EECS 489: Final Exam, Fall 2018

Duration: 70 Minutes

OFFLINE, open-book, open-notes exam.   You may consult your own notes, notes from discussion sections, programming assignments, your solutions to them, any solutions provided by the instructors, and other notes provided by the instructors, including the course lecture slides. The above listed material may be accessed in hard copy or electronic form, offline. A calculator or a calculator program is also permitted for the exam. These are the only uses of a computer permitted. Any other use of the computer and accessing the Internet/local network is strictly forbidden. You are also not allowed to compile and run any programming code during the exam. You must not consult any resources other than those listed above.

Write legibly.   If the grader cannot read what youve written, they will assume that you meant the illegible portion as a note to yourself and will ignore it. If you lose points because part of your answer is illegible, you will not be given the opportunity to explain what it says. Illegible scribble will earn zero points.

Do not ask questions during the exam.   They disturb other students. Figuring out what the question is asking is part of the exam. If you think you have to make some assumption to answer a problem, note your assumption on the test. The answers to most questions should be short; you need not use all the space provided to answer the questions. If you find yourself writing an excessively long response, you may want to think more carefully about the question.

Write your uniqname on the lower-left corner of every page.

Honor code pledge.   You are to abide by the University of Michigan/Engineering honor code. Sign below to indicate that you have kept the honor code pledge.
   "I have neither given nor received unauthorized aid on this examination, nor have I concealed any violations of the Honor Code."

Name:

Signature:

Uniqname:

|	Q1	Q2	Q3	Q4	Q5	Total
Grade	/20	/12	/12	/11	/15	/70

1 [20 points] True or False

2 points for a correct answer; **-1 point** for a wrong answer.

(i) *(True / False) Physical layer is implemented in every networking hardware.*

(ii) *(True / False) Forwarding plane works at a slower time scale than control plane.*

(iii) *(True / False) There can be loops in BGP-announced AS paths.*

(iv) *(True / False) The minimum number of addresses assigned to an organization in the classful addressing scheme is 2.*

(v) *(True / False) Datacenter networks always have full bisection bandwidth.*

(vi) *(True / False) Using source routing, we can reduce the number of rules in every switch/router to 0.*

(vii) *(True / False) TCP reacts in proportion to the extent of congestion.*

(viii) *(True / False) The minimum spanning tree on a connected graph can be used to calculate valid routing states.*

(ix) *(True / False) ARP and DHCP both rely on soft states.*

(x) *(True / False) Switched Ethernet relies on CSMA/CD to detect collisions.*

2 [12 points] Max-Min Fairness

Consider the network below with 8 nodes. The operator decides to use shortest-path routing as the inter-domain routing protocol. The cost of each link is shown in the figure. Assume every link has 100 Mbps bandwidth in each direction.
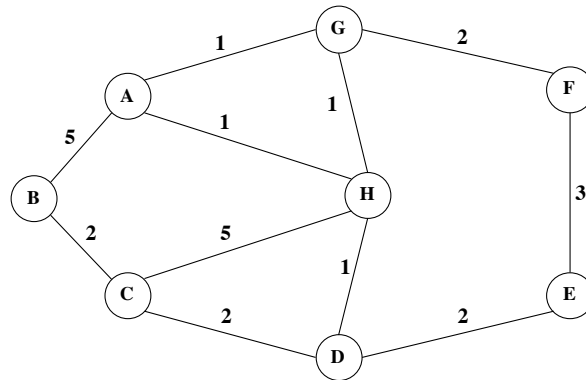
Given the source, destination, and demand of the following TCP flows ($src \to dst; demand$), what are the *steady-state* max-min fair share allocations of each of them on the given network? Assume that they are all active simultaneously. Write down the shortest path (e.g., $x \to y \to z$) and its cost as well as max-min fair allocation in the steady state and the bottleneck link (e.g., $a \to b$) that caused the allocation.

Assume that there is no loss in the network.

(i) $A \to E$; 100 Mb/s
   Shortest path:
   Cost:
   Allocation:
   Bottleneck link:

(ii) $B \to H$; 30 Mb/s:
   Shortest path:
   Cost:
   Allocation:
   Bottleneck link:

(iii) $E \to G$; 80 Mb/s:
   Shortest path:
   Cost:
   Allocation:
   Bottleneck link:
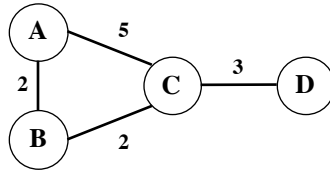

(iv) $F \to D$; 40 Mb/s:
   Shortest path:
   Cost:
   Allocation:
   Bottleneck link:

3   [12 points] Intra-Domain Routing

The distance vector (DV) routing algorithm is known to cause routing loops.

(i) **[6 points]** The network shown above uses DV, and forwarding tables of all its nodes have converged. Find one *single link failure* that will cause a routing loop to form (and thus lead to the count-to-infinity situation). Why? Will it be resolved on its own? Assume that poison reverse is not in use.

(ii) **[5 points]** Now assume that we have poison reverse enabled in this network. The idea is, if a node X uses node Y as its next hop to get to node Z, then X would advertise a cost of ∞ for node Z when sending a distance vector update message to node Y. Do you think poison reverse is sufficient to prevent the routing loop if the link between C and D fails? Explain your answer in detail.

(iii) **[1 points]** How would BGP have addressed this?

4   [11 points] Inter-Domain Routing

Consider the following network, where all ASes run OSPF as their intra-AS routing protocol, and eBGP and iBGP for inter-AS routing.
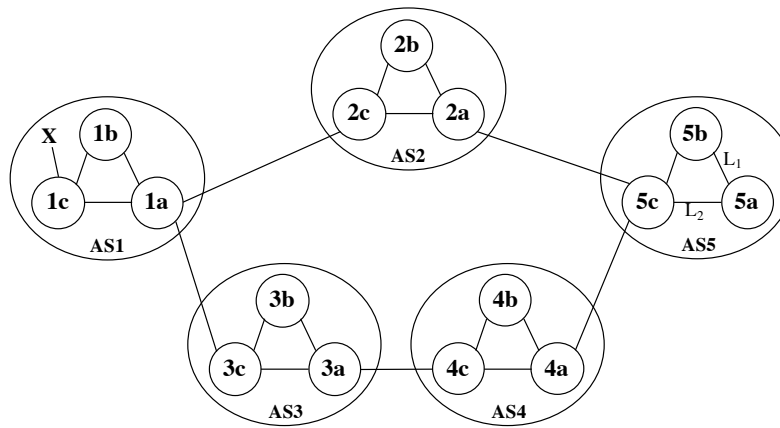
All use the following rules for route selection in priority order:

Priority	Rule
1	iBGP path length
2	ASPATH
3	eBGP > iBGP

(i) **[1 points]** Router $3c$ learns about prefix X from which routing protocol?

(ii) **[1 points]** Router $1b$ learns about prefix X from which routing protocol?

(iii) **[1 points]** Router $4b$ learns about prefix X from which routing protocol?

(iv) **[4 points]** Once router $5a$ learns about X, it will put an entry (X, L) in its forwarding table. Will L be equal to L_1 or L_2 for this entry? Why?

(v) **[4 points]** Now suppose that the link between $2a$ and $5c$ has failed. Some time later, a new link between $2a$ and $5b$ has been established. Will the updated entry (X, L) have L equal to L_1 or L_2? Why?

5 [15 points] CSMA

Consider 3 hosts – A, B, and C – that are connected to a Layer-2 broadcast medium. The propagation delays between A and B and between B and C are both 1 ms each. Signals from A do not reach C and vice versa. A has to send data to B that takes 3 ms to transmit (i.e., transmission time is 3 ms). C also has to send data to B with a transmission time of 10 ms. A starts transmitting at 0 ms and C starts at 4 ms.

Given the broadcast medium, collisions are inevitable. A and C both use deterministic exponential back-off whenever a backoff is necessary. Specifically, A waits 2^i ms and C waits 4^i ms, where i is the i-th time a host attempts to retransmit; $i \geq 1$. For example, the first time C has to back-off for whatever reason, it backs off for 4 ms; the next back-off duration for C is 16 ms, and so on.

Assume that if a host senses carrier, it continues to listen for carrier and can transmit immediately once carrier ceases. Also assume that transmission times for all signals other than data are 0 ms.

(i) **[1 points]** What is a suitable MAC protocol in this environment?

(ii) **[12 points]** Draw below all control and data message exchanges that will occur using the MAC protocol you have selected along with any required back-offs. Each exchange is a labeled arrow, where the label has the message type and the arrow starts from the source and ends at the destination for that message. Mark timestamps, but the markings need not be drawn to scale. No need to write message formats.

A   B   C

0

4

Time (milliseconds)
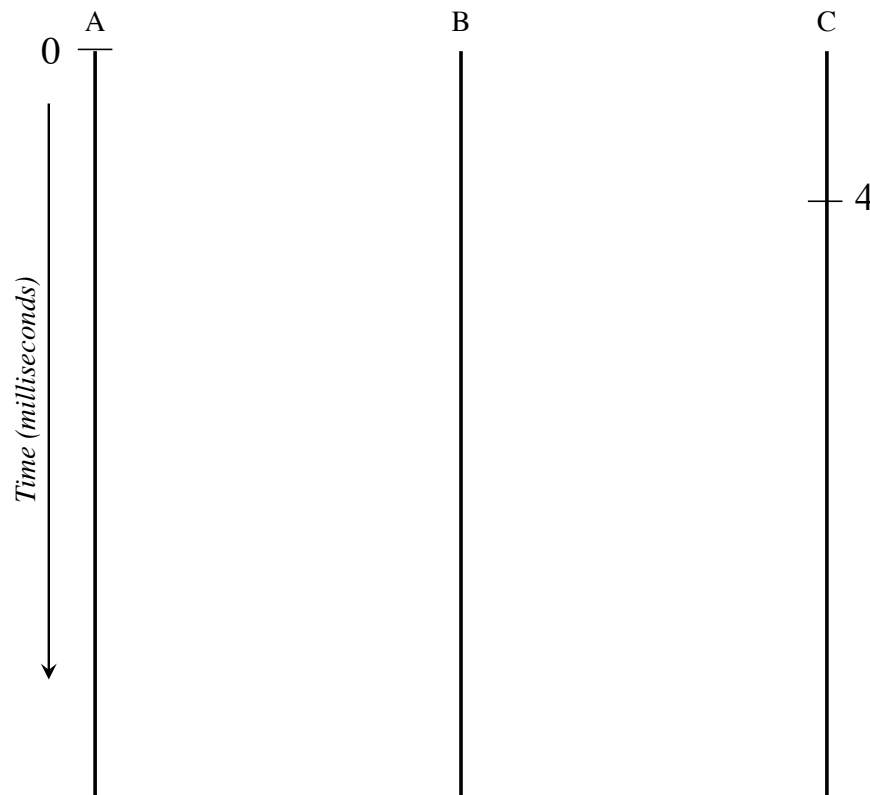
(iii) **[2 points]** What is the completion time of A's transmission? What about C's?