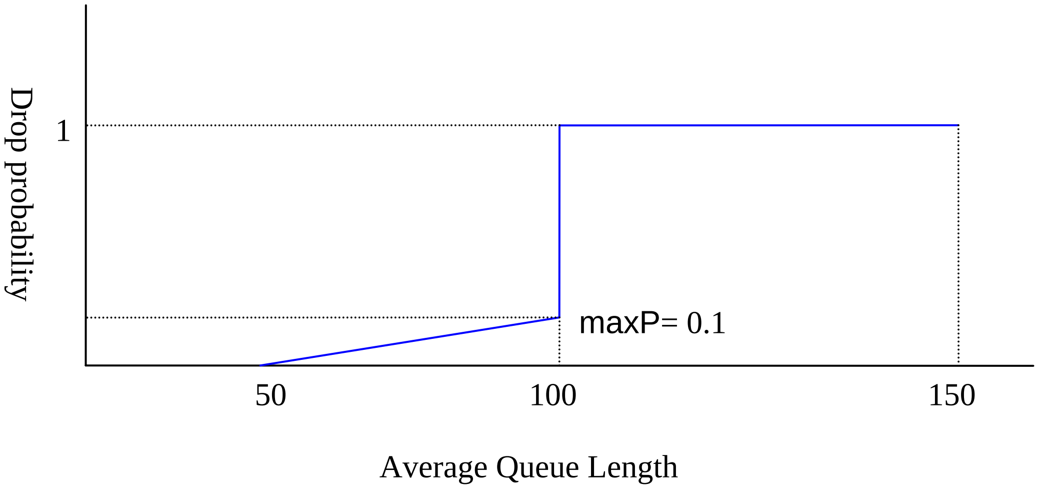
(b) Also, all possible sequence numbers will run out very quickly and as a result a sequence number may need to be reused before the previous packet that used it has vanished from the network.

A simple solution is to use a multiplier extension, so rather than each byte having its own sequence number, a group of bytes will share the same sequence number.

8. Why the TCP congestion control algorithm is not appropriate for the Ad-Hoc Networks? Briefly describe how the Ad-Hoc Transmission Control Protocol (ATCP) works.

TCP interprets packet loss as congestion and slows down. On the other hand in Ad-Hoc networks, a packet can be lost due to channel errors. In this case the node should try to transmit at full speed, not slow down. The ATCP is a sublayer between the IP and transport layer. When it detects a packet loss due to the channel, it simply sets the transmission window of TCP to 0, until it manages to correctly deliver the packet.

9. Consider the Random Early Detection RED algorithm with MinThreshold = 50, MaxThreshold= 100, maximum buffer size =150 and maxP= 0.1. Draw the curve that gives the packet drop probabilities for all values of average queue lengths. Suppose that a packet arrives when the average queue length is 60, what is the minimum probability that it will be dropped? What is the minimum probability that it will be dropped if the average queue length is 45?



If the average queue length is 60, the minimum drop probability will be 0.02. If the average queue length is 45, then them minimum drop probability will be 0. (These are the minimum drop probabilities because the queue length refers to the average and thus it is possible that even the average queue length is 60 or 45, at that moment, the buffer could be ful)l.

10. What are the Denial of Service attacks and what are possible approaches that attackers may use to exploit the network so they can succeed?

    DoS attacks are attacks that try to shut down the target (victim) preventing authorized users from using any of the resources. Such attacks either create an artificial congestion or exploit the operation of some protocols (logic or software attacks).

11. Suppose that a receiver in its effort to control the bursts from the transmitter it delays sending ACKs until it has enough empty buffers. What is a possible problem with such a strategy? How can these problems be resolved?

    Delaying the acknowledgements may be interpreted as lost packet and/or congestions thus it will trigger unwanted actions (e.g., packet retransmissions) at the sending side. A possible solution is to allow the receiver to explicitly tell the transmitter how much data to send (e.g., use a flow control window as in TCP).

12. Suppose that a 1-Mbps link is shared by 5 well behaving TCP and 5 UDP sources. The maximum window of the TCP sources is 50KB. The UDP sources transmit at a rate 25KB per second. What is the maximum transmission rate that can be achieved by the TCP and UDP sources?

