Name_____ Student ID_____ Department/Year_____

Final Examination

Introduction to Computer Networks
Class#: 901 E31110
Fall 2008

9:30-11:10 Tuesday
January 13, 2009

Prohibited

1. You are not allowed to write down the answers using pencils. Use only black- or blue-inked pens.
2. You are not allowed to read books or any references not on the question sheets.
3. You are not allowed to use calculators or electronic devices in any form.
4. You are not allowed to use extra sheets of papers.
5. You are not allowed to have any oral, visual, gesture exchange about the exam questions or answers during the exam.

Cautions

1. Check if you get 18 pages (including this title page), 6 questions.
2. Write your **name in Chinese**, student ID, and department/year down on top of the first page.
3. There are in total 150 points to earn. You have 100 minutes to answer the questions. Skim through all questions and start from the questions you are more confident with.
4. Use only English to answer the questions. Misspelling and grammar errors will be tolerated, but you want to make sure with those errors your answers will still make sense.
5. If you have any extra-exam emergency or problem regarding the exam questions, raise your hand quietly. The exam administrator will approach you and deal with the problem.

1. (Transport) TCP is a transport layer service that provides a number of major functions: (1) reliable data delivery, (2) connection management, (3) flow control, and (4) congestion control. Based on your understanding of TCP, address the following questions:
   (1) In classic reliable data delivery mechanisms, packet losses are detected by timer timeouts. Setting the timeout interval is a challenging task. If it is set too long, the source will need to wait unnecessarily before it can retransmit. If it is set too short, there could be false-retransmissions due to unexpected delay in the network. Can you describe how TCP determines its timeout interval? (5%)
   (2) Packet losses may be detected by a timeout or by receiving three duplicate acknowledgements. Describe how TCP's congestion control mechanism reduces the window size when the loss is detected by timeout vs. by duplicate acknowledgements. Which one is more aggressive in reducing the congestion window size? Why do you think TCP's congestion window adaptation mechanism is designed with such different reduction strategies for different kinds of packet losses? (5%)
   (3) TCP's congestion control mechanism works in two states. When the cwnd is smaller than or equal to the slow start threshold (ssthresh), the TCP source is in the slow start state. When the cwnd is larger than the ssthresh, the TCP enters the congestion avoidance state. The congestion window size (cwnd) increases in a different way in a different state. Describe how cwnd is incremented upon receiving a new acknowledgement in each state. Which one is more aggressive increasing the window size? Why do you think TCP's congestion window adaptation mechanism is designed with such different increment strategies in different states? (5%)

Sample Solution:

(1) Take RTTSample per data-ack pair. The **EstimatedRTT** is derived by taking a smoothed weighted average on the history of RTTSamples.

Take deviation of RTTSample to EstimatedRTT per RTTSample. Calculate **DevRTT** as a smoothed weighted average of all the derivations.

Make the **TimeoutInternval** EstimatedRTT+4DevRTT. 4DevRTT is to accommodate short-term variations on EstimatedRTT.

(2) Timeout: drop to 1
Dupack: drop to half

Timeout is more aggressive

When the source can receive duplicate acks, that means **the packets subsequent of the packet gets dropped** can still get thru to the destination.
On the other hand, when the source detects a timeout, that indicates that the **packets (if there are any) subsequent of the dropped one are also dropped**.
In the first case, the congestion is not so bad as the 2nd one. Thus, a less aggressive window reduction strategy when experiencing dup acks and a more aggressive window reduction strategy when detecting timeouts.

(3) Slow start: cwnd = cwnd+1 (per ack)
Congestion avoidance: cwnd = cwnd+1/cwnd (per ack)

Slow start is more aggressive.

In slow start, the congestion window starts from 1. To **shorten the time TCP identifies the ideal congestion window size** and since the **congestion window size is small anyway**, TCP can afford to be aggressive.
After entering the congestion avoidance stage, the **window size is getting large**. To avoid **a sudden surge of packet drops**, TCP increases the congestion window size more conservatively.

2. (Transport) TCP's congestion window adaptation mechanism is inspired by the binary search principle. In that, the correct solution falls in a given the range of possible solutions. One attempts with the solution in the middle of the range. Pick the smaller one between the middle numbers if there is an even number of numbers in the range. Given the feedback that the correct solution is larger or smaller than the middle number, one may re-adjust the range of the solution range and try the middle number in the new range until the guess is correct.
   (1) Suppose the range is 1-32. Using binary search to identify the solution, the first guess will be 16. If the first guess is too high, what will be your second guess? If your second guess is too low, what will be your third guess? If your third guess is too high, what will be your fourth guess? If your fourth guess is too low, what will be your fifth guess? Can you confirm the correct solution, in the worst case, by the 5^{th} guess? (5%)
   (2) If the range is 1-64, how many guesses does it take at maximum to confirm the correct solution? If the range is 1-128, how many guesses does it take at maximum to confirm the correct solution? If the range is $1\text{-}2^k$, how many guesses does it take at maximum to confirm the correct solution? (5%)
   (3) The binary search algorithm works efficiently at identifying the correct solution given the range of solutions. Suppose now the correct solution may change, which means the range determined from prior guesses might not be valid for the subsequent guesses. To begin with, 1 is the smallest number possible and M is the highest number possible. Follow the principle of **'guess the middle number in the range'**. What will be your first guess? If the first guess is too high, what will be your second guess and why? If your second guess is too low, what will be your third guess and why? If the first guess is too low, what will be your second guess and why? If your second guess is too high, what will be your third guess and why? (5%)
   (4) Suppose you are designing a new version of TCP based on the 'memory-less' binary search algorithm developed in (3). The minimum window size is 1, maximum window size is M. Denote the current window size is C. When your TCP detects a packet loss which indicates the congestion window is too high, what will be the new congestion window size? If your TCP receives a new acknowledgement packet (no loss detected and the congestion window is too low), what will be the new congestion window size? (5%)
   (5) Compare and contrast how this new TCP and the original TCP in the congestion avoidance state in terms of how they decrease and increase the window size. Which one is more aggressive in increasing/decreasing the window size? Why isn't TCP

designed based on the exact memory-less binary search? (10%)

Sample Solution:

(1) 8, 12, 10, 11. Yes.

(2) 6, 7, k

(3) M/2(between 1~M), M/4 (between 1~M/2), 5/8M(between M~M/4)
M/2(between 1~M), 3M/4(between M/2~M), 3M/8(between 1~3M/4)
(If M is odd, the formula is slightly different. You get the credits as long as you get the idea.)

(4) C/2, (C+M)/2
(If M is odd, the formula is slightly different. You get the credits as long as you get the idea.)

(5) Increase:
TCP: increases by 1/cwnd per ack (1 per RTT)
New: increase to half of current and max
The new one is more aggressive. If TCP increases as aggressive as the new one, there may be **a sudden surge of consecutive drops**.

Decrease:
TCP: decreases to 1 or decreases to half
New: decreases to half
TCP is sometimes more aggressive. Timeout indicates consecutive loss which is a result of **persistent in-network buffer overflow** (buffer in routers are full for a longer period of time).

3. (Network) Suppose NTU is allocated a block of IP address: 200.23.16.0/20. Based on your knowledge on IP addressing, address the following questions.
   (1) How many hosts at maximum can NTU supply with public IP addresses? (5%)
   (2) Suppose there are 8 colleges that NTU needs to distribute the IP addresses to. Suppose the IP address demand from each college is the same. What will be the block of IP addresses allocated to each college? How many hosts can each college supply with public IP addresses? (5%)
   (3) Suppose there are 6 colleges that NTU needs to distribute the IP addresses to. Two of the colleges request twice as much IP addresses as the other 4. What will be the block of IP addresses allocated to each college? How many hosts can each college supply with public IP addresses? (5%)
   (4) Suppose NTU's College of EECS is allocated with the block 200.23.30.0/23. Write out the network address of the EECS subnet in binary form. Suppose the NTU EECS gateway receives packets going to destinations: 200.23.28.1, 200.23.30.1, and 200.23.31.1. Write these IP addresses out in binary form. Should the gateway forward them all into the NTU EECS subnet? If not, which one(s) should be blocked? Can you list all IP addresses allowed in the NTU EECS subnet (xxx.xxx.xxx.xxx ~ xxx.xxx.xxx.xxx)? (10%)
   (5) If the NTU EECS subnet is growing so fast and exceeding the number of hosts allowed, what can the NTU EECS network administrator do to cope with the problem? (5%)

Sample Solution:

(1) 2^{12}

(2) 200.23.16.0/23
200.23.18.0/23
200.23.20.0/23
200.23.22.0/23
200.23.24.0/23
200.23.26.0/23
200.23.28.0/23
200.23.30.0/23
512 hosts for each college

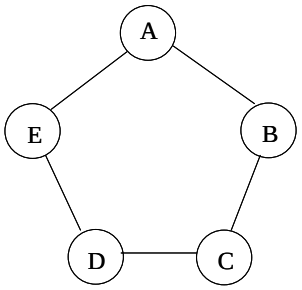
(3) 200.23.16.0/22
200.23.20.0/22

200.23.24.0/23
200.23.26.0/23
200.23.28.0/23
200.23.30.0/23
1024 hosts for 200.23.16.0/22 and 200.23.20.0/22
512 hosts for 200.23.24.0/23, 200.23.26.0/23, 200.23.28.0/23, 200.23.30.0/23
(Solutions will be accepted as long as there are 2 subnets of 1024 hosts)

(4) 11001000.00010111.00011100.00000001
11001000.00010111.00011110.00000001
11001000.00010111.00011111.00000001
The gateway should forward for 220.23.30.1 and 220.23.31.1, but block 220.23.28.1
The range is 200.23.30.0~200.23.31.255

(5) NAT (Or whatever feasible solution you can come up with)

4. (Network) Consider a 5-node ring network as follows. The link costs are all the same and value is 1. Follow the link state (LS) routing principle to obtain the routing table.



> ***if*** (LS report received on incoming link && the LS report has not been received before)
>
> ***then*** flood LS report onto all but the incoming link

(1) Suppose each node sends its link state (LS) report to all the outgoing links and then the LS reports are further propagated using the algorithm above to reach the whole network. Suppose the LS reports are the same in size, M bytes each. In order for all nodes to receive LS reports from all other nodes, how many bytes of LS reports are transmitted over the network? (5%)

(2) Continue from (1). Suppose the delay to send a LS report over any link is T seconds and the links are full-duplex. I.e., the packets will not collide. If A, B, C, D, and E nodes start sending their LS reports all at the same time, how much time does it take for all nodes **to receive** all LS reports? (5%)

(3) Continue from (1) and (2). Suppose all LS reports have arrived at all nodes. Compute the shortest paths from node E to every other node using the LS routing principle by filling in the blanks in the tables below. When 2 nodes have the same distance to the traversed set, pick the nodes to traverse next in alphabetic order. (5%)

(4) Derive the routing table for node A, B, C and D using LS routing. (10%)

(5) Suppose A is the multicast source. C and D are the multicast receivers. For each data packet to be sent from the source to every receiver, how many copies of the data will be transmitted over link A-B, B-C, C-D,D-E, and E-A for source-based tree multicast using RPF, assuming the unicast routing table from (4)? How many copies of the data will be transmitted over link A-B, B-C, C-D,D-E, and E-A for group-shared tree multicast centered at D, again assuming the unicast routing table derived from (4)? (5%)

(6) Answer the questions in (5) when A, B, C, D, and E are all receivers. (5%)

(7) Based on the observation in (5) and (6), argue which of the source-based vs. group-shared trees is more bandwidth efficient for dense groups and which is more bandwidth efficient for sparse groups. (5%)

Sample Solution:

(1) 30M

(2) 2T

(3)

Step	Travel Set	D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D), p(D)
0	E	1,E	∞	∞	1,E
1	EA	1,E	2,A	∞	1,E
2	EAD	1,E	2,A	2,D	1,E
3	EADB	1,E	2,A	2,D	1,E
4	EADBC				

(4)

A:

D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D),p(D	D(E),p(E)
0,A	1,A	2,B	2,E	1,A

B:

D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D),p(D	D(E),p(E)
1,B	0,B	1,B	2,C	2,A

C:

D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D),p(D	D(E),p(E)
2,B	1,C	0,B	1,C	2,D

D:

D(A),p(A)	D(B),p(B)	D(C),p(C)	D(D),p(D	D(E),p(E)
2,E	2,C	1,D	0,D	1,D

(5)

	Source-based	Group-shared
A-B	1	0
B-C	1	0
C-D	2	2
D-E	1	1

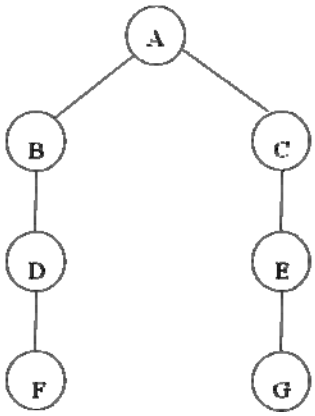
E-A	1	1

(6)

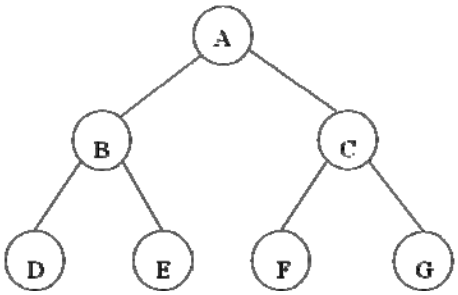
	Source-based	Group-shared
A-B	1	0
B-C	1	1
C-D	2	1
D-E	1	2
E-A	1	2

(7) Dense group: source-based tree or group-shared tree
Sparse group: group-shared tree

5. (Network) Consider the following 7-node networks below. One is a 7-node string, and the other is a 7-node tree. The link costs are all the same and the value is 1. Follow the distance vector (DV) routing principle to obtain the routing tables for every node.



7-node String



7-node Tree

(1) Suppose A is the node who starts sending its routing table. Suppose the delay to send a DV routing table over any link is T seconds. The links are full-duplex. I.e., the packets will not collide on the links. Address the following question for each of the 7-node networks. What is the amount of time it takes for all nodes **to converge** to the shortest paths to all other nodes? (Hint: convergence means the moment that the last DV table update is made) (5%)

(2) Suppose B is the node who starts sending its routing table. Suppose the delay to send a DV routing table over any link is T seconds. The links are full-duplex. I.e., the packets will not collide on the links. Address the following question for each of the 7-node networks. What is the amount of time it takes for all nodes **to converge** to the shortest paths to all other nodes? (Hint: convergence means the moment that the last DV table update is made) (5%)

(3) Observe from (1) and (2). Think for general strings and binary trees. Suggest the factor(s) that determine(s) the convergence time of DV routing. (10%)

Sample Solution:

(1) 7T vs. 4T

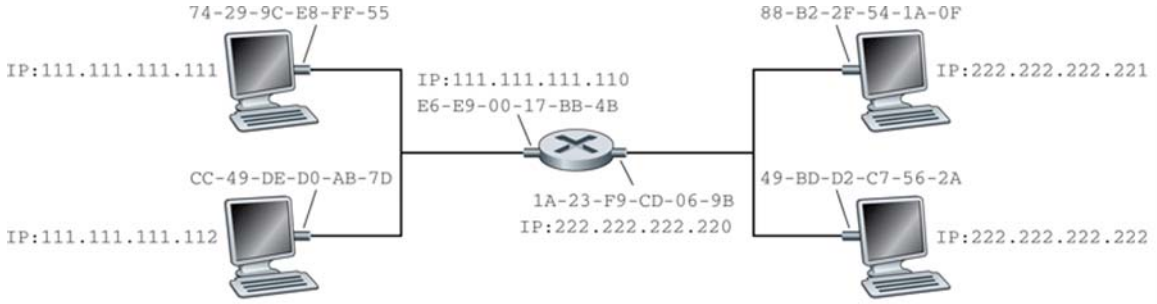
(2) 8T vs. 5T

(3) From the starting node, the DV needs to reach the node neighboring the farthest node to the starting node in order to learn about this ‘farthest node to starting node’. After that, the information about this node needs to be propagated to the other end of the network.

Convergence time = O(
Distance from starting node to farthest node in the network +
Farthest distance between two nodes in the network)

The two factors that determine the convergence time are: (1) Diameter of the network and (2) Position of the starting node. The longer the diameter, the longer the convergence time. The longer the farthest distance from the starting point to other nodes in the network, the longer the convergence time.

6. (Link) A node uses ARP to find mapping of the destination's IP and MAC. If the data destination is on the same subnet as the data source, the operation is straightforward. The operation gets slightly complicated if the data destination is not on the same subnet. Consider the following familiar scenario where destination is not on the same subnet. Suppose that the ARP tables on all nodes are empty and the data is going from 111.111.111.111 to 222.222.222.222. Complete the blanks in the packet frames in the packet flow provided below. (15%)



Sample Solution: