Problem Set #2

Due Thursday Oct 8th before the class

Problem 1 (6pts)

Explain collision domain and broadcast domain with respect to a hub, switch, and a router.

Generally broadcast domain is the area at which a message can be broadcast and collision domain is the are of a network where messages can collide.

In terms of hubs, switches, and routers a hub creates a broadcast domain for everything that connects to it but also creates a collision domain. This is because messages are simply passed everywhere once received. Switches also create a broadcast domain but limit the collision domain because it switches packets between hosts. The router can limit both broadcast and collision domain as it has the same collision protection as switches but also limits the scope of broadcast messages.

In addition to the textbook I also referenced this video when answering this question: www.youtube.com/watch?v=ck3gx9HB9-k

---

Problem 2 (5pts)

Consider the following networked computers connected by Bridge X and Y. Bridge X has interface 1,2 and 3. Bridge Y has interface 1 and 2. Assume at the beginning the address tables of Bridge X and Y are all empty. Write down the address tables of Bridge X and Y after the following communication finished.

1. A send a packet to C
2. B send a packet to D
3. C send a packet to E
4. E send a packet to A
5. D send a packet to A

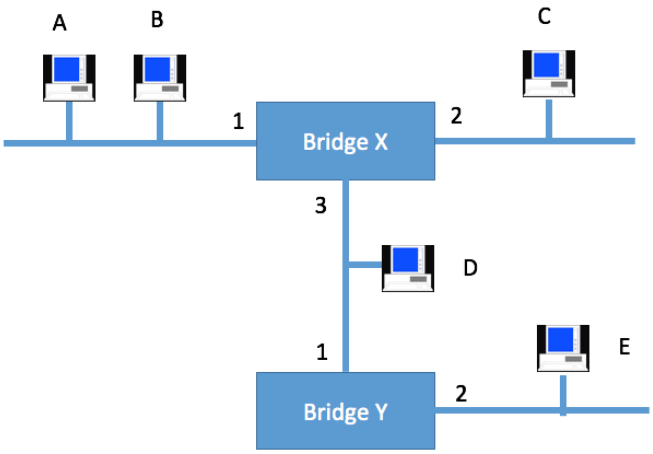


Figure 1

Bridge X

Address	Interface
A	1
B	1
C	2
E	3
D	3

Bridge Y

Address	Interface
A	1
B	1
C	1
E	2
D	1

This is because when the bridges don't have prior information on a destination they are going to broadcast the packet. So for example, sending a packet from A to C will entail bridge X receiving the packet from A on port 1. From here Bridge X can record which interface

A is located on and then it will forward the packet on ports 2 and 3 since the location of C is unknown. This means bridge Y also will receive the packet from A to forward to C and so it can record A as being connected to port 1. This process is repeated for all nodes (with the caveat of when locations are known the message is only sent where needed).

Problem 3 (5pts)

Given the extended LAN shown in Figure 2, indicate which ports are not selected by the spanning tree algorithm. Note that the bridge with the smallest ID becomes a root.