    *Each of the UDP sources transmits at a rate of 25*8=200Kbps, therefore all 5 of them can capture the1-Mbps capacity of the link. The TCP will mostly see the link as congested and as a result they will have very small window unable to transmit anything! TCP sources will experience denial of service.*

13. What is the difference between IP and MPLS routing?

    IP routing is performed based on a globally unique IP address whereas in MPLS the routing is performed based on a locally unique label; however, the same label can be used in various routers for different flows.

14. A VoIP application is generating voice samples at a rate 8KBytes per second, puts them in 30msec packets and sends them over a 2Gbps link with propagation delay 100msec. What is the minimum, maximum and average packetization delay?

    Minimum packetization delay is 0, maximum is 30 ms and the average is 15 ms (assuming uniform sampling).

15. Congratulations, you have just hit the jackpot! You have the option to select any of the questions 1 through 14 and you will receive a bonus equal to the grade that you received in the selected question.

    Any question you are confident that you got full marks.

16. In an IETF meeting concerning VoIP, a researcher from the WasteBandwidth.com Co Ltd. explains that generating 30msec packets is a big problem for their applications especially when the destination is far away or when the path includes congested links. For this reason, he suggests a new standard where voice samples are sent immediately to the destination. How would you respond?

Sending 1 byte of information with 40 bytes of header (overhead) wastes bandwidth and makes the congestion problem even worse. They should look for a solution to their problem elsewhere.

17. In the context of security, what is an intrusion detection system (IDS) and what are some general approaches that can be used to for it?

    An IDS is a system that detects when an unauthorized person (intruder) has gained access to some of the private resources of a network. Some general approaches that can be used for detecting such incidents include statistical technique and pattern matching.

18. The GiveMeMoney (GMM) Bank Ltd., uses RSA and has picked p=3 and q=5. As a results, n=pq=15 and z=(p-1)(q-1)= 8. Then GMM picks e= 3 and d= 11. Subsequently, it publishes (15, 3) on its website and keeps (15, 11) secrete. Now you want to transfer £7 to the account number 12. How are you going to send it (Use the format <Account Number=*, Amount= *>)? Suppose that GMM received the message <Account Number= 3, Amount= 13> from you. How does it interpret it?

    12^3 mod 15 = 3 and 7^3mod 15 = 13, <Account Number= 3, Amount= 13>

    3^{11}mod 15 = 12 and 13^{11}mod 15 = 7, Account number= 12 and amount= 7.

19. Is there a way for Bob and Alice to agree on a common key over a network that is observed by Trudy but without her being able to interpret it? If yes, is it possible for Trudy to break the protocol if she is able to intercept messages? If no, what Bob and Alice need to do in order to securely exchange a common key (other than meeting each other or talking on the phone)?

    Yes two people can agree on a common private key over a network using the Diffie-Hellman Key Exchange Algorithm. Trudy can still break it pretending that she is either or Alice (man-in-the-middle attack)

20. What are the possible approaches for delivering quality of service guarantees in networked applications?

    Integrated and Differentiated Services, MPLS is also built over differentiated services.

21. Suppose that a source generates 1KB packets with average rate 900 packets/second according to a Poisson distribution. All packets traverse the same 3 routers each of which needs 1 msec to forward the packet from the incoming to the outgoing port. You run an experiment on the network and find out that the average delay between source and destination is significantly longer than the expected 3 msec. You consult with the system administrator who suggests that the problem should be the propagation delay. What do you think?

    No, the discrepancy is expected due to the queueing delay.

22. Suppose that the path from A to E goes though links AB with capacity 100Mbps and propagation delay 20ms, BC with capacity 50Mbps and delay 10ms, CD with capacity 1Gbps and delay 50ms and finally link DE with capacity 150Mbps and propagation delay 10ms. Currently node A has an estimate of the round trip delay RTT= 200ms. Subsequently it receives the following 5 RTT samples: 210ms, 220ms, 230ms, 240ms and 250ms. What is the new estimate of the RTT given that the weight given to the new samples is 1/8?

    For each update we use the formula RTT_{new}= (7 RTT_{old} + Sample)/8. As a result the RTT estimate is evaluated as follows:

RTT_old	*200*	*201*	*204*	*207*	*211*
Sample	*210*	*220*	*230*	*240*	*250*
RTT_new	*201*	*204*	*207*	*211*	*216*

23. For the network of question 22, suppose that the variance estimate is initially 20. How would you set the TCP timeout after the 5 samples are received?

Similarly to 22 we evaluate the new variance estimate using the formula D_{new}= (7D_{old}+ |RTT - Sample|)/8. The results are shown in the table below:

RTT_old	*200*	*201*	*204*	*207*	*211*
Sample	*210*	*220*	*230*	*sample 240*	*250*
RTT_new	*201*	*204*	*207*	*211*	*216*
D_old	*20.0*	*18.6*	*18.3*	*18.9*	*20.2*
D_new	*18.6*	*18.3*	*18.9*	*20.2*	*21.9*

*As a result the timeout is set to RTT + 4D = 216 + 4*22= 304 ms.*

24. Consider a TCP source that transmits the following sequence of packets:

Time (ms)	0	10	20	30	300	310	320
SeqNumber	1	2	3	4	3	4	5

It also received the following sequence of acknowledgements

Time (ms)	200	220	310	320	340
AckNumber	1	2	3	4	5

Assume that the initial RTT estimate (at time 0) is 100ms. Determine the new RTT estimate that results after the reception of ACK 5. Assume that the weight of each new sample is 1/8.

This question is similar to 22 except the acknowledgements of packets 3 and 4 should not be considered since it is not clear whether the received ACK is due to the first of the second packet. So, only ACKs 1, 2, and 5 should be considered. The result is shown in the table below:

ACK	*1*	*2*	*3*	*4*	*5*
RTT_old	*100*	*113*	*125*	*125*	*125*
Sample	*200*	*210*	*??*	*??*	*20*
RTT_new	*113*	*125*	*125*	*125*	*112*

25. In 10 years you are asked (as Department’s Alumni) to talk to new graduate students. Give one reason they should take this class (it will no longer be mandatory) and one reason for not taking it.

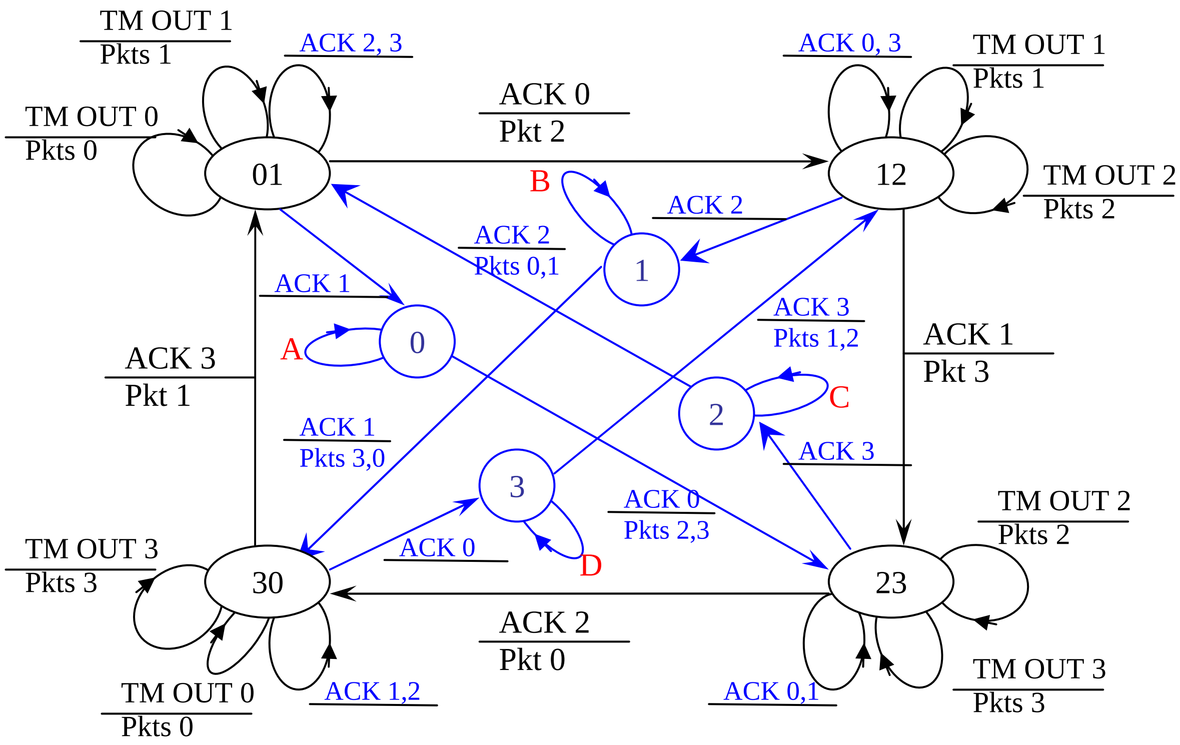
PART II

1. Imagine that a two-way handshake rather than a three-way handshake were used to set up a connection. In other words, the third message is not required. Are deadlocks now possible? Give an example or show that none exists.

