CS 168
Spring 2024

Introduction to The Internet

Midterm

PRINT your name: ____________________________, ____________________________
(last) (first)

PRINT your student ID: ____

PRINT the SID of your left/right neighbors and check their photo.

SID on left: ____________________________ SID on right: ____________________________

- You have 110 minutes.
- If you have a question during the exam, raise your hand and a proctor will come over to you. We will not answer your question on the spot, but we will make a clarification to everyone if we believe it's necessary.
- If you need to use the restroom, please bring your exam, answer sheet, phone, and ID to the front. Only one student is allowed to use the restroom at a given time.
- If you finish early, please bring your exam and answer sheet up to the front. **You may not leave early once there are 15 minutes remaining in the exam.**

For questions with **circular bubbles**, you may select only one choice.

○ Unselected option (completely unfilled)

● Only one selected option (completely filled)

For questions with **square checkboxes**, you may select one or more choices.

☐ You can select

■ multiple squares (completely filled)

Q1 *Honor Code* (0 points)

Read the following honor code and sign your name.

> I understand that I may not collaborate with anyone else on this exam, or cheat in any way. I am aware of the Berkeley Campus Code of Student Conduct and acknowledge that academic misconduct will be reported to the Center for Student Conduct and may further result in, at minimum, negative points on the exam.

SIGN your name: ____

Q2 *True/False* **(10 points)**

Q2.1 (1 points) In a layered architecture, layers provide modularity and allow for abstraction.

● True ○ False

Q2.2 (1 points) There is only one protocol available at L4.

○ True ● False

Q2.3 (1 points) The end-to-end principle ensures that packets are always securely/reliably transferred from one host to another.

○ True ● False

Q2.4 (1 points) eBGP exchanges routes betweeen ASes while iBGP creates routes between destinations within an AS.

○ True ● False

Q2.5 (1 points) Loss is always a sign of congestion in the network.

○ True ● False

Q2.6 (1 points) The TCP header is the outermost header on packets that use TCP.

○ True ● False

Q2.7 (1 points) For most networks, the peak of aggregate demand for resources is greater than the aggregate of peak demands.

○ True ● False

Q2.8 (1 points) Circuit switching is easier to implement than packet switching

○ True ● False

Q2.9 (1 points) Per-flow fair queueing provides isolation between flows.

● True ○ False

Q2.10 (1 points) Sockets are the interface between the application layer and the network layer.

○ True ● False

Q3 *Multiple Choice* **(10 points)**

Q3.1 (2 points) Propagation delay

☐ Tracks the end-to-end transmission time of the entire packet.

☐ Dominates total latency for extremely large packets.

☐ Depends on transmission delay.

■ Is proportional to the physical distance between two hosts.

☐ None of the above

Q3.2 (2 points) Consider a fixed route between a source and destination host. Which of the following delay components are constant between packets sent?

■ Propagation

☐ Transmission

☐ Queuing

☐ None of the above

Q3.3 (2 points) Which of the following is true?

☐ The host only has to implement the application and transport layers.

■ Routers implement the network, data link, and physical layers.

☐ Data reliability needs to be handled by switches.

☐ There are multiple protocol choices for every networking layer.

☐ None of the above

Q3.4 (2 points) What is the purpose of the history data structure in Project 1?

☐ To determine which routes are stale.

☐ To implement triggered updates.

■ To keep track of which neighbors have received which advertisements.

☐ To keep track of which advertisements the local router has received.

■ To implement incremental updates.

☐ None of the above

Q3.5 (2 points) Consider a link shared by flows from five hosts, all destined for a single server. Each host is connected to the shared link by its own access link. The access bandwidths are as follows:

- Host H1: 0.5 Gbps
- Host H2: 1.0 Gbps
- Host H3: 1.5 Gbps
- Host H4: 2.0 Gbps
- Host H5: 3.0 Gbps

Assume all hosts have an infinite amount of data to send, and try to send as fast as their access link will allow. If the capacity of the shared link is 6 Gbps, what are the max-min fair allocations for each host? (all in Gbps)