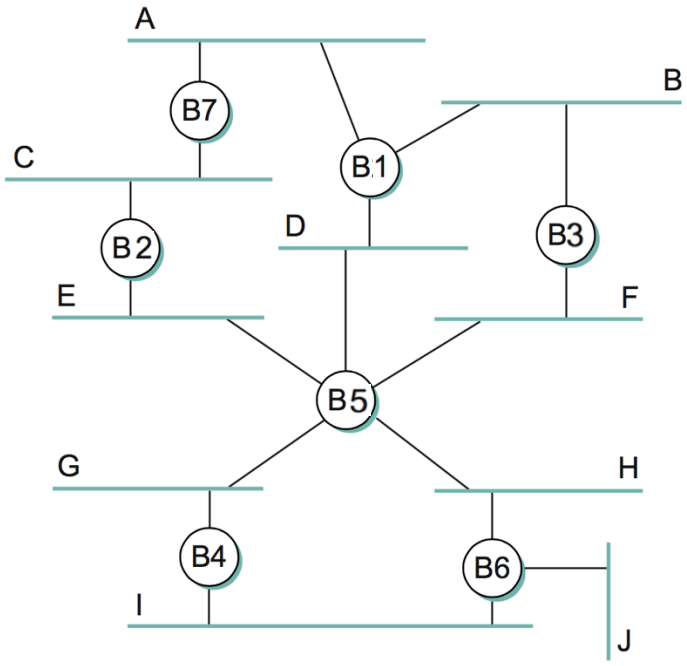


Figure 2

The spanning tree algorithm would be initiated with every node believing that it is the root and they will broadcast a message with the information (BK, 0, BK) (where K is the id of the bridge originating the message). After the first round of message passes, B1 would still believe itself to be the root and B3, B7, and B5 would also believe it to be the root (via paths B, A, and D respectively). These nodes would have a path length of one recorded as they are 1 hop from B1. B4 would still believe to be the root with B6 conjuring with that and B2 would believe itself to be the root.

On the next iteration both B4 and B6 would now believe B1 to be the root based on configuration messages sent via B5 (paths G and H). B2 would also believe that B1 is the root and it would select path E since B5 is a lower id than B7. That would leave all paths converted to believing B1 is root and now it would be the only one sending configuration messages.

In sum, **paths in the tree** -> A, B, D, E, G, H

Paths not selected be the tree -> C, F, I, J

Problem 4 (5pts)

> Still considering Figure 2. If Bridge B1 suffers catastrophic failure. Again indicate which ports are not selected by the spanning tree algorithm.

When B1 fails then configuration messages would cease to be sent throughout the tree. This would cause the algorithm to be run again how it was described in the previous step. This time B2 would become the root and all cases are resolved based on this fact.

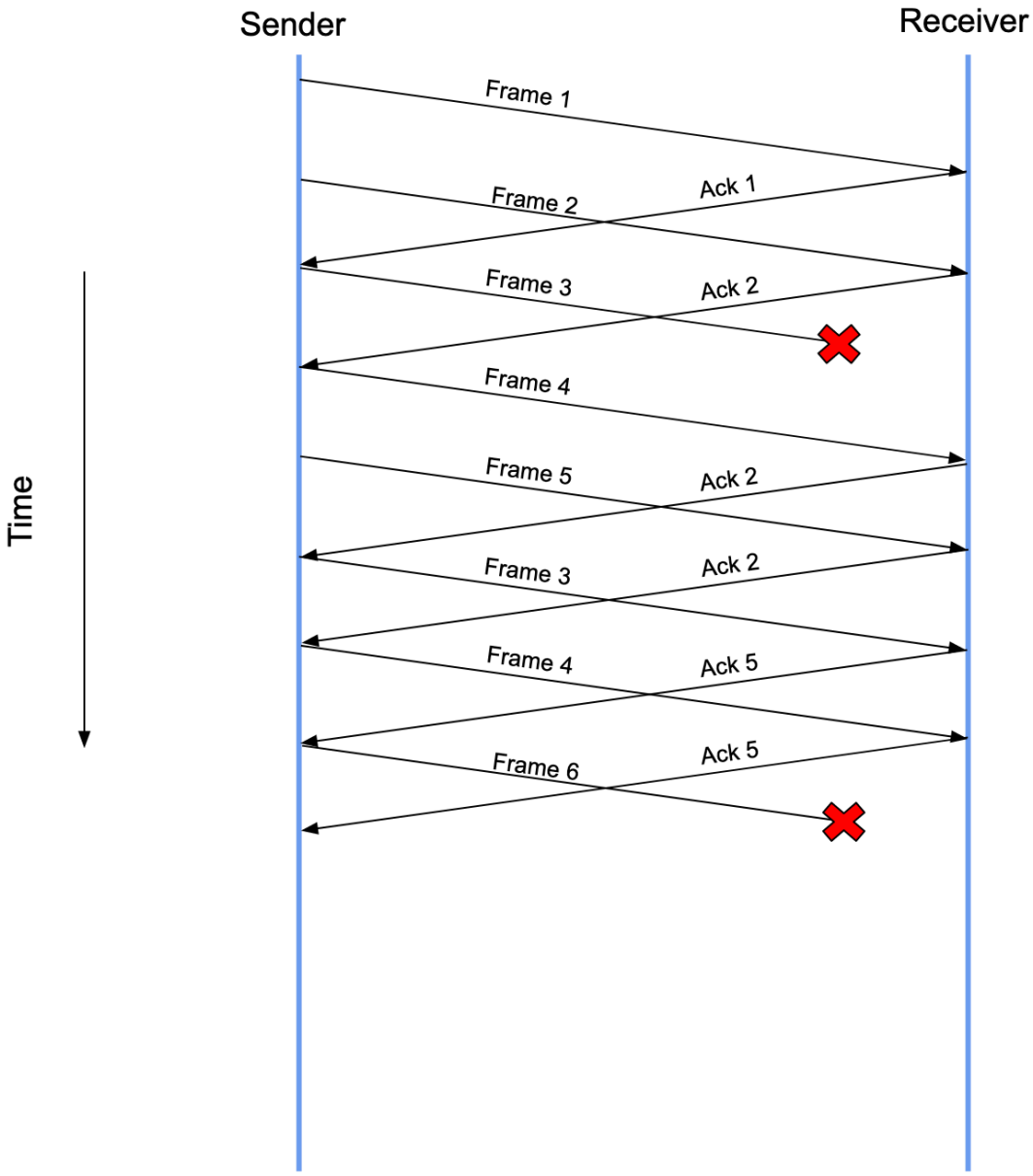
Paths in tree -> C, E, F, G, H

Paths not in tree -> A, B, D, I, J

Problem 5 (6pts)

Draw a time line diagram for the sliding window algorithm with SWS = RWS = 3 frames, for the following situations. Use a timeout interval of about 2 × RTT. And assume 2 frames must be send ½ RTT apart which means if everything is normal Sender will receive ACK and then immediately send the next frame.

Frames 3 and 6 are lost on their first transmissions. Draw the algorithm with time line diagram till Frame 6 is sent. (6pts)



First the algorithm will send frames 1, 2, and 3. Cumulative Acks 1 and 2 will be received and will advance the window to be range(3, 5). Frame 3 is lost and so no Ack will be received. However frames 4 and 5 are still going to be transmitted because this is the current window. When receive the ACK 2 from frame 4 we know that there is an issue. We must resend from the start of the window.

This will start by sending frame 3 and 4. Before frame 5 is sent we receive ACK 5 which is the result of receiving frame 3 and having stored frames 4 and 5. Now we advance the window to (6, 8) and begin sending frame 6.

Problem 6 (6pts)

Consider the GBN protocol with a sender window size of N=4 and a sequence number range of 1,024. Suppose that at time t, the next in-order packet that the receiver is expecting has a sequence number of k. Assume that the medium does not reorder messages. Answer the following questions:

(a) What are the possible sets of sequence numbers inside the sender's window

at time t? Justify your answer. (2 pts)

One set of sequence numbers is that which as sender and receiver perfectly aligned. The sender has received all ACKs sent by the receiver and as such the sender's window is from k to k + 3. (In reality this is really k mod 1024 and k + 3 mod 1024 since the k could exceed 1024).

The worst case is that the receiver sent ACKs k-4 to k-1 but they were never received. This would mean the sender is going to resend all 4 of these messages so the window is only k-4 to k-1. (Same applies about the mod).

Other sequences are possible because of lost ACKs. Any contiguous set of 4 values from k-4 to k+3 could also occur. (Mod).

(b) What are all possible values of the ACK field in all possible messages currently

propagating back to the sender at time t? Justify your answer. (2 pts)

The possible range in ACKs are from k - 4 to k - 1. (Note about mods applies). This is because the receiver isn't going to be acknowledging anything below the possible sender's window (because this would necessarily already have been acknowledged) and cannot be acknowledging the packet for which it is expecting (because this would imply receipt of the package).

(c) With the Go-Back-N protocol, is it possible for the sender to receive an ACK for a packet that falls outside of its current window? Justify your answer with an example. (2 pts)

This is possible if the timeout is too small. In a simple example with N = 2. The sender can send frames 1 and 2. These are received by the receiver who sends back an Ack 1 and ack 2. Before these are received the sender timeout and retransmits frames 1 and 2. When the receiver sees this it can resend ack 2 for each of these packets. At the same time the sender can receive the original Acks and advance its window and send frames 3 and 4. From here the new round of Acts would be sent to the sender with Ack = 2 which is less than the current window.

It would be impossible for the Ack that falls outside of the window to be above the current window.

Problem 7 (10 points)

(a) Is 10.72.0.255/255.255.254.0 a valid IP address for a host? **[2pts]**

255.255.254.0 is an invalid host IP address as the last byte cannot be 0 as it is not a valid host number.

10.72.0.255 is invalid because the 255 byte is reserved for broadcast.

(b) Divide the 10.72.0.0/16 subnets into five large networks of 8192 IPs each, 8 medium-sized networks of 2048 IPs each, and 10 small sized networks of 128 IPs each. **[6pts]**

The initial 10.72.0.0 subnet means that 10.72 is fixed. From here we know that 8192 = 2^13 so we need at least 13 bits for the host. This leaves 32 - 13 = 19 bits for the subnet mask. From this we can have 5 large networks as:

1. 10.72.128.0/19
2. 10.72.192.0/19
3. 10.72.224.0/19
4. 10.72.64.0/19
5. 10.72.32.0/19

For the medium sized networks we would need 2048 = 2^11 or 11 bits for host. The subnet mask would then be 32 - 11 bits = 21. Our medium networks could be:

1. 10.72.96.0/21
2. 10.72.104.0/21
3. 10.72.112.0/21

4. 10.72.120.0/21
5. 10.72.160.0/21
6. 10.72.168.0/21
7. 10.72.176.0/21
8. 10.72.184.0/21

Finally the small networks. 128 = 2^7 means 7 host bits needed. This amounts to a subnet mask = 32 - 7 = 25. They could be as follows:

1. 10.72.0.0/25
2. 10.72.0.128/25
3. 10.72.1.0/25
4. 10.72.1.128/25
5. 10.72.2.0/25
6. 10.72.2.128/25
7. 10.72.3.0/25
8. 10.72.3.128/25
9. 10.72.4.0/25

10.10.72.4.128/25

Altogether this still leaves 2^16 - 5*2^13 - 8*2^11 - 10*2^7 = 6912 unused addresses within the range. (One assumption I did make through this is that the address blocks were ok to include the invalid hosts).

(c) Is 192.168.2/23 and 192.168.3/23 representing the same subnet? Please justify your answer. **[2pts]**

192.168.2 in binary is **11000000.10101000.0000000**1**0**
192.168.3 in binary is **11000000.10101000.0000000**1**1**

The /23 means the subnet includes the 23 most significant bits or in other words the bolded versions of the two above. These portions match and as such these addresses are within the same subnet.

Problem 8 (8 points)

An organization has been assigned the prefix 192.168.1.0/23 and wants to form subnets for 4 departments which have the following number of hosts:

Department A:	130 hosts
Department B:	120 hosts
Department C:	60 hosts

Department D: 31 hosts

(a) Give a possible arrangement of subnet masks to make this possible. **[5pts]**

First, from the work in 7c I know that the 1 in 192.168.1.0 is a part of the host portion of the ip.

Now I know that Department A will need at least 8 bits (256 addresses) as this is large enough to hold 130 hosts. Department B will need at least 7 bits (128 addresses) as this is large enough to hold 120 hosts. Department C will need at least 6 bits (64 addresses) as this is large enough to hold 60 hosts. Finally, Department D will need at least 5 bits (32 addresses) as this is large enough to hold 31 hosts.

Allocating A first I know that the total subnet mask will be 32 - 8 bits and thus 24. I will assign department A 192.168.0.0/24.
Using the same process here I know department B can get 192.168.1.0/25, department C can get 192.168.1.128/26, and department D can get 192.168.1.192/27.

(b) Suggest what the organization might do if department C grows to 65 hosts. **[3pts]**

They could do one of several things. The worst solution would probably be to put D, C, and D in the same group so that they could then have two 256 address subnets. This would have issues with multiple departments receiving each others broadcasts which wouldn't be the best. They could allocate a smaller subnet to encompass the overflow of the C department and try to have the switches virtually simulate a complete network. Or they could try using NAT within the current subnet.

Problem 9 (12 points)

For the network given below in Figure 3, give global distance-vector tables for each node when**:**

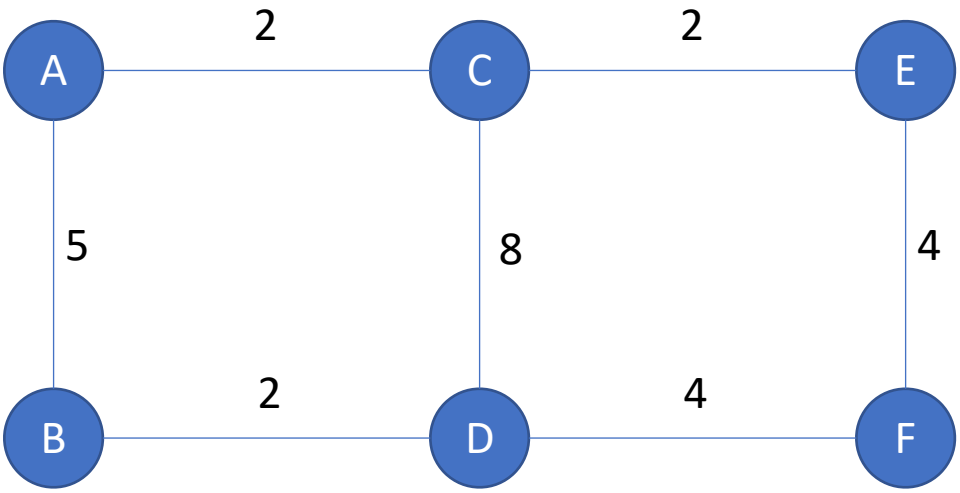


Figure 3

(a) Each node knows only the distance of its immediate neighbors. **[4pts]**

	A	B	C	D	E	F
A	0	5	2	inf	inf	inf
B	5	0	inf	2	inf	inf
C	2	inf	0	8	2	inf
D	inf	2	8	0	inf	4
E	inf	inf	2	inf	0	4
F	inf	inf	inf	4	4	0

(b) Each node has reported the information it had in the first step (a) to its immediate neighbors. **[4pts]**

	A	B	C	D	E	F
A	0	5	2	7	4	inf
B	5	0	7	2	inf	6
C	2	7	0	8	2	6
D	7	2	8	0	8	4
E	4	inf	2	8	0	4
F	inf	6	6	4	4	0

(c) Repeat step (b) one more time. **[4pts]**

	A	B	C	D	E	F
A	0	5	2	7	4	8
B	5	0	7	2	9	6
C	2	7	0	8	2	6
D	7	2	8	0	8	4
E	4	9	2	8	0	4
F	8	6	6	4	4	0

Problem 10 (8 points)

Again for the network graph in Figure. 3. Show how the link-state algorithm builds the routing table for node D.

(a) Show the detailed steps with the link-state algorithm. **[5pts]**

The first step is flooding the network with LSPs. For example, node A would send out the packet (ID = A, [(B , 5), (C, 2)], 0, TTL). The selection of sequence number as 0 is because I'm assuming this is the first time the routing table is being built so I chose the lowest sequence number. TTL would be replaced by a reasonable time to live. All other nodes would also be sending LSP packets with their own pertinent information.

With LSPs generated the next step is sending them to neighbors. For node A these packets originally go to its neighbors, B and C, who then forward them on to their neighbors (excluding A since they received the packet from A). This dissemination process is repeated for all LSPs until the entire graph information has been distributed.

Now each node has all the information it needs to construct the graph and run Dijkstra's algorithm. I will again use A as the example. It would start the algorithm

with only A on the Q and the cost for each node as infinity (except for A, the root). The order it would pick elements are confirmed (and the cost) are C (2), E(4), B(5), D(7), and then F(8).

This now represents the distance to all nodes from the source A.

(b) Show the final routing table of node D. **[3pts]**

	A	B	C	D	E	F
A	0	5	2	7	4	8
B	5	0	7	2	9	6
C	2	7	0	8	2	6
D	7	2	8	0	8	4
E	4	9	2	8	0	4
F	8	6	6	4	4	0

Problem 11 (6 points)

The network graph is shown in Figure. 4.

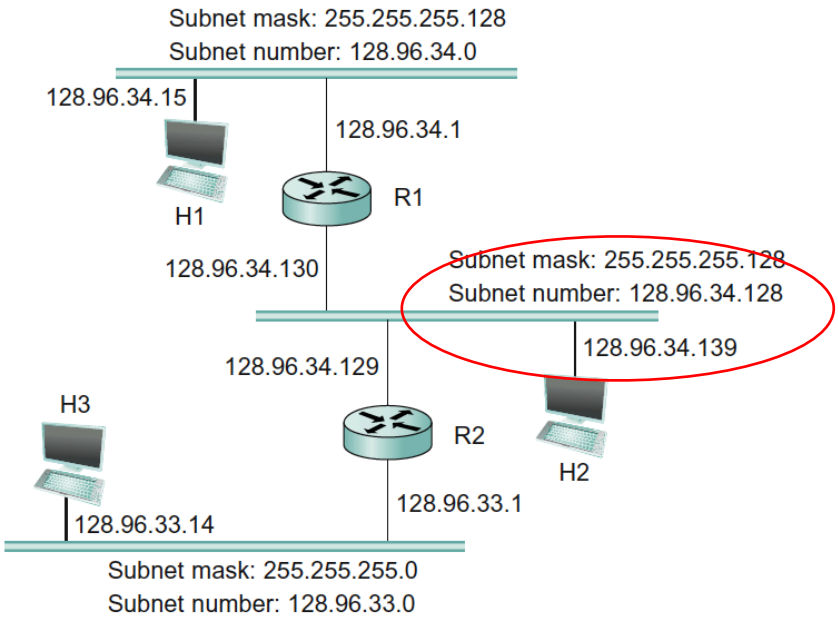


Figure 4

(a) Host H1 sends a packet to the destination 128.96.34.126. Explain how this packet traverses in the network described below. You need to describe who received the packet and what are their reactions. **[2pts]**

First H1 ensures that the destination is not within it's own subnet. 128.96.34.126 AND 255.255.255.128 is 128.96.34.0 and since this matches the subnet number of H1 it doesn't need to forward the packet to the router and can send the packet within the subnet.

(b) Host H3 sends a packet to the destination 128.96.34.250. Explain how this packet traverses in the network. **[2pts]**

H3 starts by ensuring the recipient isn't within its own subnet. 128.96.34.250 AND 255.255.255.0 is 128.96.34.0 which is not the subnet number of H3. It would then pass the message to the router.

The router then begins comparing the address with subnet masks in its forwarding table to see if it will match the subnet number associated with the mask. In the event that it does the router will forward it according to the information given. If there is no match it will forward based on the default route. 128.96.34.250 AND 255.255.255.128 does match the subnet number 128.34.128 and as such the packet will be forwarded within the adjacent subnet.

(c) The subnet of H1 has now two different teams and would like to split it into two subnets. Please add one more subnet and add R3 and change the network configurations as you need. Note that you are allowed to modify the network as least disruptive as possible. **[2pts]**

For H1 you could split the node in to two subnets with mask 255.255.255.192 and subnets numbers of 128.96.34.64 and 128.96.34.0 these each could link up to a router R3 that maintains the original subnet mask and number and will then be able to receive packets normally and distribute them to the proper subnet from there. The configuration would now be something like:

```
                    ————————————————————————-
                    |                                   |
H1 Group 1 ————R3 ———————— H1 Group 2              R1
```

Where everything beyond R1 remains unchanged.

Problem 12 (8 points)

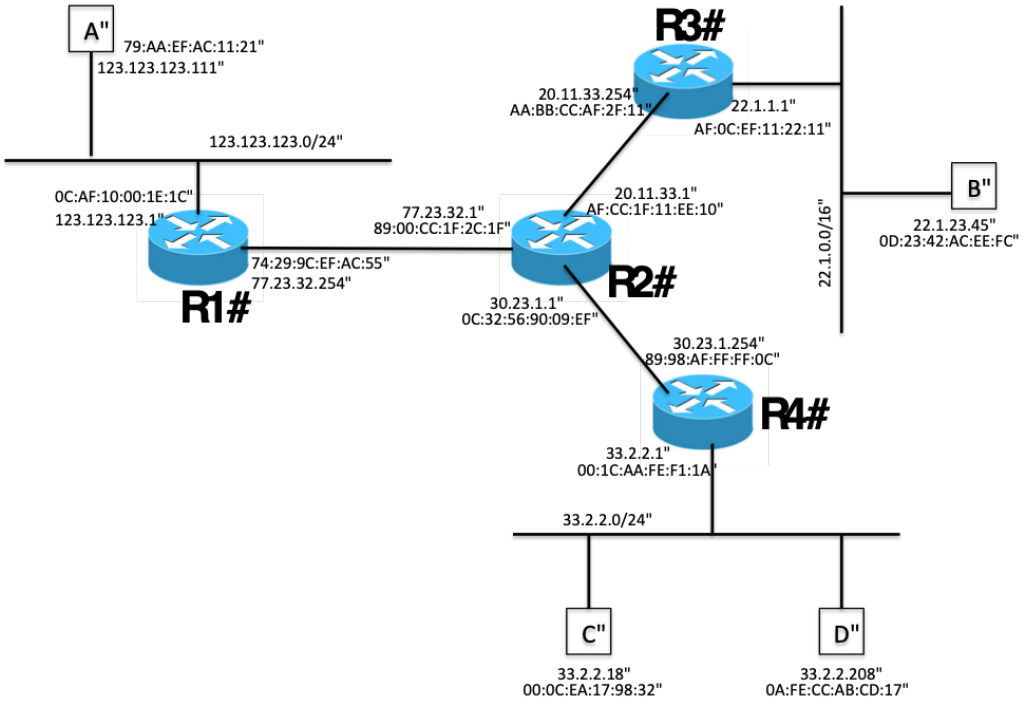


Figure. 5

Above in Figure 5 is the network graph with 4 routers (R1, R2, R3, R4) and 4 hosts (A, B, C, D). Each router interfaces and hosts are labeled with both IP and MAC address,

Routing is enabled so that any two hosts can communicate with each other and also the default gateway of each host is set to its gateway router.

(a) Suppose that B send an IP packet to C through R3, R2, R4. Write down the IP packet's content (src MAC, dst MAC, src IP, dst IP) along the path in the Table given below: **[4pts]**

	src MAC	dst MAC	src IP	dst IP
B -> R3	0D:23:42:AC:EE:FC	AF:0C:EF:11:22:11	22.1.23.45	32.2.2.18
R3 -> R2	AA:BB:CC:AF:2F:11	AF:CC:1F:11:EE:10	22.1.23.45	32.2.2.18
R2 -> R4	0C:32:56:90:09:EF	89:98:AF:FF:FF:0C	22.1.23.45	32.2.2.18
R4 -> C	00:1C:AA:FE:F1:1A	00:0C:EA:17:98:32	22.1.23.45	32.2.2.18

Table. 1

(b) When A sends out an ARP query for its default gateway, what is the reply to that query? **[2pts]**

The default gateway for A is R1 so the arp query would be 0C:AF:10:00:1E:1C.

(c) Suppose the routers use link-state routing protocol, what will be R3's routing table entries? **[2pts]**

Subnet	Address
123.123.123.0/24	AF:CC:1F:11:EE:10
32.2.2.0/24	AF:CC:1F:11:EE:10
22.1.0.0/16	0D:23:42:AC:EE:FC

This represents the destination subnet and the MAC address to forward the information to.

Problem 13 (6 points)

Suppose a computer just boot up, connected to wireless network and successfully obtained IP, gateway and DNS address. Now it wants to access www.yahoo.com from its browser. Describe the sequence of packets exchanged to and from this computer until the webpage starts to load. (include what kind of protocol is used and what is the content of the packets)

The first thing the computer will do is send a DNS request for www.yahoo.com. This will be met with a response containing the IP address of yahoo. These requests are done via udp.

To connect to yahoo the computer is likely going to use https (port 443) which uses tcp. The first thing that will need to be done is establishing a connection with the server via tcp handshake. The first packet sent will be to yahoo's server. It will have the host's ip and a client port and will be directed to yahoo's ip obtained from the DNS with the port 443 attached to it. This packet will be a SYN packet with some sequence number, x.

Yahoo can then respond with a SYNACK with another sequence number, y and an ACK x+1. The handshake is completed when the host ACKs the SYNACK from yahoo with content y+1.

From here a connection has been established and the two can now communicate via tcp and exchange the web content. As this happens and the hosts begins to receive packets the web content will begin loading.

Problem 14 (9 points)

Consider the simple network in Figure 5 below. X, Y and Z are routers and their link costs are as specified. Assume the network uses a Distance Vector algorithm is used. Y's and Z's routing tables are look like Table 2.

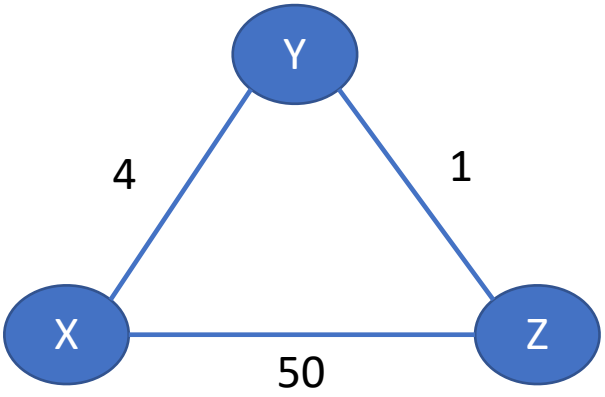


Figure. 5

Node Y/Distance	Via X	Via Z
X	4	6

Node Z/Distance	Via X	Via Y
X	50	5

Table. 2

(a) Now Let assume the cost of link X-Y suddenly changed to 100. Please write down the Y's and Z's routing table regarding distance to X, after Y updates this information to Z and then Z updates its information back. **[3pts]**

Node Y/Distance	Via X	Via Z
X	100	51

Node Z/Distance	Via X	Via Y
X	50	101

(b) Please write down the Y's and Z's routing table regarding X after Y updates this information to Z again and then Z updates back again. **[3pts]**

Node Y/Distance	Via X	Via Z
X	100	51

Node Z/Distance	Via X	Via Y
X	50	52

(c) How many updates did Y get until its distance to X have converged with Distance Vector algorithm? **[3pts]**

Node Y only needed the original update from X about the increase in path cost and then the response from Z to converge.