Deadlocks are possible. For example, a packet arrives at A out of the blue, and A acknowledges it. The acknowledgement gets lost, but A is now open while B knows nothing at all about what has happened. Now the same thing happens to B, and both are open, but expecting different sequence numbers. Timeouts have to be introduced to avoid the deadlocks.

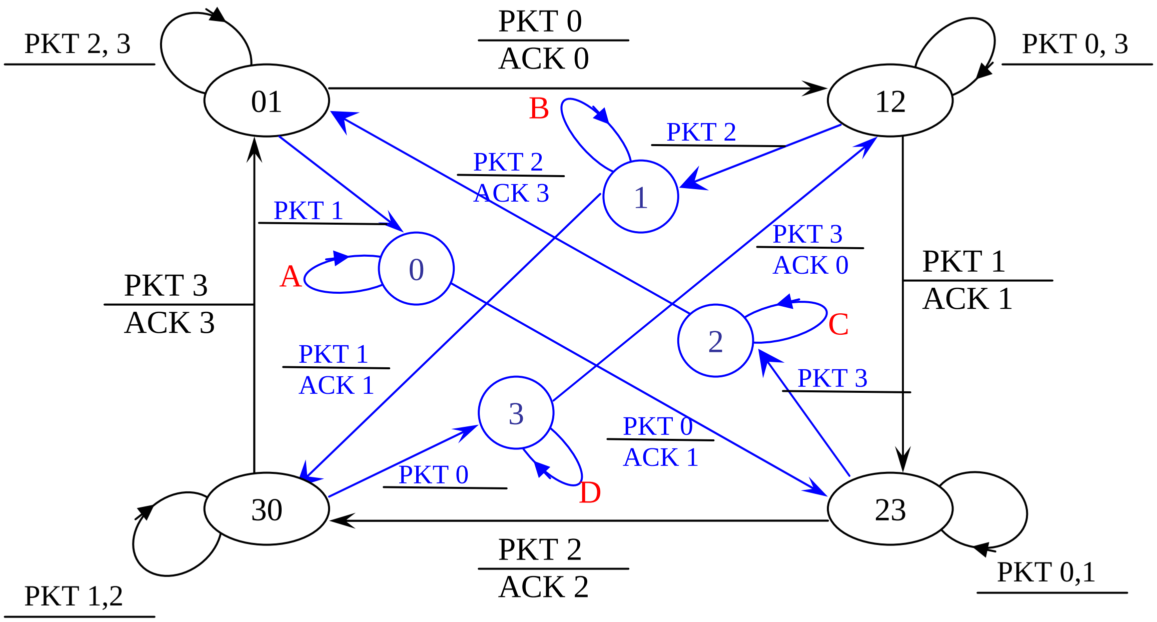
2. Draw the finite state machine for the transmitter and receiver of the selective repeat protocol when the frame sequence number is 2 bits long. Assume that the network cannot reorder or loose any acknowledgements. Also assume that the network can only loose packets with sequence numbers 0 and/or 1, but not the remaining packets.

Since we have 2 bit long sequence number, the maximum allowable window is 2. The state represents the sequence number of the unacknowledged packets, for example the state 01 means that the transmitter has sent packets 0 and 1 and it is waiting for the corresponding acknowledgments. Over each transition we write the event that triggers the transition, i.e., either a timeout event (TM OUT) or the reception of an acknowledgement packet (ACK). Finally, under the triggering event we write the action of the transmitter, i.e., send certain packets (Pkt xyz).



Due to space limitation in the figure we describe the self-loop transitions A,…,D here. A is triggered by either the reception of ACKs 1, 2, or 3 and no action is taken, or it is triggered by a TM OUT 0 event which causes a retransmission of pkt 0. B is triggered by either the reception of ACKs 0, 2, or 3 and no action is taken, or it is triggered by a TM OUT 1 event which causes a retransmission of pkt 1. C is triggered by either the reception of ACKs 0, 1, or 3 and no action is taken, or it is triggered by a TM OUT 2 event which causes a retransmission of pkt 2. Finally, D is triggered by either the reception of ACKs 0, 1, or 2 and *no action is taken, or it is triggered by a TM OUT 3 event which causes a retransmission of pkt 3.*

At the receiver side, we have



where again due to space limitation the self loops A,…,D indicate the arrival of a non expected packet and no action is taken.

3. Assume that TCP implements an extension that allows window sizes much larger than 64 KB. Suppose that you are using this extended TCP over a 1-Gbps link with round trip latency 100 msec to transfer a 10MB file. If TCP sends 1 KB packets (assuming no congestion and no lost packets)

   a. How many RTTs does it take until slow start opens the send window to 1 MB?

   RTT0=1KB, RTT1= 2KB, RTT2=4KB,… RTTn=2^nKB. To have 1MB=2^{10} window we need n=10. Therefore after 10 RTTs the window size will become 1MB.

   b. How many RTTs does it take to send the file?

   By the 10th RTT we have transmitted 1MB and the window is 1MB, therefore by the end of the 11th RTT we have transmitted 2MB and the window is now 2MB. Similarly during the next 3 RTTs we transmit another 2MB, 4MB and another 2MB, thus we need 14 RTTs to transmit the entire file.

   c. If the time to send the file is given by the number of required RTTs multiplied by the link latency, what is the effective throughput for the transfer? What percentage of the link bandwidth is utilized?

   *Effective throughput is the file size over the time needed i.e., 8*10Mb/(14*100ms)=53.3Mbps, which corresponds to a bandwidth utilization of 5.7%.*

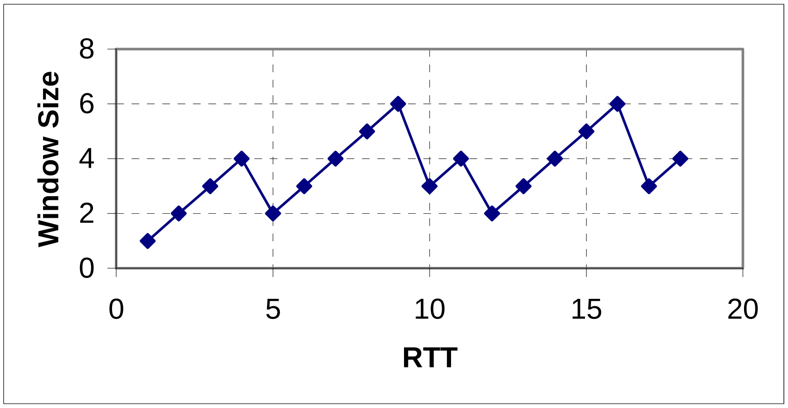
   d. Assuming that the threshold for getting into the congestion avoidance stage is 500KB. How long (number of RTTs) does it take to send the file?

After the 9th RTT the window size will become 512KB and we would have transmitted 511KB. After this point the window will start increasing linearly. A more aggressive approach is to increase the window by 512KB per round trip cycle. In this case, after the 10th RTT we would have transmitted 1MB and the window will start increasing by 512KB per RTT. Next after the next 5 RTTs, the sender will send 1MB, 1.5MB, 2MB, 2.5MB, and 2MB for the total of 10MB. Thus it requires 15 RTTs. A more conservative approach is to increase the window by a packet (1KB) per round trip cycle. In this case, the window will be 512KB, 513KB, …,530KB and thus we will require 28 RTTs.

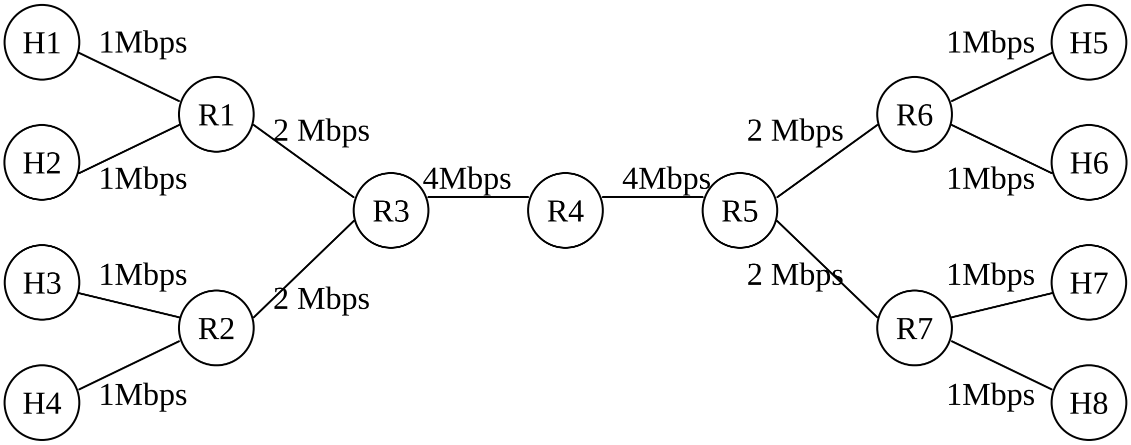
4. Consider an additive increase multiplicative decrease congestion control algorithm (window size increases linearly and it is halved when congestion is detected) (no slow start) that works in units of packets rather than bytes and that starts each connection with a congestion window equal to one packet. Give a detailed sketch of the algorithm. Assume that the delay is only due to propagation (infinite transmission capacity) and that when a group of packets is sent, only a single acknowledgement is returned. Plot the congestion window as a function of the round-trip times for the situation in which the following packets are lost: 9, 25, 30 and 49. For simplicity assume a perfect timeout mechanism that detects a lost packet exactly 1 RTT after it is transmitted.

RTT	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
Window	*1*	*2*	*3*	*4*	*2*	*3*	*4*	*5*	*6*
SeqNum	*1*	*2,3*	*4,5,6*	*7,8,9,10*	*9,10*	*11,12,13*	*14,15,16,17*	*18-22*	*23-28*

RTT	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*
Window	*3*	*4*	*2*	*3*	*4*	*5*	*6*	*3*	*4*
SeqNum	*25,26,27*	*28,29,30,31*	*30,31*	*32,33,34*	*35-38*	*39-43*	*44-49*	*49-51*	*52-55*



5. Consider the network shown below which has eight hosts H1,…,H8 and seven routers R1,…,R7 each of which is much faster than any of the links.



All links are full-duplex with bandwidths as shown in the figure. Show which routers can never be congested and which ones are more vulnerable to congestion. For the routers that can get congested, show a traffic patterns that causes congestion.

Router R4 can never be congested. The worst case scenarios are either H1,…,H4 want to send 4 Mbps to either of H5,…,H8 or vice versa. In either case, there is enough link capacity on both sides. All other routers can become congested.

- *Suppose H2, H3 and H4 want to send 1Mbps each to H1. In this case, R1 becomes congested (similar scenarios apply to R2, R6 and R7, e.g. suppose H1, H2, and H3 want to send 1Mbps each to H4. In this case, R2 becomes congested).*
- *Suppose that all H2,…,H8 want to send 1Mbps each to H1. In this case, R3 also becomes congested.*

6. Suppose that a router has three input flows and one output flow. It receives the packets listed in the Table below, all at about the same time, in the order listed, during a period in which the output port is busy but all queues are otherwise empty. Give the order in which the packets are transmitted, assuming

   a. Round robin

   b. Fair queueing

   c. Waited fair queuing with flow 2 having weight 2 and the other two flows having weight 1.

Packet	Size	Flow
1	100	1
2	100	1
3	100	1
4	100	1
5	190	2
6	200	2
7	110	3
8	50	3

- *Round Robin: 1 – 5 – 7 – 2 – 6 – 8 – 3 – 4*

- *Fair Queueing: Since all packets arrived at the same time, we don't need to scale the arrival time, so their transmission time will be given by $F_k = D_{k-1}+S_k$. The results of the calculation are shown in the table below:*

Packet	Size	Flow	$F_k = D_{k-1}+S_k$
1	100	1	*100*
2	100	1	*200*
3	100	1	*300*
4	100	1	*400*
5	190	2	*190*
6	200	2	*390*
7	110	3	*110*
8	50	3	*160*

Therefore packets would be transmitted in the following order: 1 – 7 – 8 – 5 – 2 – 3 – 6 – 4.

- *Weighted fair queueing with flow two weight= 2. In this case packets from flow two will be accelerated using $F_k = D_{k-1}+S_k/2$. The results are shown in the table below*

Packet	Size	Flow	$F_k = D_{k-1}+S_k$
1	100	1	*100*
2	100	1	*200*
3	100	1	*300*
4	100	1	*400*
5	190	2	*95*
6	200	2	*195*
7	110	3	*110*
8	50	3	*160*

Therefore packets would be transmitted in the following order: 5 – 1 – 7 – 8 – 6 – 2 – 3 – 4.