● H1=0.5, H2=1.0, H3=1.5, H4=1.5, H5=1.5

○ H1=1.2, H2=1.2, H3=1.2, H4=1.2, H5=1.2

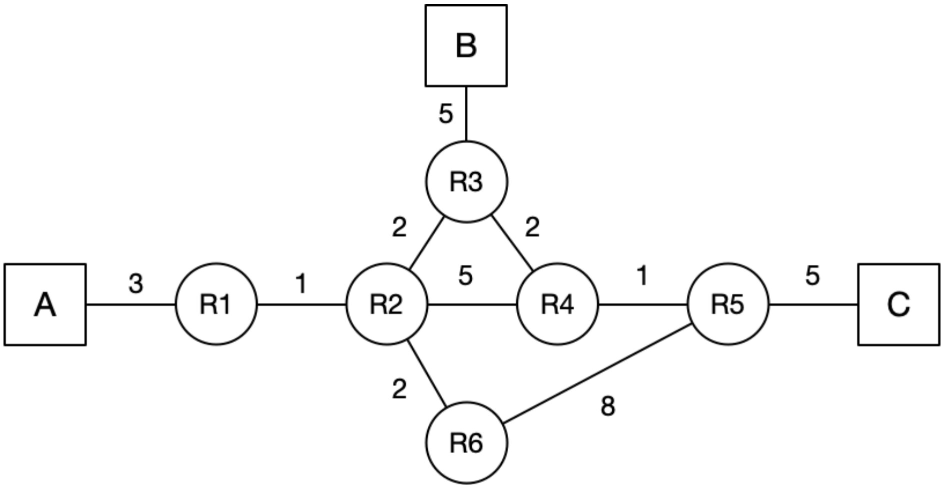
○ H1=0.5, H2=1.0, H3=1.2, H4=1.2, H5=1.2

○ H1=0.5, H2=1.0, H3=1.2, H4=1.5, H5=1.8

○ None of the above

Q4 *Link State* (15 points)

Consider the following network graph with three hosts (A, B, C) and six routers (R1 - R6).



For the following questions, assume that the routers run a link-state routing protocol and the routing state has converged. Every link is up unless otherwise noted. When picking between equal-cost paths, the routers pick the route through the neighbor with the lower ID number.

Note that all subparts are independent of one another (changes made in one subpart do not affect the subsequent ones).

Q4.1 (2 points) Suppose that the link between R3 and R4 goes down. R3 and R4 have recomputed their routes, but they have not yet sent updates. What route will a packet from A to C take?

Solution: A → R1 → R2 → R3 → R2 → R3 (loop)

Q4.2 (2 points) Suppose that the link between R2 and R3 goes down. R2 and R3 have recomputed their routes, but have not yet sent updates. What route will a packet from A to C take?

Solution: A → R1 → R2 → R4 → R5 → C

Q4.3 (4 points) Suppose that our routers failed to track sequence numbers in link-state advertisements when sending link state advertisements to each other.

a) In less than 20 words, describe why this would cause problems.

Solution: Do not know which advertisements you have seen before / sent to other routers → can create loops

b) As a quick fix, which **single router** can you remove in this topology to fix the above problem without partitioning the network?

○ R1 ○ R3 ○ R5

○ R2 ● R4 ○ R6

Q4.4 (3 points) Consider a scenario in which **exactly one** router is misconfigured such that it **chooses the next hop with the highest-cost path instead of the least-cost path**. Select the routers such that if the selected router is the **only** router that is misconfigured, the network will have valid routing state.

A valid routing state ensures that a packet will reach its destination; this does **not** guarantee that the packet takes the shortest route. Note that the misconfigured router will still choose valid routes.

■ R1 ☐ R3 ■ R5

☐ R2 ☐ R4 ■ R6

Q4.5 (4 points) Assume that at time t=0, A sends a packet to C. At t=0.5 seconds, the link between R4 and R5 goes down, and R4 and R5 instantaneously recognize and recompute their routes.

Assume that link-state advertisements are processed and propagated instantaneously. A link's propagation delay is equal to the link costs in the diagram (in seconds), i.e. R1 - R2 has a 1-second delay, R2 - R3 has 2-second delay, etc). You can ignore all processing and queuing delays.

Does the packet reach its destination? If so, write down the route the packet from A to C takes, if not explain why in 10 words or less.

● Yes ○ No

Solution: A → R1 → R2 → R3 → R2 →R6 → R5 → C

Q5 *Forwarding* (18 points)

Consider a router in a network that uses a least-cost routing protocol, with ties broken by taking the route from the link with the smallest port number. The router has 4 ports and its default route sends all traffic onto port 1. Table 1 lists the routes that our router sees advertised at each port.

You can find some useful binary conversions in the table below.

Port	Destination	Cost
1	1.0.0.0/8	10
	2.1.0.0/16	15
	2.2.192.0/20	12
	4.0.0.0/8	10
	2.2.96.0/17	15
2	1.1.0.0/16	8
	2.2.128.0/17	14
	4.0.0.0/8	8
3	3.0.0.0/8	10
	2.2.204.0/20	13
	1.0.10.0/24	8
4	3.4.0.0/16	11
	1.1.0.0/16	8
	2.2.0.0/17	14

Table 1: Routes Advertised at each port

Decimal	Binary
192	11000000
128	10000000
96	01100000
204	11001100
208	11010000
64	01000000
32	00100000

For the following 7 subparts, determine which ports the packets with the following destinations are forwarded to based on the advertisements given above.

Q5.1 (2 points) A packet with destination 3.4.0.1

○ 1 ○ 2 ○ 3 ● 4

Q5.2 (2 points) A packet with destination 4.0.0.1

○ 1 ● 2 ○ 3 ○ 4

Q5.3 (2 points) A packet with destination 2.2.208.1

○ 1 ● 2 ○ 3 ○ 4

Q5.4 (2 points) A packet with destination 2.3.0.10

● 1 ○ 2 ○ 3 ○ 4

Q5.5 (2 points) A packet with destination 2.2.204.13

● 1 ○ 2 ○ 3 ○ 4

Q5.6 (2 points) A packet with destination 1.1.21.7

○ 1 ● 2 ○ 3 ○ 4

Q5.7 (2 points) A packet with destination 2.2.96.22

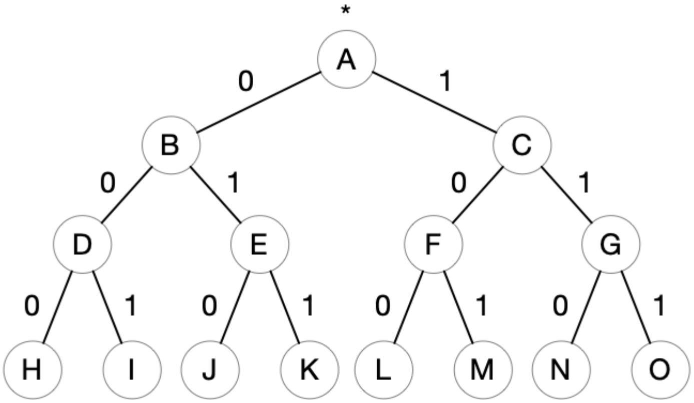
○ 1 ○ 2 ○ 3 ● 4

Q5.8 (4 points) Now consider a router with the following forwarding table:

Destination	Port
0.0.0.0/0	1
32.0.0.0/3	2
64.0.0.0/3	2
96.0.0.0/3	3
128.0.0.0/3	4

Fill out the table below with the port value that should be associated with each node in the prefix tree in order to implement the forwarding table of the router.

Note that some rows may not be used; put NA on any unused rows. Your tree must use the minimum number of nodes possible (meaning the fewest nodes in the tree to reach all labeled nodes).



Node	Port
A	
B	
C	
D	
E	
F	
G	
H	
I	
J	
K	
L	
M	
N	
O	

Solution:

Node	Port
A	Port 1
B	Port 2
C	NA
D	NA
E	NA
F	NA
G	NA
H	Port 1
I	NA
J	NA
K	Port 3
L	Port 4
M	NA
N	NA
O	NA

Q6 *BGP* (26 points)

Answer the subparts 1 to 6 based on the following autonomous system network topology. Assume all ASes have some hosts in them.

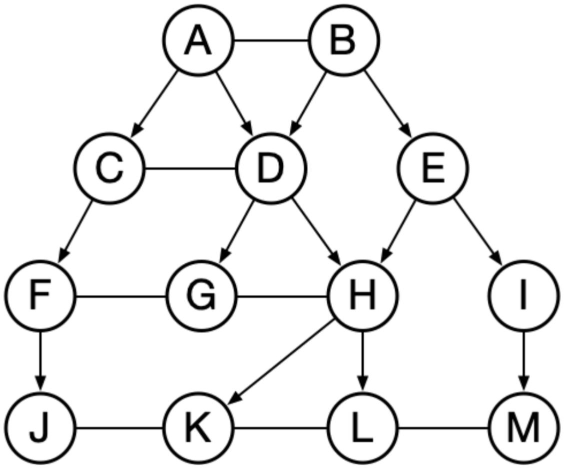


Figure 1: Network Topology

In Fig.1 a directed edge (→) denotes a provider - customer relationship while an undirected edge (-) denotes a peering relationship. Assume that the import and export policies follow Gao-Rexford rules (unless otherwise specified).

Note that all subparts are independent of one another (changes made in one subpart do not affect the subsequent ones).

Q6.1 (6 points) Determine the path (if any) between the following ASes. If there is no path possible, write "None".

a) J to M?

Solution: J → F → C → A → B →E →I→M

b) L to F?

Solution: L →H → D → C → F

c) E to F?

Solution: E → B → A → C → F

Q6.2 (2 points) Hosts under which ASes would be advertised to AS C by AS D?

Solution: D, G, H, K, L

Q6.3 (3 points) The link between CD fails, what is the path from J to L?

Solution: J → F → C → A → D → H → L

Q6.4 (4 points) Assume the entire network implements a **modified Gao-Rexford policy** where ASes will **also** export routes received from their peer(s) to their provider(s).

Determine the path (if any) between the following ASes. If there is no path possible, write “None”.

a) K to M?

Solution: K → H → L → M

b) L to F?

Solution: L → H → D → G → F

Q6.5 (2 points) Assume **none of the peered connections exist**. Which of the following ASes can M **no longer** communicate with?

- ■ A
- ☐ B
- ■ C
- ☐ D
- ☐ E
- ■ F
- ☐ G
- ☐ H
- ☐ I
- ■ J
- ☐ K
- ☐ L

Q6.6 (2 points) Assume AS H wants to implement a policy that favors advertisements from AS D over AS E. Which BGP attribute does AS H need to update to implement this policy?

Solution: `LOCAL_PREF`

For the rest of the question, assume the interdomain network topology defined by the figures below. Figure 2 shows the business relationships between the ASes while Figure 3 shows how the ASes are interconnected via their border routers.

Again, a directed edge (→) denotes a provider - customer relationship while an undirected edge (-) denotes a peering relationship. Assume that the import and export policies follow Gao-Rexford rules (unless otherwise specified).

