NAME:

Login name:

Computer Science 461
Midterm Exam
March 14, 2014

This test has five (5) questions. Put your login 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 50 points and will last for 50 minutes, so pace yourself. For the few questions that are not multiple choice, show your work.

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

Question	Score
1	/ 22.5
2	/ 10.5
3	/ 8
4	/ 9.5
Total	/ 50.5

QUESTION 1: Multiple choice (22.5 points)

For ALL multiple choice questions, CIRCLE ALL that apply / are true. A correct answer for each multiple choice option is worth 0.5 points (i.e., a question with options (a) through (e) is worth a total of 2.5 points).

1. Which are true about network switches and routers?

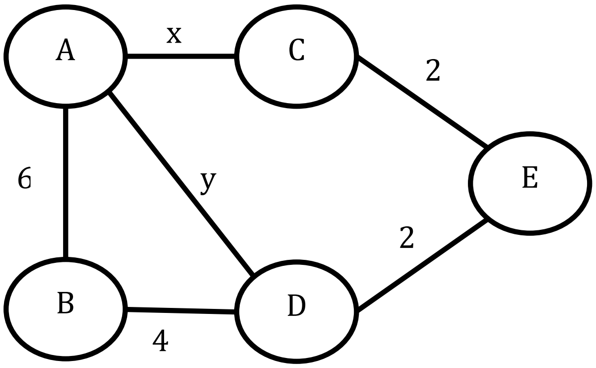
   a. Ethernet switches will always send received frames out all interfaces (except the one on which the packet was received).
   b. Both IP routers and Ethernet hubs drop packets when there is congestion.
   c. Ethernet switches learn the location of hosts on their network by observing the frames they process.
   d. If an IP router doesn't know the location of a destination IP address in a packet it receives, it will flood the packet out all interfaces (except the one on which the packet was received).
   e. In their forwarding tables, IP routers store the shortest-path distance to each IP prefix.

2. Which are true about Ethernet protocols?

   a. Ethernet is commonly used for the link-layer protocol for long-distance links (such as across a country).
   b. The Maximum Transmission Unit (MTU) of Ethernet is dictated by the buffer size of the link-layer endpoints.
   c. Ethernet adaptors send frames as soon as they have any available data.
   d. The Ethernet spanning tree may take a longer path through a network than that which would be calculated by a link-state algorithm (assuming both have converged).
   e. The Ethernet spanning tree protocol prevents Ethernet forwarding from ever encountering a loop.

3. Which of the following is true of distance-vector routing (DV) in a network whose longest path is of length N hops ?

    a. Bad news (such as a link going down) spreads fast such that within N exchanges, every router is guaranteed to be aware of the link failure.
    b. Good news (such as a link coming up) may result in count-to-infinity and therefore travels slowly.
    c. DV would still work if the metric was propagation delay.



4. For the above network, which constraints on *x* and *y* guarantee traffic from B to C will *always* flow through node A?

    a. x > 4
    b. y + x < 6
    c. y + x < 4
    d. x < 4

5. A network advertises the CIDR network number 50.1.56.0/22 (and no other numbers). Which IP addresses could the network own?

    a. 50.1.57.0
    b. 50.1.59.1
    c. 50.1.60.0
    d. 50.1.120.0

6. Which of the following are true about mobile devices and mobile IP:

a. Upon joining a new Wifi base station, one of the first things your computer will do is send an IP packet to the local DHCP server.
b. In Mobile IP, a mobile host is no long reachable at its “home address” as soon as it migrates to a foreign network.
c. A home agent impersonates the network address of its mobile client while the client roams outside its home network.
d. A mobile client and a foreign agent can be on the same box.

7. Which of the following are true about queue management and scheduling policies in IP routers?

a. Drop-tail queues encounter fewer backoff synchronization problems than with Random Early Detection (RED).
b. Routers implementing RED lead to greater fairness between TCP flows than those implementing drop-tail policies.
c. With ECN, if a packet is marked as ECN-capable, routers send congestion notification messages back to the sender to cause them to backoff.
d. Routers start using ECN when their queues are full.
e. Strict priority queuing ensures that all TCP flows achieve some minimal share of the link’s bandwidth capacity.

