Name: _____

DEPARTMENT OF EECS
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.02 Fall 2012

Quiz III

December 18, 2012

"×" your section	Section	Time	Recitation Instructor	TA
☐	1	10-11	Victor Zue	Ruben Madrigal
☐	2	11-12	Victor Zue	Cassandra Xia
☐	3	12-1	Jacob White	Kyu Seob Kim
☐	4	1-2	Yury Polyanskiy	Shao-Lun Huang
☐	5	2-3	Yury Polyanskiy	Rui Hu
☐	6	3-4	Jacob White	Eduardo Sverdlin-Lisker

Please read and follow these instructions:

0. Please write your name in the space above and × your section.
1. There are a lucky **13 questions** (some with multiple parts) and a lucky **13 pages** in this quiz booklet.
2. Answer each question according to the instructions given, within **120 minutes**.
3. **Please answer legibly. Explain your answers.** If you find a question ambiguous, write down your assumptions. Show your work for partial credit.
4. Use the empty sides of this booklet if you need scratch space. *If you use the blank sides for answers, make sure to say so!*

Two two-sided "crib sheets" and a calculator allowed. No other aids.

Do not write in the boxes below

1-2 [15]	3-4 [20]	5-6 [13]	7 [8]	8-9 [14]	10 [14]	11-12 [16]	Total [100]

I Packet-Switching Basics

1. [5 points]: Which of the following actions does a **switch** S in a best-effort packet-switched network (as studied in 6.02) perform **when it receives a packet**? Circle **True** or **False** for each choice.

A. True / False S sends an ACK to the sender of the packet.

B. True / False S looks up the destination address of the packet in its routing table.

C. True / False S sends a routing advertisement to its neighbors.

D. True / False S decrements the hop-limit field in the packet header.

E. True / False S guarantees that packets are forwarded toward the destination in the order in which they arrive.

2. [10 points]: A sender is connected to a switch using a link whose transmission rate is 2 Gigabits/second. The switch is connected to the receiver using a link whose transmission rate is 1 Gigabit/second. The packet size is 10 kbits. Each link is 100 meters long, and the speed of signal propagation over each link is 2×10^8 meters/second. The switch receives all the bits of a packet from the first link, and then takes 10 microseconds to process the packet, after which it enqueues the packet for transmission on the 1 Gigabit/s link. On average, there are 4 packets in the queue at the switch. There are no queues anywhere else, nor any other delays.

Calculate the **average delay, in microseconds** (accurate to one microsecond), from the initiation of a packet transmission at the sender to the completion of its reception at the receiver. Show your work.

II MAC

3. [4+6=10 points]: Answer the following questions about MAC protocols as studied in 6.02.

A. In a certain shared medium network with N total nodes, we know that at any time, only $k < N$ of the nodes are actually backlogged. Each packet is 1 time slot in length. Express the condition under which the **maximum possible utilization of slotted Aloha with a fixed probability of transmission** exceeds that of time-division multiple access (TDMA) in this network.

B. Circle **True** or **False** for each statement below.

(a) **True / False** Using contention windows in stabilized slotted Aloha guarantees that a backlogged node will attempt to transmit a packet within a finite number of time slots.

(b) **True / False** Using contention windows in stabilized slotted Aloha guarantees that a backlogged node's packet transmission will successfully be received within a finite number of time slots.

(c) **True / False** The Carrier Sense Multiple Access (CSMA) protocol with stabilization, where the packet size is larger than the length of one time slot, never suffers from collisions.

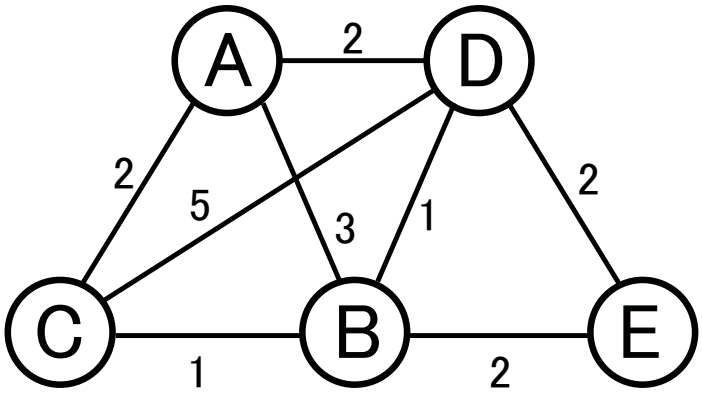
4. [4+6=10 points]: Annette Werker has set up a wireless network with an access point (AP) and N clients. Each packet is sent either from the AP to a client, or from a client to the AP. The AP and clients use slotted Aloha to share the wireless channel. Annette configures the AP to transmit a packet with probability p when it is backlogged, and configures each client to transmit a packet with probability q when it is backlogged. Assume that all clients and the AP are always backlogged and that each packet is one time slot long.

A. Derive an expression for the **utilization** of the network. Show your work.

B. If Annette's goal is for the throughput of the AP's transmissions to be equal to the aggregate throughput summed over all the client transmissions, derive an expression for p in terms of the other specified parameters. Show your work.

III Routing

5. [2+3+3=8 points]: The 6.02 link-state protocol runs in the network below (the numbers are link costs).



A. Node E receives HELLO protocol messages from nodes _____ (list all that apply).

B. Suppose node E has a bug in its implementation, which prevents it from correctly receiving any LSAs originating from nodes A and C. Circle **True** or **False** for each choice below.

(a) **True / False** E will compute a route to every node in the network.

(b) **True / False** E will correctly compute a minimum-cost route to every node in the network.

C. Suppose node E has a bug in its implementation, which prevents it from correctly receiving any LSAs originating from nodes B and C. Circle **True** or **False** for each choice below.

(a) **True / False** E will compute a route to every node in the network.

(b) **True / False** E will correctly compute a minimum-cost route to every node in the network.

6. [5 points]: In a network running a **link-state** routing protocol, exactly one node is buggy. Instead of Dijkstra's algorithm to minimize path costs, the buggy node runs a different algorithm, as follows:

1. Sort the links in the network in non-decreasing order of their link costs.
2. Create a tree rooted at the switch by adding links to the tree from the sorted list, proceeding in link-cost order. Add a link only if it does not create a cycle with the links already added, continuing until there is a path in the tree from the node to every other node in the network.
3. For each destination, select as the route the link in the computed tree that connects the node to the destination.

The other nodes correctly implement Dijkstra's algorithm.

Give an example of a four-node network topology, with link costs and an example path, in which this method produces a **routing loop**. Clearly indicate the buggy node and specify the routing loop.

(Aside: The algorithm at the buggy node computes a "minimum spanning tree", and is called Kruskal's algorithm.)

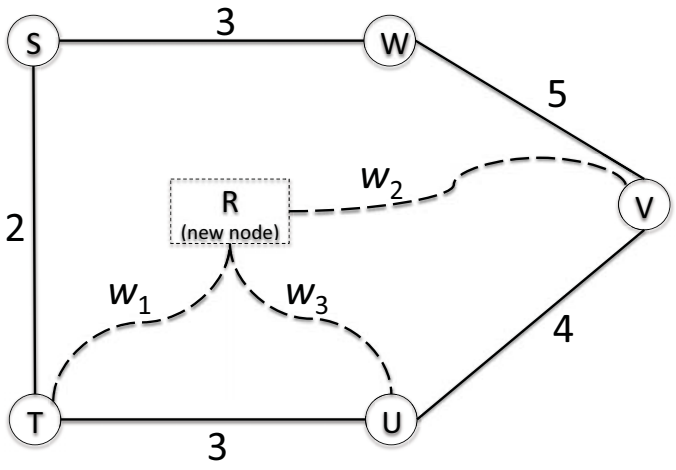
7. [8 points]: Alice has an unreliable network in which she uses a **distance-vector** approach to compute **most reliable** paths between nodes. The most reliable path from node S to a destination D is a path from S to D along which packets are *least likely* to be lost.

The link metric in this network is the packet loss probability for each link: each packet sent on link i is lost independently with probability ℓ_i, where $0 \leq \ell_i < 1$. Each node is able to compute this value for each of its links, and you may assume that this probability is the same in each direction of a link. If the routing protocol were a distance-vector protocol whose link metrics are these values, write the `integrate` step in Python.

```
def integrate(self, link, adv):
  # Your code here: explain what any new variable you introduce does.
  # adv is a vector of (dest,lossprob) tuples coming from the
  # neighbor at the other end of the link. The lossprob in the
  # advertisement is the cumulative loss probability from the node
  # making the advertisement for destination "dest".
  # Assume that the variable link.loss stores the loss rate of "link".
  # Your code should correctly compute self.routes[dest] for each dest
  # as well as self.lossprob[dest], the path metric to the destination,
  # which should be equal to the loss probability of the path to dest.
```

8. [10 points]: Ben Bitdiddle operates a network whose routing protocol computes minimum-cost paths between nodes. All link costs are **positive integers**. Initially the network has five nodes, S, T, U, V, and W, with link costs as shown in the picture below. After the routing protocol has converged to minimum-cost routes at each node, Ben adds a new node, R, to the network, with the links and link costs as shown. When the routing protocol converges, he finds that:

1. S's route to destination V has changed,
2. T's route to destination U has changed, and
3. U's route to destination V has **not** changed.



In the routing protocol, an existing route changes only if the new route has a lower cost; if there is a tie, the old route persists.

To satisfy **all of the three observations above**, some constraints on the positive-integer link costs (w_1, w_2, and w_3) must necessarily hold. Specify these constraints, with complete explanations.

9. [4 points]: The picture below shows the ARPANET from September 1971. At that time, the network ran a **distance-vector** protocol. Assume that each link has a cost of 1.

Suppose CARNEGIE believes it is MIT, sending out erroneous distance-vector advertisements that make it look like CARNEGIE is actually MIT: these route advertisements are of the form [(MIT 0), (CASE 1), (MITRE 1), ...]. The rest of the protocol runs correctly at all the nodes. There are no packet losses or other failures. Note that the nodes BBN and BBN-T (the "T" marked with a circle) are distinct nodes.

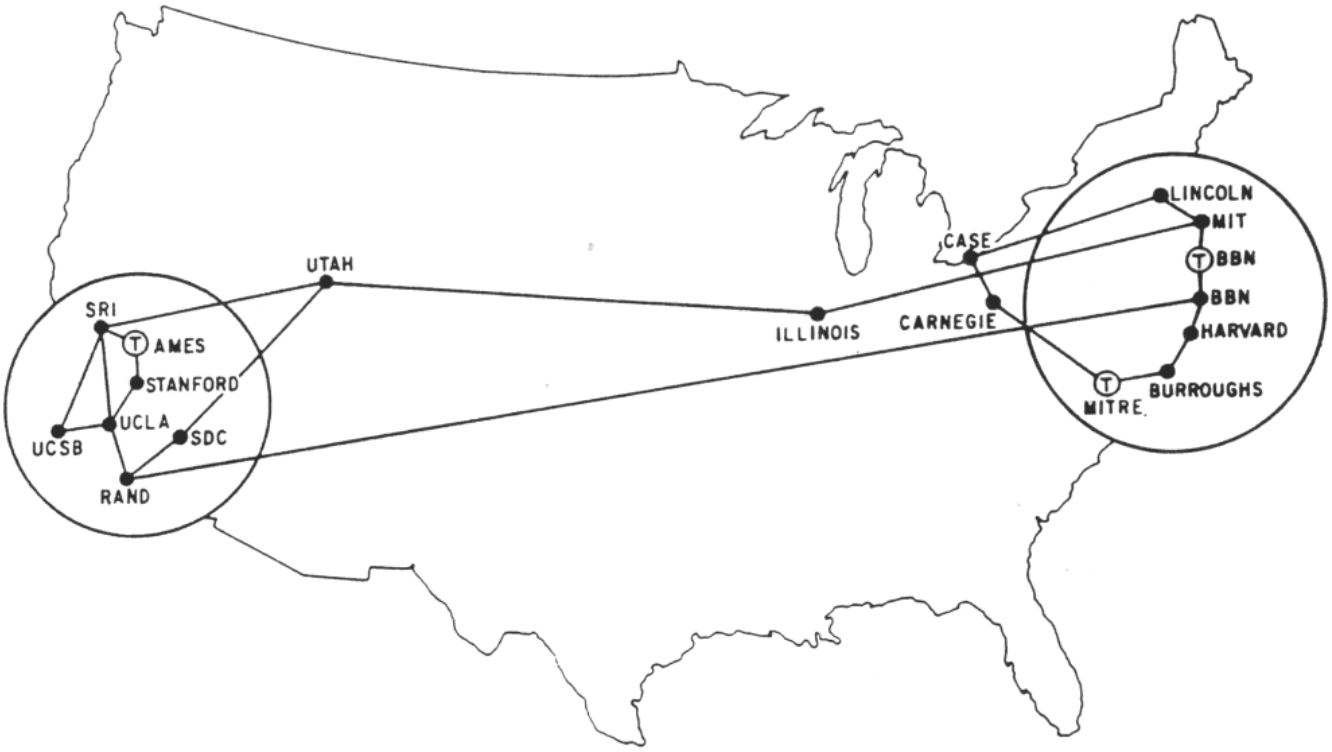


Image in the public domain, from the *ARPANET Completion Report*, January 1978.

Packets sent **from** which nodes to MIT will **definitely not reach** the true MIT now? Explain.

IV C-DNS: Concurrent Domain Name System Queries

The Domain Name System (DNS) is a critical component of the Internet infrastructure. It provides a way to convert from human-readable hostnames to IP addresses. It achieves this task using a simple request-response protocol similar to the 6.02 stop-and-wait protocol:

1. The client seeking to resolve a hostname sends a request to a DNS server.
2. The server responds with the answer.
3. If the client does not get a response within a timeout period, it retransmits the request to the server.

Alyssa P. Hacker is interested in improving the response time for DNS requests. She modifies the system to set up two **independent** DNS servers, 1 and 2. In the modified system, the client sends each request to the two servers **concurrently**. If it does not get a response from either of the two servers within τ seconds of the transmission of the original request, the client retransmits the request to both servers. The client continues to do that until it obtains a response.

The RTT between the client and server i is R_i (each R_i is a fixed value). The probability that any given request sent to server i will get a response is $p_i > 0$. The paths between each of the two servers and the client are independent of each other. Assume that $R_1 \leq R_2$ and that the timeout value $\tau > R_2$ is a constant.

10. [4 points]: Derive an expression for T_i, the expected amount of time for the client to receive a response from server i. Explain your answer.

11. [10 points]: Derive an expression for the expected amount of time before the client obtains the **first response from one of the servers**. (Remember that $R_1 \leq R_2$.) Explain your answer.

V Sliding Windows

12. [10 points]: Running the 6.02 sliding window protocol on a network path, Eager B. Eaver observes the following experimental results during the steady state of the connection. There is no other traffic along the network path, and there are no data packet losses or ACK packet losses.

Fill in the blanks in the table below. Show your calculations in the space below the table.

Window size (packets)	RTT (ms)	Throughput (packets/s)	Average queue length (packets)
15	30	_____	0
21	_____	_____	0
_____	40	_____	_____
_____	_____	800	1

13. [6 points]: Louis Reasoner decides to optimize the 6.02 sliding window protocol for a best-effort packet network. When the sender gets an acknowledgment (ACK) for packet with sequence number i, if it has not already received an ACK for any packet with sequence number $j < i$, it immediately retransmits each such packet with sequence number j. The rest of the protocol is the same as we studied in 6.02.

In Louis's network, **ACK packets are never lost**, and timeouts are large enough to not cause spurious retransmissions. Recall that a spurious retransmission is a situation when the sender retransmits a packet that has not actually been lost.

Circle **True** or **False** for each choice below, **with an explanation for your answer in each case**.

A. True / False Suppose the network reorders the packets sent between the sender and receiver, but not between the receiver and sender. Then, Louis's protocol may cause spurious retransmissions.

B. True / False Suppose the network **never reorders packets** on any path. Then, Louis's protocol will **never** cause spurious retransmissions.

FIN

Have a great winter!

6.02 Introduction to EECS II: Digital Communication Systems

Fall 2012