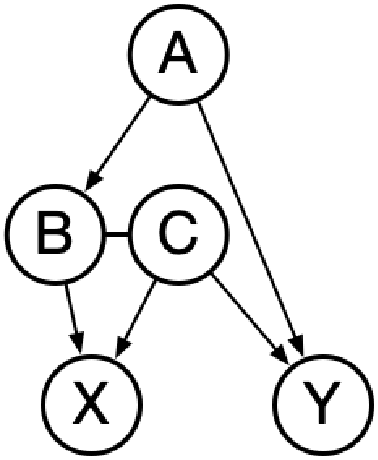


Figure 2: AS Relationship Graph

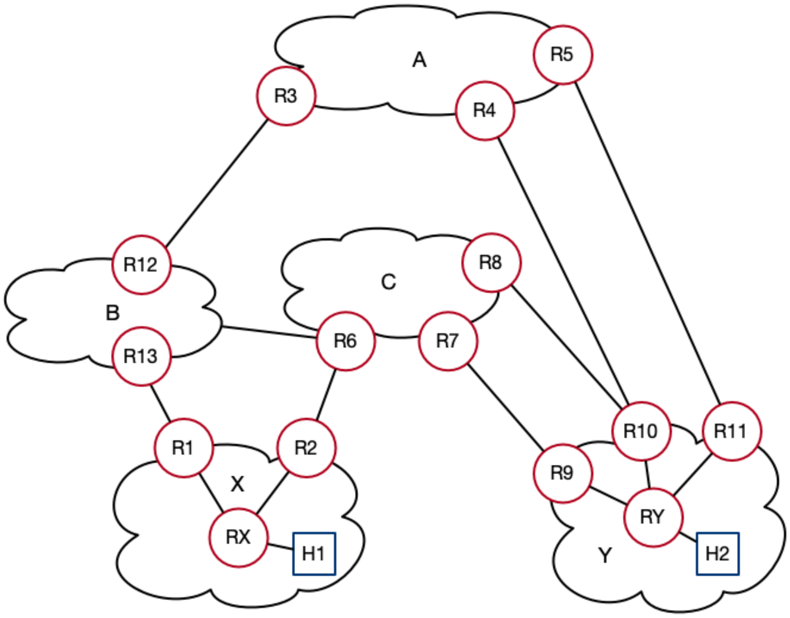


Figure 3: AS Border Router Topology

Assume the following for the rest of the question:

- All ASes can only assign values 10 and 20 for MED
- All ASes follow the standard import/export policies described in the class.
- In the case of multihoming, providers are assigned the same `LOCAL_PREF`.
- The relevant IGP costs are as follows:
  - **AS X**: IGP_cost(RX, R1) = IGP_cost(RX, R2)
  - **AS A**: IGP_cost(R3, R4) < IGP_cost(R3, R5)
  - **AS C**: IGP_cost(R6, R7) < IGP_cost(R6, R8)
  - **AS Y**: IGP_cost(RY, R10) < IGP_cost(RY, R9) < IGP_cost(RY, R11)

Q6.7 (3 points) Fill out the table below with all of the BGP advertisements AS Y receives for the destination AS X. Include the source and destination routers for the respective advertisement. You may not need all the rows of the table.

AS Path	**MED**	**Source Router**	**Destination Router**

Solution:

AS Path	**MED**	**Source Router**	**Destination Router**
X - C - Y	10	R7	R9
X - C - Y	20	R8	R10
X - B - A - Y	10	R4	R10
X - B - A - Y	20	R5	R11
X - B - C - Y	10	R7	R9
X - B - C - Y	20	R8	R10

Explanation: Each AS chooses a lower MED value for the routes they prefer their neighbor to use. This preference is based on minimizing the IGP cost of that route for the AS.

Q6.8 (2 points) Assuming that AS Y wants to implement a **hot-potato** policy, identify the advertisement that RY will import when sending packets to H1.

AS Path	**MED**	**Source Router**	**Destination Router**

Solution:

AS Path	**MED**	**Source Router**	**Destination Router**
X- C - Y	20	R8	R10

Explanation: AS Y will import the advertisement with the shortest AS PATH (since all options come from providers with the same LOCAL PREF). To choose between the two shortest options, it then imports the route that will implement hot-potato routing (get rid of the traffic as fast as possible), which means exiting via R10 since it has the lowest IGP cost.

Q6.9 (2 points) Assume that AS Y wants to implement a **cold-potato** policy, which means that, for a given AS path, AS Y aims to keep the traffic in its network for as long as possible. Identify the advertisement that RY will import when sending packets to H1.

AS Path	**MED**	**Source Router**	**Destination Router**

Solution:

AS Path	**MED**	**Source Router**	**Destination Router**
X- C - Y	10	R7	R9

Explanation: AS Y will import the advertisement with the shortest AS PATH (since all options come from providers with the same LOCAL PREF). To choose between the two shortest options, it then imports the route that will implement cold-potato routing (carry the traffic as fast as possible), which means exiting via R9 since it has the higher IGP cost.

Q7 ***Reliability*** **(10 points)**

Consider the following designs for providing reliable transport. We assume data packets are numbered consecutively. In each case, an ACK (or one or more NACKs, depending on the design) is sent by the receiver every time a packet arrives, and no ACKs (or NACKs) are sent other than at these times.

Q7.1 (3 points) Consider a modified version of TCP where there are no timers. The packet gets retransmitted after 2 dupACKs. Is this reliable? Justify your answer in 10 words or less.

○ Yes ● No

Solution: No, if multiple packets are lost (no dupacks) and the window is sent, nothing will happen

Q7.2 (3 points) Consider a design that aims to reduce the number of ACKs in the TCP protocol by using NACKs (negative ACKs) for packets that haven't been received. Whenever a data packet arrives, the receiver sends a NACK for every hole in its current window (i.e., a NACK for each packet it has not yet received that is earlier than its latest received packet). If the sender doesn't receive a NACK during a 10-second interval, the sender assumes all packets have arrived and move to the next W. Is this reliable? Justify your answer in 10 words or less.

○ Yes ● No

Solution: No, NACKs may be lost

Q7.3 (4 points) Consider a design that uses individual ACKs and a sliding window. The sender starts by sending the initial 10 packets. Whenever a data packet is received, the receiver always sends an ACK if the packet's sequence number is even and sends an ACK with 50% probability if the packet's sequence number is odd. Upon receiving an ACK, the sender randomly chooses a packet from the window to send (it chooses from all packets in the window, even those already sent). When there is a timeout, again, a random packet from the window is chosen to send. Is this reliable? Justify your answer in 10 words or less.

● Yes ○ No

Solution: Yes. The only way to move the window forward is if all packets before the window have been ACKed. This protocol only changes the order in which the packets in the window are sent (and may retransmit packets unnecessarily).

Q8 *TCP* **(10 points)**

Assume a TCP sender has an MSS=100B and a window size of 400B. There is no congestion control, so the window size is constant. The sender's highest **ACK received is for byte 1000**. Assume the timeout time is constant throughout the algorithm, and can be referenced as RTO. Unless stated otherwise, assume all packets experience an RTT of R. Assume packets are never fragmented or reordered and that packets are only lost if specified in the subpart.

We denote each packet as $p_{seq\#}$ and the time it is sent as $t_{seq\#}$. For example, the packet with sequence number 1000 is p_{1000} and is sent at t_{1000}.

Note that all subparts are independent of one another (changes made in one subpart do not affect the subsequent ones).

Q8.1 (1 points) Start from the moment after the sender has received and processed the highest ACK mentioned above. What packet is the current timer associated with?

Solution: p_{1000}

Explanation: p_{1000} is the beginning of the window.

Q8.2 (2 points) What is the highest ACK received at t_{1600}?

Solution: 1300

Explanation: Since p_{1600} has just been transmitted, that means it is the end of the window. Therefore, the beginning must start at 1700 - 400 = 1300.

Q8.3 (4 points) Assume that only p_{1400} is lost.

a) How does the sender detect its loss?

○ Timeout

● 3 dupACKs

○ The sender does not detect the loss

b) What is the highest ACKed byte after p_{1400} is successfully retransmitted and ACKed?

Solution: 1800

Explanation: Since only p_{1400} is lost, all other packets after it in the window will be sent (therefore it is detected via dupACKs). The sender will then retransmit p_{1400} and the cumulative ACK in response will ACK everything sent so far, so up to byte 1800 (1400 + 400).

Q8.4 (3 points) Assume all packets with sequence numbers **greater than or equal to 1400** are dropped. When does the timer expire?

Solution: $t_{1300} + R + RTO$ or $t_{1700} + RTO$

Explanation: The timer will expire RTO time after 1400 becomes the lowest unACKed byte (first packet in the window). This occurs when 1300 is ACKed ($t_{1300} + R$), or in other terms, when when p_{1700} is sent.

Q9 *Congestion Control* **(11 points)**

Suppose you are designing a new congestion control algorithm (CCA) and you want to experiment with its effectiveness and fairness.

You begin with CCA #1 which is described as follows:

- The initial cwnd is 1 MSS
- On receiving an ACK, add 1 MSS to the current cwnd
- On receiving 2 dupACKs, divide the cwnd by 2
- On a timeout, divide the cwnd by 2

Q9.1 (1 points) Is the CCA #1 algorithm fair?

○ Yes ● No

Q9.2 (3 points) At time `t`, when the `cwnd` size is `w`, the sender receives 2 dupACKs and `cwnd` size drops to `w/2`. Assuming no additional drops after that, how many packets must be ACKed to get back to the initial `cwnd` size, `w`, using CCA # 1? Your answer should be in terms of `w`.

Solution: w/2

Explanation: CCA #1 is a multiplicative increase, multiplicative decrease scheme. Accordingly, every CWND of ACKs, it will have doubled the CWND size. Therefore, to get back to `w` from `w/2`, it will take receiving `w/2` ACKs.

Assume that we now instead use CCA #2 with the following properties:

- Initial cwnd = 1
- On receiving an ACK, add $\frac{2MSS}{2(cwnd//2)}$ to the current cwnd (where $2(cwnd//2)$ gives the highest multiple of 2 less than cwnd – i.e., the // operator performs integer division).
- On receiving 3 dupACKs, divide the cwnd by 2
- On a RTO, divide the cwnd by 2

Q9.3 (1 points) Is the CCA #2 algorithm fair?

● Yes ○ No

Q9.4 (4 points) At time `t`, when the `cwnd` size is `w`, the sender receives 3 dupACKs and `cwnd` size drops to `w/2`. Assuming no additional drops after that, how many packets must be ACKed to get back to the initial `cwnd` size, `w`, using CCA # 2? Your answer should be in terms of `w`.

Solution:

$$\sum_{i=0}^{w/4-1} w/2 + 2i$$

Explanation: CCA #2 is an additive increase, multiplicative decrease scheme with an additive constant of 2. Accordingly, every CWND of ACKs, it will add 2 to the current CWND size. However, the CWND size is not constant as this increase occurs. Therefore, to get back to `w` from `w/2`, the sender will first receive `w/2` ACKs (to increase to `w/2` + 2), then `w/2` + 2 (to increase to `w/2` + 4), then `w/2` + 4, etc. until it gets `w/2` + `w/2` - 2 ACKs and reaches `w`. The summation above expresses this succinctly.

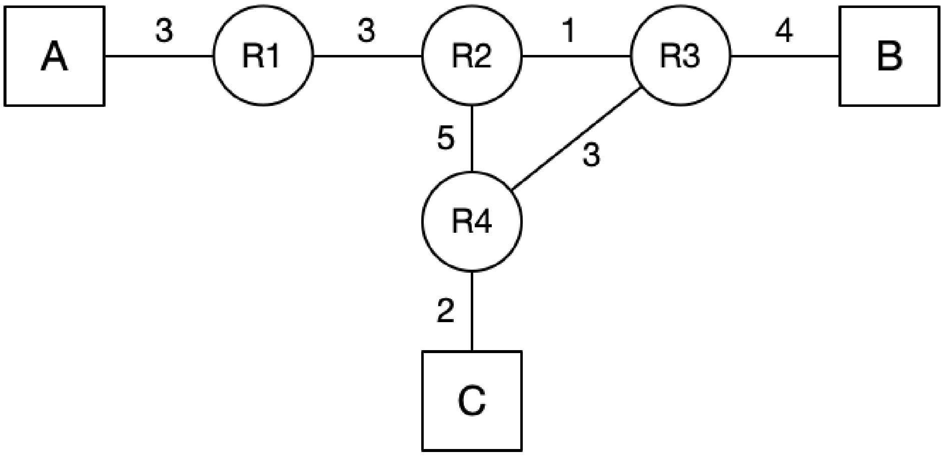
Q9.5 (2 points) Assume we run CCA #2 on a network which always reorders data packets such that for every 10 packets, the 5th packet arrives after the 9th (ie, the packets arrive in the order 1, 2, 3, 4, 6, 7, 8, 9, 5). ACKs will not be reordered by the network and will arrive in the order they are sent.

Will the average throughput be worse on this network than a network that does not have this behavior? Justify your answer in 10 words or less.

● Yes ○ No

Solution: This will cause the cwnd to cut in half every 10 packets due to dupACKs

Q10 ***DV*** **(15 points)**



Assume the following:

- The routers are running a distance vector algorithm with no poison reverse or route poisoning, though **split horizon is enabled.**
- The routing algorithm implements **incremental and triggered updates**, i.e. routers will advertise only the routes that have changed every time their table is updated.
- The routers send **periodic advertisements every 2 seconds** starting at t=0s.
- Routing table **entries expire after 6 seconds** of receiving no advertisements.
- Every link weight describes how long it takes for a packet to travel through in seconds. For example, a packet will go from R1 to R2 in 3 seconds. You can ignore all processing and queuing delays for the advertisements and packets.
- All advertisements are processed instantaneously.
- At time t = 0, all static routes are configured in the respective routing tables. Static routes never expire.