8. Which of the following are true about HTTP headers and connection management?

a. All HTTP responses must either close a connection or include a Content-Length: header to signify to end of a HTTP response body.
b. Persistent HTTP connections can have lower latency than non-persistent connections because they can avoid performing a new TCP handshake for each HTTP request.
c. When transferring many small Web objects, the only performance difference between persistent and non-persistent HTTP connections is related to connection establishment.
d. HTTP headers have a fixed size.
e. The `Host:` field in HTTP allows the same web server to server content for multiple domains.

9. Which of the following about UNIX socket programming are true.

   a. **`accept()` returns when the receiver completes the three-way TCP handshake.**
   b. When a TCP sender calls `close()`, the network stack immediately sends a FIN packet to its peer.
   c. **The return value of `recv()` specifies the number of bytes read from a socket, or if the socket was closed or an error was encountered.**
   d. `send()` on a TCP socket causes a sender to generate a TCP/IP packet and send it out the appropriate network interface.

10. You've learned about multiple communication protocols across the different network "layers". List one or more data-plane protocols for each layer:

   a. Layer 7 (Application) -- Give 2 examples: **HTTP, SMTP, NFS**
   b. Layer 4 (Transport) -- Give 2 examples: **TDP, UDP**
   c. Layer 3 (Network) -- Give 1 example: **IPv4**
   d. Layer 2 (Link) -- Give 2 examples: **Ethernet, WiFi (802.11), SONET**

QUESTION 2 : TCP Congestion Control and and Flow Control (10.5 pts)

Consider the following behavior of a TCP connection (using the congestion control algorithm we learned in class).

A time 0, a TCP sender initiates a connection. As soon as the connection is established, the TCP sender will begin sending data. The MSS is 1KB and RTT is 100 ms.

1.A) Assuming the connection does not lose any data or experience any timeouts, at what time will the sender's congestion window be 16KB?

1 RTT for setup, then transitions 1 -> 2 -> 4 -> 8 -> 16 (4RTT)
= 5 RTT = 500ms

Right after the sender's congestion window has reached a size of 16KB, a timeout occurs. After the timeout is detected, the sender continues sending more data over the established connection.

1.B) Assuming no additional packet loss or timeouts, how long (since the observed timeout) will it take for the congestion window to build to size 14KB?

After timeout, drops to 1 MSS, then does fast retransmit to ½ previous cwnd
1-> 2 -> 4 -> 8

Then additive increase
8 -> 9 -> 10 -> 11 -> 12 -> 13 -> 14

= 9 RTT = 900ms

1.C) While its congestion window is at 14KB, the sender receives three acknowledgements for the same sequence number. How long after receiving the third acknowledgement will it take for the sender's congestion window to be at least 9KB again?

Just drops by ½ cwnd to 7MSS
Then does additive increase:
7 -> 8 -> 9
= 2 RTT = 200 ms

2. Consider a scenario with two hosts, Alice and Bob. A web server running on Alice is trying to send data to a browser on Bob. For each TCP connection, Alice's TCP stack maintains a buffer of 512 bytes and Bob's TCP stack maintains a buffer of 1024 bytes. For simplicity, assume TCP sequence numbers began at 0 in this problem.

2.A) Bob's stack received up to byte 560 in order from Alice, although its browser has only read up to the first 60 bytes. What will be the ACK and window in the TCP headers that Bob next sends to Alice?

SEQ Number = 561 ; Window = 1024 – 500 = 524

2.B) Later in the same connection, Alice's congestion window is set to 1 MSS = 536 bytes and the advertised flow-control window from Bob is 560 bytes. The last ACK that Alice received from Bob is byte 700, and the last byte that Alice sends to Bob is byte 900.

2.B.i) What is the smallest byte number that Bob will not accept?

Can accept 560 – 200 = 360 bytes more. So can get up to 900 + 360 = 1260
So first byte he will not accept is 1261

2.B.ii) Assuming that Alice doesn't receive any more ACKs and her window does not change, what is the greatest byte number that Alice can send?

Can send min (congestion window, flow window)
Her congestion window is 536. Already has 200 outstanding.
Can send 536-200=336 more.

So greatest byte = min (1236, 1260) = 1236

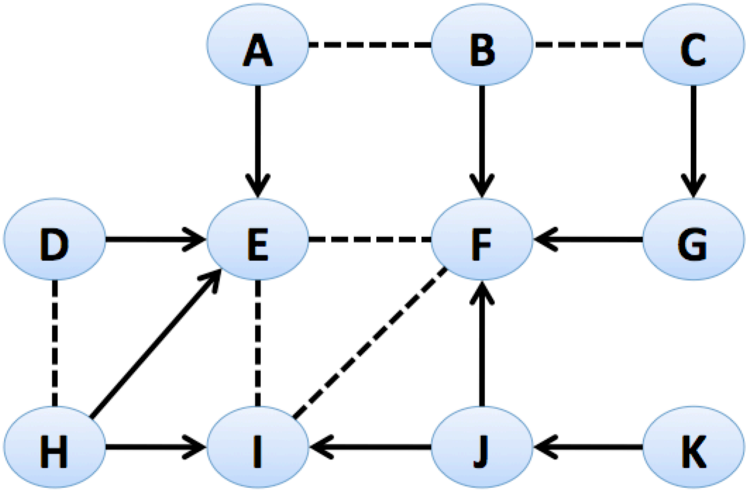
2.B.iii) Again assuming that Alice doesn't receive any additional ACKs, how many more bytes can the web server running on Alice write to its network socket before blocking?

Can fill up network buffer.
Local buffer is 512 bytes.
Buffer currently has 200 bytes (900 not ACKed – 700 ACKed), as Alice can't delete bytes between 700 and 900 because it hasn't been ACKed (and TCP needs to keep around to possibly retransmit under a loss)

So app can write 512 – 200 = 312 bytes more

QUESTION 3: Interdomain relationships (8 points)

Consider the following network graph, which represents the peering **(dotted lines)** and transit relationships **(solid directed arrows)** between Internet Autonomous Systems (ASes), represented by the vertices and labeled A through K. That is, A and B are peers, while E is A's provider and A is E's customer. (The arrows point from the payer (customer) to the payee (provider).)



This is a question exploring the relationships that ASes might establish for business reasons (e.g., either profit extraction or settlement-free peering).

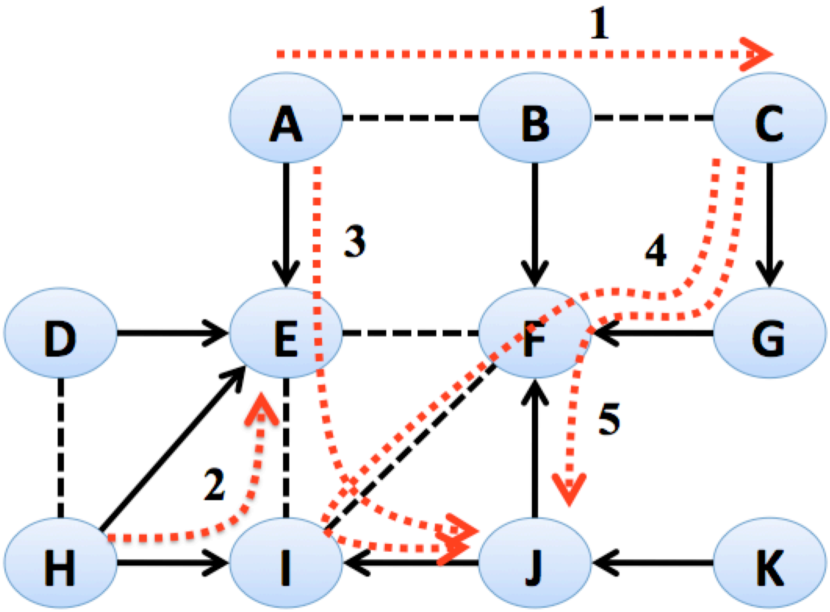
A) We are first interested in deciding which are the tier-1 ASes. Recall that Tier-1 ASes (i) have no providers and (ii) peer with all other Tier-1 ASes. Assuming that F is a Tier-1 AS, which other ASes are Tier-1 ASes?

E and I

B) Assume the tier-1 providers from part A) are maintained. Would each AS enter into the following proposed peering or customer-provider relationships?

- B becomes a customer of C? **Yes** No
- D becomes a customer of H? **Yes** No
- F and G become peers? Yes **No**
- J and K become peers? Yes **No**
- I becomes a customer of H? Yes **No**
-

C) In this next figure , we labeled some potential paths (in long dotted arrows) that are announced via BGP. Consider whether these paths respect the Gao-Rexford stability conditions (or more informally, are likely policy-compliant given the economic goals of the actors). So for the path labeled 1, A's addresses would be announced to B, which in turn re-announces those addresses to C.



Which of the labeled paths are allowed under normal BGP routing policy assumptions?

1. A→B→C: Valid **Invalid**

2. H→I→E: **Valid** Invalid

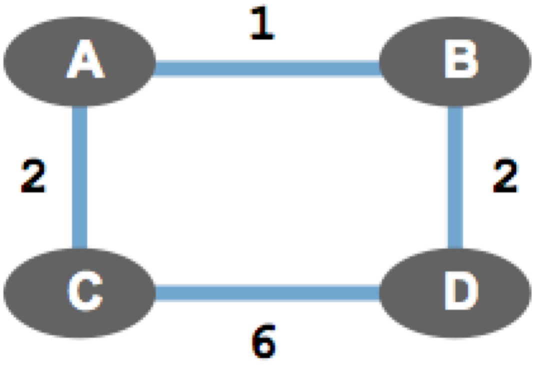
3. A→E→I→J: **Valid** Invalid

4. C→G→F→I→J: **Valid** Invalid

5. C→G→F→J: **Valid** Invalid

QUESTION 4: Routing Convergence (9.5 points)

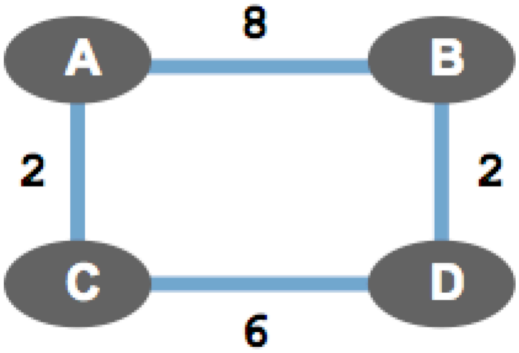
1. The following question considers a network of 4 routers (A, B, C, D) that run a distance vector protocol between them. Link costs between each router are shown on the figure.



1.A. **Given the figure above, calculate the shortest paths between each router to router D.** In the table below, write down the next hop on the shortest path from each router to D, as well as that shortest path's cost.

Router	Router's next hop on Shortest Path to D	Shortest Path Cost from Router to D
A	**B**	**3**
B	**D**	**2**
C	**A**	**5**
D	D	0

1.B. Now, assume the cost of link A↔B increases from 1 to 8. Write down the series of cost changes that A and C compute **(in calculating their shortest path to D)** as the protocol reconverges. Assume that in a given round, a node will consider any updates it received from the prior round, recompute any path cost changes, and (if needed) deliver updates to its neighbors.



Immediately after the link cost change, A updates its distance matrix and routing table. Below are calculations that nodes A and C will make during each round of the distance vector algorithm based on either the link cost change or distance vector updates from its neighbors. For this problem, you may ignore updates from B and D and only focus on the distance vector updates exchanged between A and C.

Fill out the series of calculations for A and C that can occur in computing their distance matrices and routing tables. Assume that the nodes are not "poisoning" the reverse routes. You may NOT need to use all the tables given. After the nodes exchange the final set of distance vector updates, write: "DV updates & No Cost Changes == Convergence" after the last box that you fill in for which there are path cost changes.

A’s path costs to D	Path cost
Via B	10
Via C	**7**

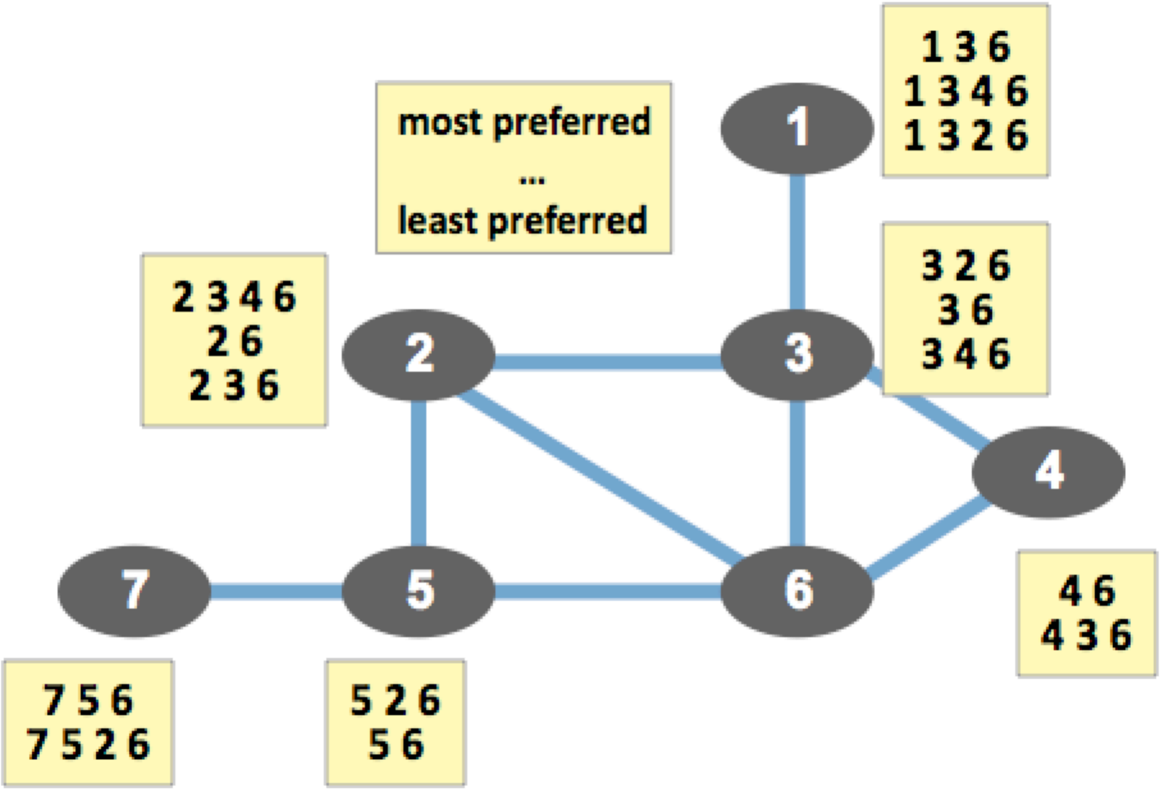
C’s path costs to D	Path cost
Via A	**9**
Via D	**6**

A’s path costs to D	Path cost
Via B	**10**
Via C	**8**

C’s path costs to D	Path cost
Via A	**10**
Via D	**6**

A’s path costs to D	Path cost
Via B	
Via C	

2. The below figure shows a network topology, with each vertex representing an Autonomous System (AS). ASes are announcing and withdrawing paths via BGP, and their path preferences to AS 6 are shown in the dashed box next to each vertex. Preferred policies are listed at the top of each list, less preferred policies are at the bottom. Assume that ASes announce their currently preferred routes to their neighbors (i.e., don't consider any notion of payment in this graph).



Does this AS topology have a stable paths solution? If yes, give the preferred routes at each node that form the solution (you can draw on the figure). If not, explain why.

Yes, 46, 26, 326, 526, 7526, 1326