NAME:

Login name:

Computer Science 461
Midterm Exam
March 11, 2011
10:00-11:50pm

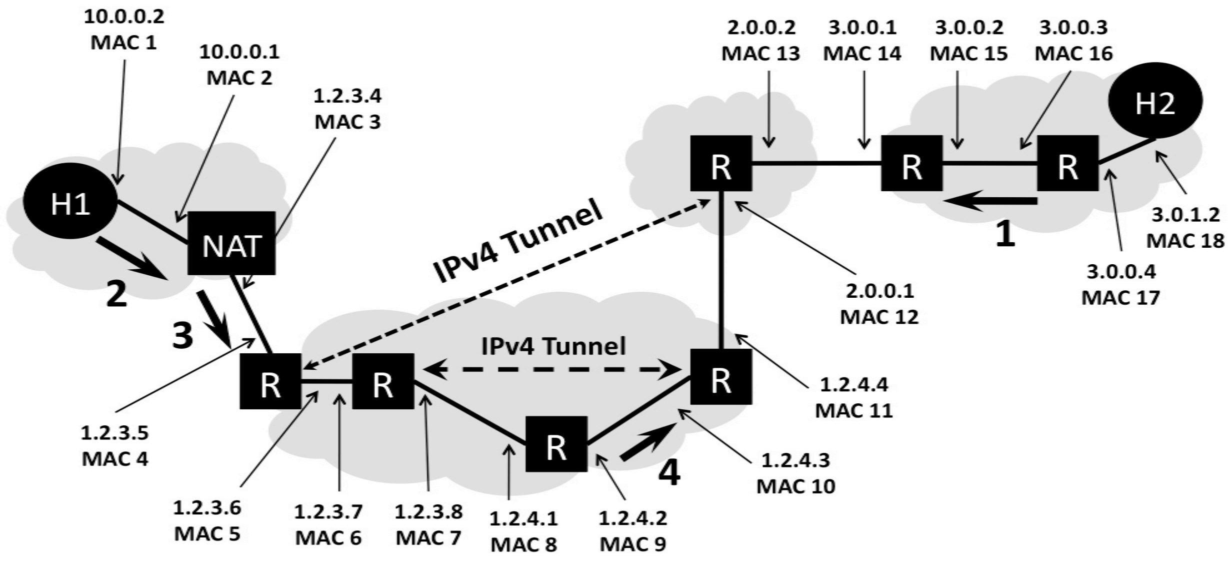
This test has five (5) questions. Put your name on ***every page***, and write out and sign the Honor Code pledge (in cursive) before turning in the test.

The exam will be scored out of 62 points and will last for 105 minutes, so pace yourself. Show your work for problems. Partial credit will often be given.

"I pledge my honor that I have not violated the Honor Code during this examination."

Question	Score
1	/ 6.5
2	/ 10
3	/ 7
4	/ 15
5	/ 23.5
Total	/ 62

QUESTION 1: Routing, addressing, and tunneling (6.5 points)



The above figure shows a network topology. The LAN on the left uses a NAT to connect to the Internet and includes a client host H1. The LAN on the right includes a webserver H2. Packets between the two endpoints are routed along the path shown by a heavy dark lines, which includes two IPv4 tunnels. All packets which traverse the path use both tunnels. The various network interfaces have IP and MAC addresses as shown.

H1 has established an HTTP session with web server H2 and data packets are flowing between the two machines. As an example, we have filled in the headers for packet 1 (traveling from the server H2 to the client H1). Note that you should order your headers from "outermost" in, as shown: Ethernet should be listed before IP, because the Ethernet packet exists first on the wire.

Header Type	***Source***	***Destination***
Ethernet	MAC 16	MAC 15
IP	3.0.1.2	1.2.3.4

You only have to fill in the header type and the source and destination address for the network and datalink layer headers for packets 2, 3, and 4 (these packets are all traveling from the client H1 to the server H2, as marked on the figure with heavy black arrows and numbers). Note: You might not need to use all the rows supplied.

(a) [1.5 points – 0.5 for MAC, 1 for IP. Split credit equally over fields. i.e., 1/3 point for each IP field; 1/6 point for each MAC field.] Header for packet 2:

Header Type	***Source***	***Destination***
Ethernet	MAC 1	MAC 2
IP	10.0.0.2	3.0.1.2

(b) [1.5 points – 0.5 for MAC, 1 for IP] Header for packet 3:

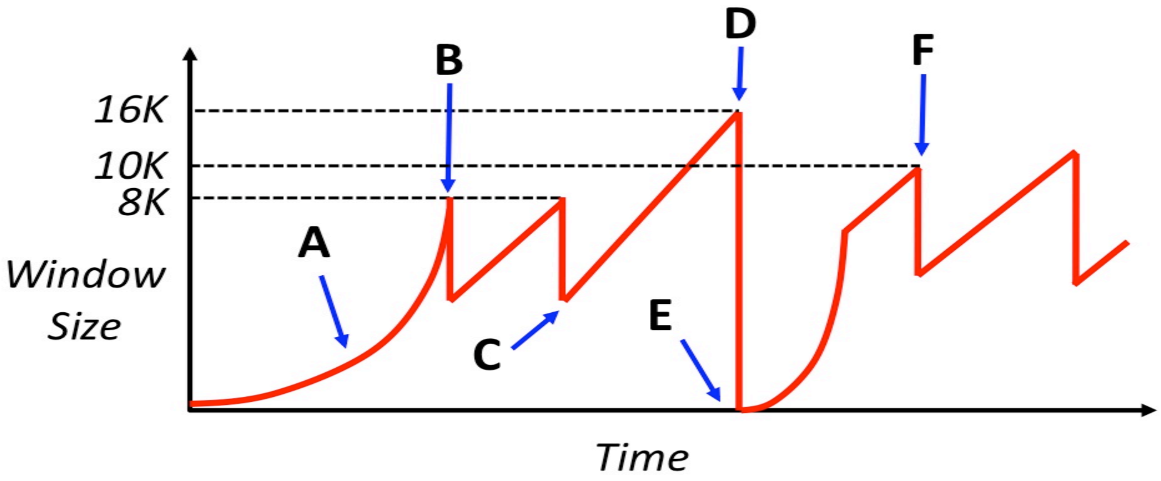
Header Type	***Source***	***Destination***
Ethernet	MAC 3	MAC 4
IP	1.2.3.4	3.0.1.2

(c) [3.5 points – 0.5 for MAC, 1 for each of 3 IP headers] Header for packet 4:

Header Type	***Source***	***Destination***
Ethernet	MAC 9	MAC 10
IP	1.2.3.8	1.2.4.3
IP	1.2.3.6	2.0.0.1
IP	1.2.3.4	3.0.1.2

QUESTION 2 : TCP and Congestion Control (10 POINTS)

Consider the following graph of TCP throughput (**NOT DRAWN TO SCALE**), where the y-axis describes the TCP window size of the sender. We will later ask you to describe what happens on the right side of the graph as the sender continues to transmit.



1. [5 points – 1 for each] The window size of the TCP sender decreases at several points in the graph, including those marked by B and D.

(a) Name the event at B which occurs that causes the sender to decrease its window.

Triple duplicate ACK

(b) Does the event at B necessitate that the network discarded a packet (Yes or No)? Why or why not?

No. It could be due to reordering due to queuing or asymmetric paths.

(c) Name the event at D which occurs that causes the sender to decrease its window.

Timeout

(d) Does the event at D necessitate that the network discarded a packet (Yes or No)? Why or why not?

No. Congestion in either direction could cause RTT > RTO (retrans. timeout).

(e) For a lightly-loaded network, is the event at D MORE likely or LESS likely to occur when the sender has multiple TCP segments outstanding? (Write "MORE" or "LESS")

LESS

2. [1 points] Consider the curved slope labeled by point A. Why does the TCP window behave in such a manner, rather than have a linear slope? (Put another way, why would it be bad if region A had a linear slope?)

This "slow-start" period quickly discovers the maximum acceptable throughput that the path supports – otherwise, AI (additive increase) could take too long (each a full RTT).

3. [3 points – 1 for each] Assume that the network has an MSS of 1000 bytes and the round-trip-time between sender and receiver of 100 milliseconds. Assume at time 0 the sender attempts to open the connection. Also assume that the sender can "write" a full window's worth of data instantaneously, so the only latency you need to worry about is the actual propagation delay of the network.

(a) How much time has progressed by point B?

It depends if you interpret B as before or after you received the triple duplicate ACK, so we will accept both answers:

1 RTT (TCP handshake) + 3-4 RTT in slow-start (1, 2, 4, (8) MSS)
= 4 or 5 RTT = 400 or 500 ms

(b) How much time has progressed between points C and D?

4 MSS to 16 MSS = 12 periods of RTT = 1.2 s

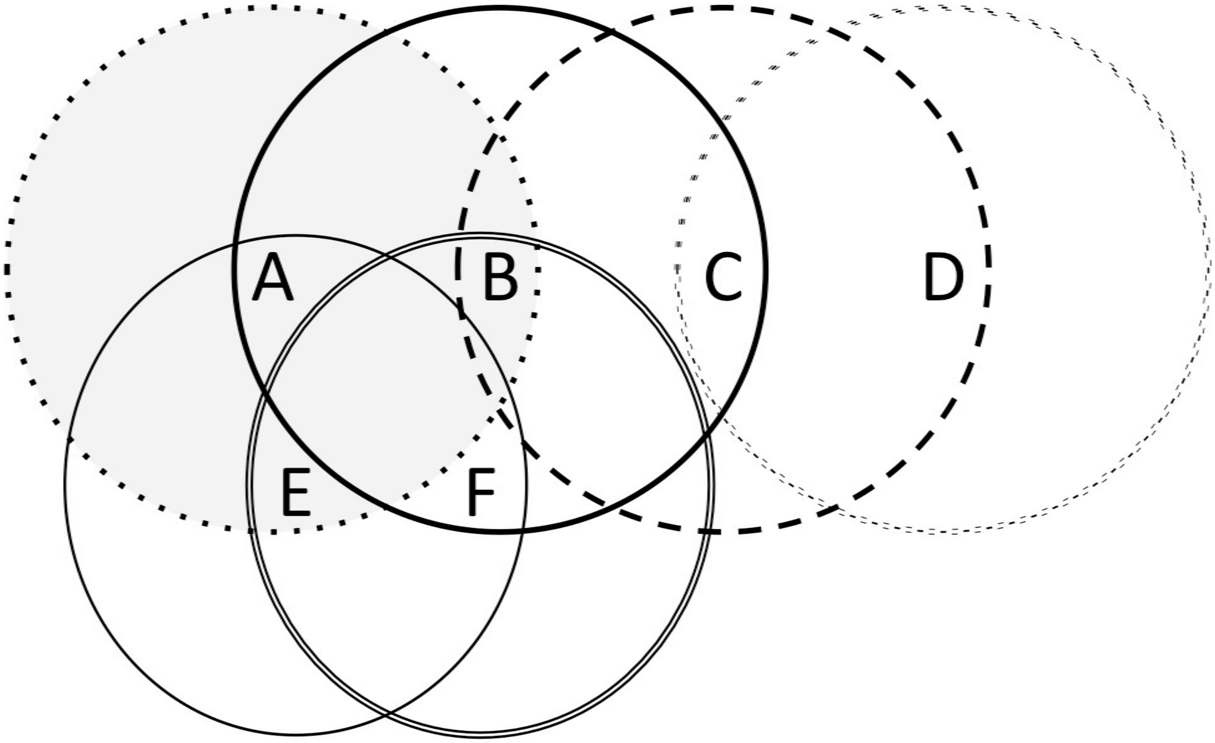
(c) How much time has progressed between points E and F?

First: slow start to 8K window size (1->2, 2->4, 4->8 MSS), then AI from 8 to 10 MSS window size (8->9->(10) MSS). Again, similar question about "where" F exists as question (a), so we will accept either 5 or 6 RTT = 500 or 600 ms.

4. [1 point] If the sender shares its network with other clients whose traffic traverses the same IP routers, give one explanation for why point D is higher than point B?

Changing cross-traffic by other concurrent senders across same routers.

QUESTION 3: Wireless (7 points)



Consider the wireless topology above, comprised of 6 nodes. Circles around each node illustrate their transmission range, e.g. A's range is shown by the dotted, shaded circle. Assume that if the transmissions of two nodes' will interfere at a location if and only if they transmit at the same time and their transmission areas overlap. In these problems, assume that losses only occur due to collisions.

1. [2 points – 1 for each] When node A transmits to node B, list the potential hidden terminals from A (in either direction, i.e., those who might clobber A's transmission or those who A's transmission might clobber) and exposed terminals.

(a) Hidden terminals: **C, F**

(b) Exposed terminals: **E**

2. [2 points – 1 for each] What about when node B transmits to node C?

(a) Hidden terminals: **D**

(b) Exposed terminals: **A, F**

3. [3 points – 1 for each] You considering using a "Request to Send (RTS) / Clear to Send (CTS)" protocol to reduce these potential problems from hidden and exposed terminals.

(a) When using RTS/CTS, explain what would prevent a hidden terminal from clobbering a sender?

Hidden terminal would see the CTS of the sender's desired destination, but not the RTS of the sender, and choose not to send to the same destination as had sent the CTS.

(b) When using RTS/CTS, explain how an exposed terminal decides it is safe to send to another destination?

Exposed terminal would see the RTS of the sender but not the CTS (from the sender's desired destination), and know it's safe to send.

(c) Is RTS/CTS more like statistical (time-division) multiplexing, normal time-division multiplexing, or frequency-division multiplexing? No explanation needed.

Statistical multiplexing

QUESTION 4: Short Answers (15 points)

For the following questions, be specific, but you don't need to be prolific. Answers like "for efficiency" are too vague to receive credit.

1. [3 points] Ethernet and IP

(a) [2 points] Suppose Ethernet was the only existing LAN technology, so every host in the Internet was part of a local Ethernet and thus had a globally-unique Ethernet address. Your friend suggests that IP isn't necessarily anymore, and the entire Internet could just be one large, switched Ethernet instead. Give two reasons why using existing Ethernet protocols for this is a bad idea from a networking perspective (i.e., don't consider security or privacy).

1. **Flooding doesn't scale for broadcast transmission (when address hasn't been learned yet by switch)**
2. **Even if switched Ethernet, flooding doesn't scale for ARP or spanning tree.**
3. **MAC addresses aren't hierarchical – switches couldn't aggregate addresses for more compact routing tables, unless in IP prefixes.**

(b) [1 point] What about the other way around, why do we not simply assign IP addresses to network adaptors, instead of dealing with both MAC (Ethernet) and IP addresses? Give one reason.

1. **Approach would give up ability to have topological addressing and hence not support aggregation, again leading to large routing table sizes**
2. **End-hosts would have to be part of routing protocol and announce addresses when they migrate, or the equivalent of global "ARP" for an IP address, neither which scales**

2. [4 points] You've been hired by a local company to set up a router that both serves as the network's bidirectional firewall and also NATs hosts in the corporate network.

(a) [1 point] The corporate network consists of about 100,000 machines that simultaneously access the Internet using port-based NAT. What problem could arise if the corporate network only has a single public IP address?

There are more machines that want to externally communicate than ports (~65K). Either NAT request fails or multiple machines effectively thrash

over IP assignments, breaking connections as ongoing ports are "reassigned" to new nodes.

(b) [1 point] The company wants to run a Web server on a machine behind the NAT. They desire to make this server, and only this server, accessible from the public Internet. (In other words, other internal hosts should be able to connect to external addresses, but not vice versa.) What do you need to do to set this up?

Manually configure the NAT to map port 80 on its public IP address to the internal (private) IP address:80 of the webserver. This is referred to as "port forwarding".

(c) [2 points - .25 for IP and port, .5 for others] Now assume that the corporate network is not NATted, and the corporation owns the IP prefix 1.0.0.0/8. Every corporate host has a publicly-routable IP address, and the Web server running on port 80 is located at 1.2.3.4. What firewall rules are needed to ensure that the server, and only the server, can accept incoming connections? For now, consider only TCP traffic.

Fill in the five missing entries of the following table. A "*" represents the wildcard, meaning "match anything for this field". Remember that a firewall first tries to match the first rule in its list, before moving on to the second, third, etc.

Source IP	**Source Port**	**Dest IP**	**Dest Port**	**TCP Flags**	**Allow/Deny**
*	*	**1.2.3.4**	**80**	**SYN**	ALLOW
*	*	*	*	SYN-ACK	ALLOW
1.0.0.0/8	*	*	*	SYN	**ALLOW**
*	*	*	*	SYN	**DENY**
*	*	*	*	*	ALLOW

3. [4 points – 1 for each] Using TCP, a sender has sent out a total of ten data segments, each of 1000 bytes. Assume that the sender's initial sequence number (ISN) is 5000. If a TCP segment were to include 1000 bytes with sequence numbers 20,001 through 21,000, you can refer to the segment simply as "segment 21,000".

(a) After the sender receives a packet with "ACK 8001", how many TCP segments, if any, does the sender know that the receiver definitely received?

3 packets (6000, 7000, 8000)

(b) The sender subsequently receives two more packets with “ACK 8001”. Given this occurrence, which of the TCP segments, if any, does the sender conclude might be lost and thus should be retransmitted?

It may conclude packet 8001-9000 is possibly lost and should be retransmitted

After transmitting this segment, the sender receives an “ACK 12001” in response.

(c) Based on this information, which of the original 10 segments can the sender conclude might be lost, if any?

None.

(d) Which of the original 10 can the sender assume were definitely received?

7 packets (bytes 5001-12000)

4. [4 points] We now test some of your knowledge of UNIX socket programming. Recall the web proxy you implemented for assignment 1.

(a) Your proxy is downloading a web object from an origin webserver through an open TCP connection. The download has been progressing nicely, when your latest *read* call on the connection’s socket returns 0.

(a.1) [0.5 points] What does this signify? (in general about a UNIX file descriptor)

End of file (EOF) was reached, nothing more to read.

(a.2) [1 point] What network event occurred to cause this? (be specific)

The remote peer closed() their socket, the remote network stack sent a FIN.

(b) Your proxy has been sending the web object back to the client browser through a TCP connection (named by the file descriptor *fd*). The 5000 byte object is currently stored in a single contiguous memory buffer *buf* (also of length 5000).
When calling *write* in your code, you execute the following code:

write (fd, buf +2000, 1000)

Remember that the *write* system call takes a file descriptor, a pointer to a memory region, and the length (in bytes) of data from that region to write to the file descriptor.

(b.1) [0.5 points] Which byte range are you trying to transfer?

Bytes 2000-2999 (inclusive) in buf.

(b.2) [1 point] If this *write* call returned 500, what does this signify?

Only bytes 2000-2499 were written

(b.3) [1 point] Why might this occur?

The socket buffer only had space for 500 bytes, not the full 1000. The application should call write again with *write (fd, buf+2500, 500);*

QUESTION 5: Multiple choice (23.5 points)

Each multiple choice question is worth 2.5 points. Circle ALL that are true or apply.

1. When a TCP packet arrives at a host, in order to direct the segment to the appropriate socket, the operating system's network stack uses the following fields:

   a. transport protocol number
   b. destination IP address
   c. source port number
   d. destination port number
   e. destination MAC address

2. Which of the following is/are true about wireless networks?

   a. All wireless networks must use access points.
   b. The sender can always detect a collision without feedback from receiver.
   c. Collisions are minimized when RTS/CTS mechanisms are used.
   d. TCP congestion control mechanisms work poorly in wireless environments if they do not perform any type of link-layer retransmission.
   e. Wireless networks generally have higher loss rates than that in wired networks.

3. Which of the following is/are true about routers?

   a. Routers reassemble IP fragments if the next link can handle the full datagram.
   b. Routers can arbitrarily drop packets if they want.
   c. Routers can not change the IP packets they forward at all
   d. On a router with many 1 Gbps ports, the router backplane can only handle 1 Gbps on the shared bus, leading to potential congestion.
   e. In their line cards, routers lookup forwarding tables in the incoming direction and queue packets in the outgoing direction.

4. Otto Pilot built a home-brew network with 20 computers. The RTT between each computer is 10 ms. Communication between computers uses a simple UDP query and response protocol. If no response is received within 20 ms, a computer retransmits the request. Soon, Otto notices congestion collapse in his network. Which of the following techniques is/are guaranteed to prevent congestion collapse?

a. Double the timeout value from 20 ms to 40 ms.
b. Increase the size of the queue in each router from 4 packets to 8 packets.
c. Use exponential backoff in the timeout mechanism while retrying queries.
d. If a query is not answered within a timeout interval, multiplicatively reduce the maximum rate at which the client application sends query packets.
e. Use a flow control window at each receiver to prevent buffer overruns.

5. Which of the following statements is/are true about physical and link-layer protocols?

a. Manchester encoding requires that sender and receiver have clocks that run at approximately the same rate, so they can differentiate between the encoding of a single "0" and than of multiple "0"s.

b. Both Ethernet and 802.11 ("Wifi") combine Carrier Sense with Collision Detection to ensure fair media access for multiple parties on a LAN.

c. When many hosts seek to actively communicate, token-ring schemes can achieve higher total goodput on a shared LAN than Ethernet.

d. An Ethernet adapter passes every non-corrupt frame that it receives up to the network layer.

e. Ethernet switches (as compared to hubs) eliminate the need for broadcasting packets to all hosts.

6. Which of the following statements is/are true about drop-tail and RED (random early detection) mechanisms in queues? Assume that these are always coupled with a FIFO scheduling policy. The **full queue** problem occurs if routers' queues are often full and the **lockout** problem refers to a situation where a small number of flows monopolize the available queue space on a router.

a. Drop-tail solves the full queue problem.
b. RED solves the full queue problem.
c. RED solves the lockout problem for TCP flows.
d. Drop-tail has fewer burst losses than RED.
e. Both drop-tail and RED can be used with ECN (explicit congestion notification), so the router can signal congestion to the sender without dropping a packet.

7. Which of the following statements is/are true about fair queuing algorithms?

a. For a router serving *n* flows, fair queuing ensures that no flow can transmit at more than $1/n^{\text{th}}$ the link's capacity.

b. Fair queuing is often used in the core of the Internet to minimize denial-of-service attacks by individual senders.

c. Fair queuing algorithms conceptually track the number of bytes each flow consumes, rather than the number of packets

d. Fair queuing algorithms can be based both on unique flows (5-tuples) or on unique classes of traffic (specified in the IP TOS field).

e. Fair queuing with drop-tail policies experiences synchronized losses between multiple senders.

8. Which of the following applications use tunneling?

a. IPv6 communication across IPv4 networks
b. Virtual Private Networks
c. Network Address Translation
d. Mobile IP
e. HTTP Proxying

9. A network C advertises the CIDR network number 192.3.48/20 (and no other numbers). What network numbers (all 24 bits) could AS C own? **(4-bit prefix of second-to-last octet is 0011 = 48)**

a. 192.3.128 (10000000)
b. 192.3.49 (00110001)
c. 192.3.64 (01000000)
d. 192.3.1 (00000001)
e. 192.3.62 (00111110)

10. [1 point] Describe one thing you would like to see changed in the class. (Only wrong answer is "nothing")