Q10.1 (2 points) Fill out the routing table of R4 after all routes converge.

Destination	**Next-Hop**	**Cost**
A		
B		
C	Direct	2

Solution:

Destination	**Next-Hop**	**Cost**
A	R3	10
B	R3	7
C	Direct	2

Q10.2 (3 points) How long will it take for R4's routing table to reach its converged state? At this point, not all routers might have converged yet.

Solution: 7 seconds

Explanation: Host A: R1 learns about the static route at time t=0, sends the an advertisement to R2. at t=3 R2 receives this advertisement and send it to R4 and R3. At time, t=4 R3 receives the route to A and at t=7 R4 receives it. Host B: R3 sends an advertisement to R4 at t=0. At t=3 R4 receives this. So 7 seconds total.

Q10.3 (4 points) Assume that at time t:

- All routers have converged
- **R2** renews the expiration time for **all** of its routes
- **All routers** send their periodic advertisements

Assume at time **t+1s, R1 crashes**. How many seconds after t will R2's routing table reflect this failure?

Solution: 9 seconds

Explanation: At time t, R1 sends a periodic advertisement to R2 and then crashes at time t + 1. This does not affect the advertisement already in transit, though. This advertisement reaches R2 at t + 3 and R2 updates its table accordingly. This entry will expire at t + 3 + 6 since R1 will not hear any more updates form R1 after t + 1.

Q10.4 (6 points) Regardless of your answer to the previous subpart, assume, R2's routing table reflected the failure of R1 at time `t'`. Assume `t'` is the time at which all routers send periodic advertisements as well. Fill out the routing table at R2 and R3 at time `t'+3`. All rows at the table should be filled. If a route entry is expired, mark it under the "Expired" column with "X". Ignore all **triggered** updates that might have happened prior to `t'`, but do consider any periodic updates that have happened (for example at time `t'- 2`).

Routing Table for R2

Dest.	Next-Hop	Cost	Expired
A			
B			
C			

Routing Table for R3

Dest.	Next-Hop	Cost	Expired
A			
B	Direct	4	
C			

Solution:

Routing Table for R2

Destination	Next-Hop	Cost	Expired
A	R4	15	
B	R3	5	
C	R3	6	

Routing Table for R3

Destination	Next-Hop	Cost	Expired
A	R2	16	
B	Direct	4	
C	R4	5	

Explanation: If there was a periodic advertisement at time t' this means there was a periodic advertisement at time t' - 4 and at t' - 2 as well. The question timeline is as follows: At t' - 4: there is a periodic advertisement from all routers (including R2 about the route to A since it didn't reflect the failure yet) At t'- 2: there is another periodic advertisement from all routers (including R2 about the route to A since it didn't reflect the failure yet) At t' R2's routing table reflects the change. And all routers send a periodic routing update. The fact that R2 needs to send a triggered update and the ordering doesn't matter here, since all advertisements send out instantly anyway (due to question assumptions) At time t' + 1, R4's advertisement for route to A with cost 10 (using R3) reaches R2. In a normal scenario, R2 would not accept this advertisement, but now that it's table entry expired, it'll accept it and will send a triggered update to R3 about this route with cost 15. At time t' + 2, R3 will receive R2's advertisement, since it's routing table still sees R2 as the nexthop, it will accept this advertisement, and update it's routing table to be 16. At time t'+3, R2 will receive another advertisement from R4 (this is the advertisement sent on t'-2) however, this advertisement will not have the updated value yet, so R2 will not update it's table yet.

This question is a modified version of one of the topologies of project 1 which shows that counting to infinity problem is still possible even if you implement split horizon.

Nothing on this page will affect your grade in any way.

Doodle

Congratulations for making it to the end of the exam! Feel free to leave any final thoughts, comments, feedback, or doodles here: