CMPT 371 SOLUTION ASSIGNMENT 4 (see assignment page for due date)

NOT ALL PROBLEMS WILL BE GRADED, A TOTAL OF 100 POINTS WORTH OF PROBLEMS CHOSEN FROM THE FOLLOWING PROBLEMS WILL BE GRADED. YOU WILL RECEIVE SOLUTIONS OF ALL PROBLEMS.
You must complete problems 1, 2 and 7. You should choose 40 points of questions (or parts of questions) from the remaining questions.

1. Consider the CIDR routing table shown below. This is the routing table for router A.
   a) **[10 points]** One of the network addresses in the routing table is not a legal network address for a network with the stated mask. Which network address is incorrect? Why? To correct the address choose the legal address closest to the incorrect address.
   b) **[10 points]** Is the following forwarding table optimized so that the first match found is the "best" match? Explain why or why not. If the table is not optimized to assure that the first match found is the "best" match rewrite the table in an order that does assure the first match found is the "best".
   c) **[10 points]** Draw a diagram illustrating the networks and routers (router A and gateway routers) described in the routing table. Label each network with the network address in the form 12.12.12.13/12. Show which networks each router is connected to. Label the Ethernet interfaces on router A. Label the correct interface on each gateway router with the gateway address.
   d) **[10 points]** For the address 10.15.53.70, explain in detail, in order, the actual steps taken to determine where to send the packet based on the forwarding table below. Show steps and all calculations used to determine where to send the packet, In particular show the calculations used to combine the mask in each row and the destination IP address to calculate the address to compare to the destination network address in the same row. For the purpose of this calculation you may assume the first matching entry you find in the routing table is the correct entry. Use the unordered routing table given below NOT the ordered version you built earlier in part b) of this problem.
   e) **[20 points]** For IP packets with destination addresses 10.15.53.88, 10.45.45.45, 10.15.64.23, 10.5.35.88 and 10.15.53.128. Line in the routing table below is used, which interface is the packet sent through, and what the IP address of the host the packet will be sent to in the Ethernet layer.

Destination network address	Gateway address	mask	Interface	
10.5.32.0	*	255.255.224.0	eth3	Line 1
10.15.52.0	10.5.35.69	255.255.254.0	eth1	Line 2
10.16.191.0	10.5.32.32	255.255.252.0	eth3	Line 3
10.40.0.0	*	255.248.0.0	eth2	Line 4
10.15.48.0	10.40.18.12	255.255.240.0	eth2	Line 5
10.5.35.64	*	255.255.255.192	eth1	Line 6

SOLUTION

a) **[10 points]** One of the network addresses in the routing table is not a legal network address for a network with the stated mask. Which network address is incorrect? Why? Correct the address, choose the legal address closest to the incorrect address

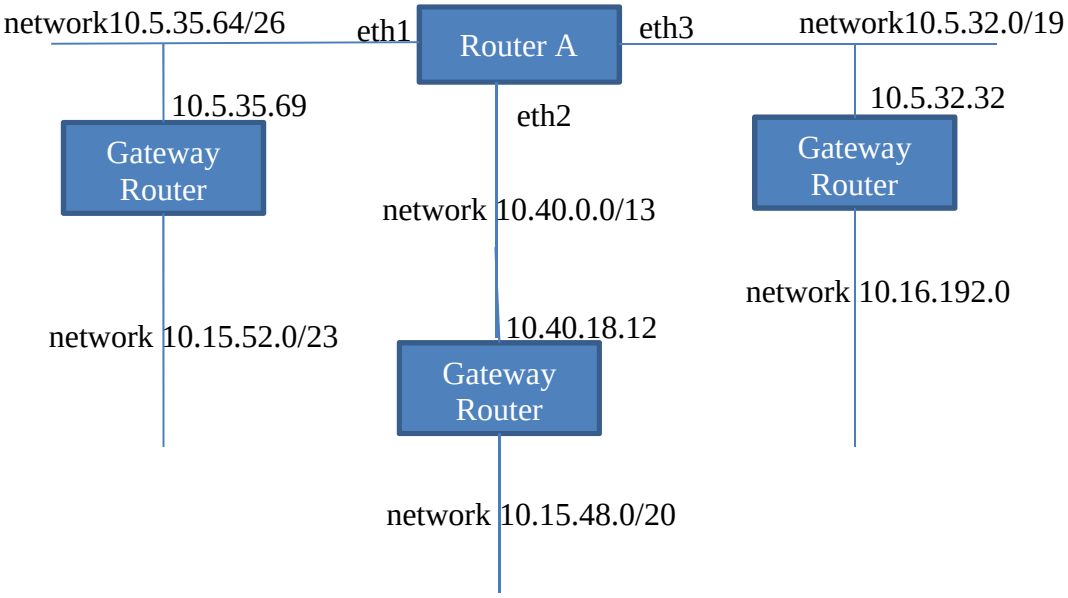
The network address that is not a legal subnet address for the given mask is 10.16.191.0. The mask for this address is 255.255.252.0. In binary 252 is 11111100 which means there are $2^{(32-22)}= 2^{10}= 2^2*256$ addresses. This means that A valid CIDR network with this mask needs to start on a multiple of 4 (2^2) in the second group from the right of the four groups in the dotted decimal address. 191 is not a multiple of 4, so the network address is not valid. The closest valid address (multiple of 4) is 192. So the network address should be 10.16.192.0

b) **[10 points]** Is the following forwarding table optimized so that the first match found is the "best" match? Explain why or why not. If the table is not optimized to assure that the first match found is the "best" match rewrite the table in an order that does assure the first match found is the "best".

The routing table is not optimized so that the first matching entry found will always be the "best" entry. For any address in the network 10.5.35.64, the address will be part of network 10.5.32.0 as well. To find the "best" fit need to choose the longest match, which is network 10.5.35.64, (it has the longer netmask) which is currently after 10.5.32.0. To assure the "best" fit in all cases the networks should be ordered from longest netmask to shortest netmask

Destination network address	Gateway address	mask	Interface
10.5.35.64	*	255.255.255.192	eth1
10.15.52.0	10.5.35.69	255.255.254.0	eth1
10.16.192.0	10.5.32.32	255.255.252.0	eth3
10.15.48.0	10.40.18.12	255.255.240.0	eth2
10.5.32.0	*	255.255.224.0	eth3
10.40.0.0	*	255.248.0.0	eth2

c) **[10 points]** Draw a diagram illustrating the networks and routers (router A and gateway routers) described in the routing table. Label each network with the network address in the form 12.12.12.13/12. Show which networks each router is connected to. Label the Ethernet interfaces on router A. Label the correct interface on each gateway router with the gateway address.



d) [10 points] For the address 10.15.53.70, explain in detail, in order, the actual steps taken to determine where to send the packet based on the forwarding table below. Show steps and all calculations used to determine where to send the packet, In particular show the calculations used to combine the mask in each row and the destination IP address to calculate the address to compare to the destination network address in the same row. For the purpose of this calculation you may assume the first matching entry you find in the routing table is the correct entry. Use the unordered routing table given below NOT the ordered version you built earlier in part b) of this problem.

For the *address Y*=10.15.53.70 *the following steps will be taken*

- ***Y is ANDed with the mask of the first entry***

  10.15.53.70 AND 255.255.255.192 ***gives 10.15.53.64***

  ```
  00001010  00000111  00110101  01000110
  11111111  11111111  11111111  11000000
  00001010  00000111  00110101  01000000
  ```

  10.15.53.64 does not match the destination network address 10.5.35.64 so consider the next entry

- ***Y is ANDed with the mask of the third entry***

  10.15.53.70 AND 255.255.224.0 gives 10.15.32.0

  ```
  00001010  00000111  00110101  01000110
  11111111  11111110  11100000  00000000
  11101011  00100000  00100000  0000000010.15.52.0
  ```

  10.15.32.0 does not match the destination network address 10.5.32.0 so consider the next entry

- ***Y is ANDed with the mask of the third entry***

10.15.53.70 AND 255.255.254.0 gives 10.15.52.0

```
00001010   00000111   00110101   01000110
11111111   11111111   11111110   00000000
--------------------------------------------
00001010   00000111   00110100   00000000
```

10.15.52.0 does match the destination network address 10.15.52.0

b) **[20 points]** For IP packets with destination addresses 10.15.53.88, 10.45.45.45, 10.15.64.23, 10.5.35.88 and 10.15.53.128. Line in the routing table below is used, which interface is the packet sent through, and what the IP address of the host the packet will be sent to in the Ethernet layer.

10.15.53.88	***line 2***	***eth1***	***10.5.35.69***
10.45.45.45	***line 4***	***eth2***	***10.45.45.45***
10.15.64.23	***No line***	***------***	***packet dropped***
10.5.35.88	***Line 6***	***eth1***	***10.5.35.88***
10.15.53.128	***line 2***	***eth1***	***10.5.35.69***

2. [30 points] Consider the distributed Bellman-Ford algorithm used in the first generation internet. At station A, new routing tables have just arrived from A's nearest neighbors B, C, D, and E. The cost from A to B is 2, for A to C is 5, for A to D is 3, and for A to E is 7. These newly received routing tables are given below. Show your work. Based on these newly received routing tables, calculate a new routing table for node A. Assume all the distance vectors are processed at the same time.

	from B		*from C*		*from D*		*from E*	
	Cost	***Next***	***Cost***	***Next***	***Cost***	***Next***	**Cost**	**Next**
A	***2***	***A***	***5***	***A***	***3***	***A***	**7**	**A**
B	***0***	***-***	***1***	***B***	***5***	***A***	**9**	**A**
C	***1***	***C***	***0***	***-***	***5***	***G***	**8**	**J**
D	***5***	***A***	***5***	***G***	***0***	***-***	**6**	**J**
E	***9***	***A***	***10***	***B***	***6***	***J***	**0**	**-**
F	***2***	***C***	***1***	***F***	***6***	***G***	**11**	**A**
G	***4***	***C***	***3***	***G***	***2***	***G***	**8**	**J**
H	***3***	***H***	***2***	***H***	***7***	***G***	**12**	**A**
J	***7***	***A***	***7***	***G***	***2***	***J***	**4**	**J**
K	***8***	***A***	***8***	***G***	***3***	***J***	**5**	**J**
L	***4***	***L***	***5***	***B***	***9***	***A***	**7**	**J**
M	***5***	***C***	***4***	***M***	***9***	***G***	**14**	**A**
N	***4***	***C***	***3***	***F***	***8***	***G***	**13**	**A**

Solution

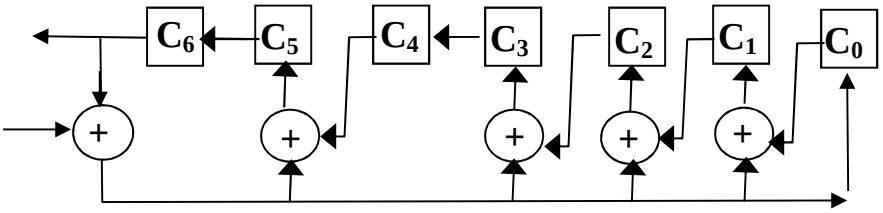
	B	***C***	***D***	***E***	*New table*	
	Cost	***Cost***	***Cost***	**Cost**	***Cost***	***Next***
A	***4***	***10***	***6***	**14**	***-***	***-***
B	***2***	***6***	***8***	**16**	***2***	***B***
C	***3***	***5***	***8***	**15**	***3***	***B***
D	***7***	***10***	***3***	**13**	***3***	***D***
E	***11***	***15***	***9***	**7**	***7***	***E***
F	***4***	***6***	***9***	**18**	***4***	***B***
G	***6***	***8***	***5***	**15**	***5***	***D***
H	***5***	***7***	***10***	**19**	***5***	***B***
J	***9***	***12***	***5***	**11**	***5***	***D***
K	***10***	***13***	***6***	**12**	***6***	***D***
L	***6***	***10***	***12***	**14**	***6***	***B***
M	***7***	***9***	***12***	**21**	***7***	***B***
N	***6***	***8***	***11***	**20**	***6***	***B***

3. A CRC is constructed to generate a 7 bit Frame Check sequence for a 18 bit message. The generator polynomial is $P(X) = X^7 + X^5 + X^3 + X^2 + X + 1$. The message bits for a particular message are
   1 1 0 0 0 1 1 0 1 0 0 1 1 1 0 0 0 1
   a) [10 points] Write the message polynomial *M*(*X*)

$$M(X) = X^{17} + X^{16} + X^{12} + X^{11} + X^9 + X^6 + X^5 + X^6 + X^5 + X^4 + 1$$

   b) [10 points] Draw a shift register circuit to perform the calculation of the CRC bits.

$$P(X) = X^7 + X^5 + X^3 + X^2 + X + 1$$



   c) (20 points) Determine the FCS using modulo 2 divisions. Show your work.

```
                                1 1 1 1 0 0 0 0 0 1 1 0 0 1 0 1 1
                  ┌──────────────────────────────────────────────────
1 0 1 0 1 1 1 1 √ │ 1 1 0 0 0 1 1 0 1 0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0
                  │ 1 0 1 0 1 1 1 1
                  │   1 1 0 1 0 0 1 1
                  │   1 0 1 0 1 1 1 1
                  │     1 1 1 1 1 0 0 0
                  │     1 0 1 0 1 1 1 1
                  │       1 0 1 0 1 1 1 0
                  │       1 0 1 0 1 1 1 1
                  │             1 1 1 1 0 0 0 1
                  │             1 0 1 0 1 1 1 1
                  │               1 0 1 1 1 1 0 0
                  │               1 0 1 0 1 1 1 1
                  │                     1 0 0 1 1 0 0 0
                  │                     1 0 1 0 1 1 1 1
                  │                       1 1 0 1 1 1 0 0
                  │                       1 0 1 0 1 1 1 1
                  │                         1 1 1 0 0 1 1 0
                  │                         1 0 1 0 1 1 1 1
                  │                           1 0 0 1 0 0 1
```

d) (20 points) Determine the FCS using the shift register circuit. Show the content of the shift register after each bit of the message has been shifted into the shift register circuit.

```
0 0 0 0 0 0 0                                   1 1 0 0 0 1 1 0 1 0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0
1 1 0 0 0 1 1                                   0 1 0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0
1 0 0 0 1 1 0                                   1 1 0 1 0 0 1                                   1 0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0
1 0 1 0 0 1 1                                   1 1 1 1 1 0 0                                   0 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0
1 1 1 1 0 0 0                                   1 0 1 0 1 1 1                                   0 1 1 1 0 0 0 1 0 0 0 0 0 0 0
0 1 0 1 1 1 0                                   0 0 0 0 0 0 1                                   1 1 1 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 1 1                                   1 1 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 1 1 1                                   1 0 0 0 1 0 0 0 0 0 0 0
0 0 0 1 1 1 1                                   0 0 0 1 0 0 0 0 0 0 0
0 0 1 1 1 1 0                                   0 0 1 0 0 0 0 0 0 0
0 1 1 1 1 0 0                                   0 1 0 0 0 0 0 0 0
1 1 1 1 0 0 0                                   1 0 0 0 0 0 0 0
1 1 1 0 0 0 1                                   0 0 0 0 0 0 0
1 0 1 1 1 1 0                                   0 0 0 0 0 0
0 1 1 1 1 0 0                                   0 0 0 0
0 0 1 0 0 1 1                                   0 1 1 0 1 1 1                                   0 0 0
0 1 0 0 1 1 0                                   1 1 0 1 1 1 0                                   0 0
1 0 0 1 1 0 0                                   1 0 1 1 1 0 0                                   1 1 1 0 0 1 1                                   0
0 0 1 1 0 0 0                                   1 1 0 0 1 1 0
0 1 1 0 1 1 1                                   1 0 0 1 0 0 1
1 1 0 1 1 1 0
1 0 1 1 1 0 0
1 1 1 0 0 1 1
1 1 0 0 1 1 0
1 0 0 1 0 0 1

        1  0 0 1 0 0 1
```

e) [10 points] Can the errors represented by each of the following error polynomials $E(X)$ be detected by the CRC? Why or why not?

i. 1010001000010010000

The error will be detected because the CRC detects all odd numbers of single bit errors; in this case there are 5 single bit errors

ii. 0000000011011000000

This is a burst error of length 5, For this CRC there are 7 FCS bits so any burst error with length less than or equal to 7 will be detected. This error will be detected

iii. 0001101011110100000

```
                1 1 1 0 0 1
          ┌──────────────────
1 0 1 0 1 1 1 1 │ 1 1 0 1 0 1 1 1 0 1 0 0 0 0 0
                  1 0 1 0 1 1 1 1
                  ---------------
                    1 1 1 1 0 0 0 0
                    1 0 1 0 1 1 1 1
                    ---------------
                      1 0 1 1 1 1 1 1
                      1 0 1 0 1 1 1 1
                      ---------------
                            1 0 0 0 0 0 0 0
                            1 0 1 0 1 1 1 1
                            ---------------
                                1 0 1 1 1 1 0 0
                                1 0 1 0 1 1 1 1
                                ---------------
                                      1 0 0 1 1
```

Will be detected, does not divide exactly by P(X)

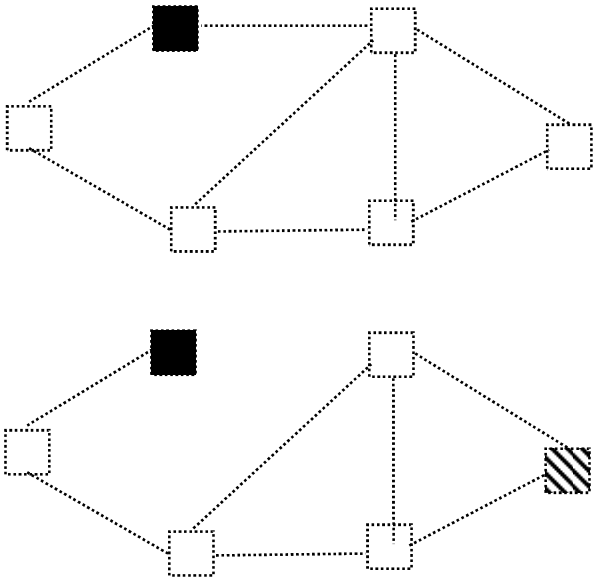
4. **[30 points]** Consider a system using flooding with a hop counter. Suppose that the hop counter is originally set to the diameter of the network. When the hop count reaches zero, the packet is discarded except at its destination. Does this always insure that a packet will reach its destination if the case that there exists at least one operable path? Why or why not? Give an example or counter example.

The diameter of the network is defined as the length of the longest of the minimum cost paths through the network (The length of the minimum cost path is one more than the number of intermediate stations along that path). The hop count is set to the diameter of the network. As the packet moves through the network, the hop count is decremented each time the packet is retransmitted.

Given that all links in the network of diameter N are in service then the station which is connected to the source by the longest minimum cost path is reached by the packet the hop count will be zero. All stations will be reached on or before the hop for which the hop count is decremented to zero.

However, if one of the intermediate links fails there may be stations that will not be reached before the hop count is decremented to zero. Since the packet is not forwarded when the hop count is zero, the packet will not reach these stations. The failure of the link must increase the length of the longest minimum cost path through the network to cause this to happen.

An example is shown below. The solid box indicates a source station. This longest minimum cost path (assuming unit costs) is of length 2. If a link fails, resulting in the net shown in the lower figure, the hatched node has a minimum cost path of length 3. A packet with a hop count of 2 will never reach this station. There is still an operable path to the hatched station, but the length of the minimum cost path exceeds the maximum hop count (the maximum



5. Consider the following AS. This AS is running OSPF as its internal routing protocol and BGP as an external routing protocol. Routers are indicated by cubes and networks internal to the autonomous system by clouds. Networks external to the autonomous system are indicated by a network labels N8, N9 and N10. Weights of paths leaving each router are given adjacent to that router. The AS is divided into the areas illustrated by the dotted blue parallelograms.

a) **[5 points]** Which routers in the autonomous system must support both OSPF and BGP? Briefly explain why

AS border routers run OSPF to manage routing inside the AS and BGP (an external gateway protocol) to manage routing between ASs. The AS border routers for this AS are R5, R6 and R4.

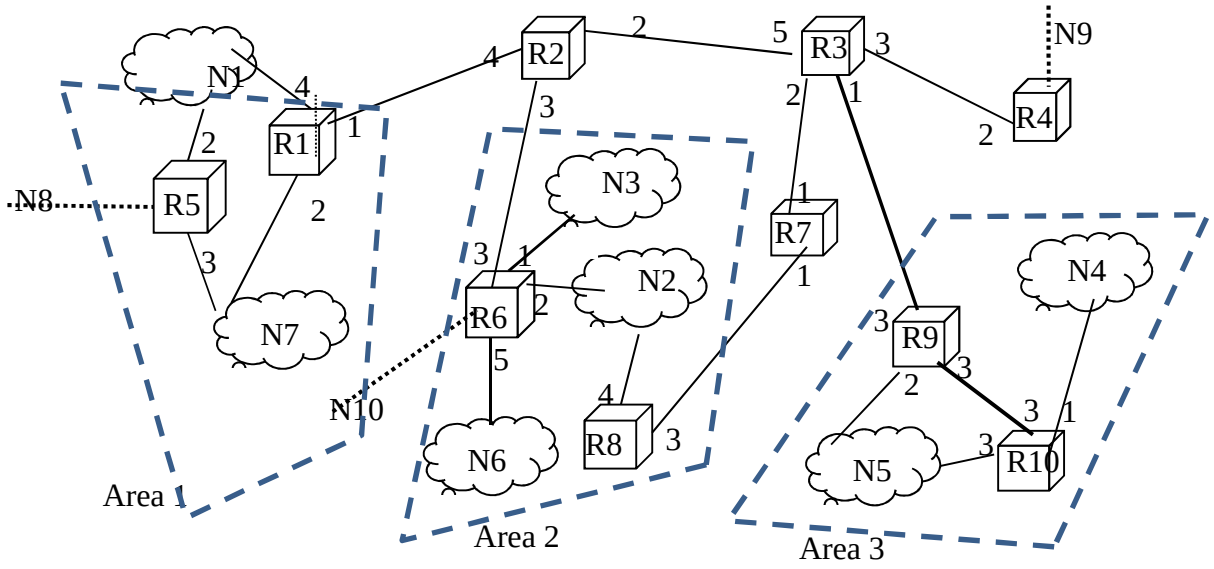
b) **[5 points]** Which of the routers in the autonomous system are border area routers? Briefly explain why border area routers run more than one instance of OSPF.

Area border routers run one "copy" of OSPF to manage routing within the area, and one "copy" of OSPF to manage routing in the backbone area. Routing in the backbone area allows routers in different areas to obtain routing information from other areas. R1, R6, R8 and R9 are border area routers.

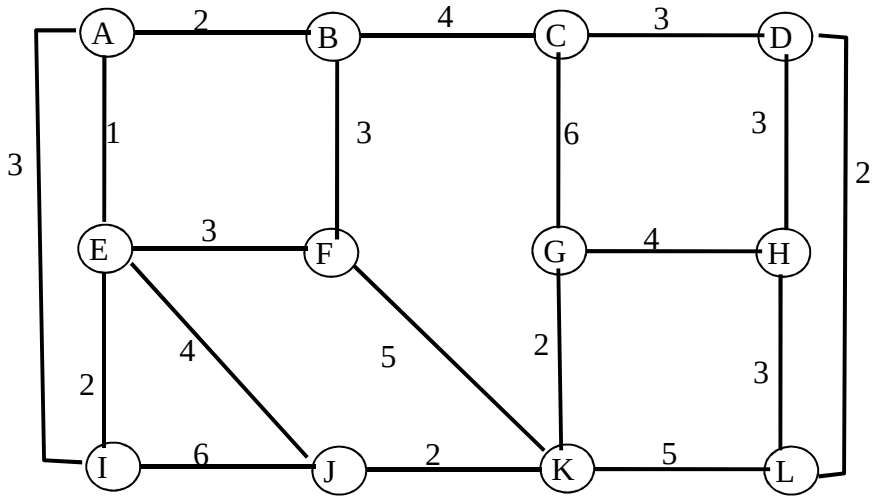
c) **[10 points]** Which of the routers in the AS are backbone routers? Briefly explain why boundary routers run more than one instance of OSPF.

R1, R2, R3, R4, R6, R7, R8 and R9 are backbone routers. Backbone routers are part of the backbone area that connects all the other areas. Border area routers are also backbone routers

d) **[5 points]** Which networks are stub networks? Briefly explain why.
N3, N6, N4, Data from one router is not sent to another router through a stub network



6. [30 points] Generate a least cost route to all other routers from Router A in the autonomous system below using Dijkstra's routing algorithm. In cases where there are multiple paths with the same cost choose the path with the fewest hops. Complete a table similar to the one illustrated in the animated detailed example in your class notes. When two paths have the same cost, choose the path with the smallest number of hops.



DYKSTRA'S ALGORITHM RESULTS

iteration	FROM NODE A	A		B		C		D		E		F		G		H		I		J		K		L	
1	{A}	**0**	**A**	2	A					**1**	**A**							3	A						
2	{A,E}	**0**	**A**	**2**	**A**					**1**	**A**							3	A	5	A-E				
3	{A,E,B}	**0**	**A**	**2**	**A**	6	A-B			**1**	**A**	4	A-E					**3**	**A**	5	A-E				
4	{A,E,B,I}	**0**	**A**	**2**	**A**	6	A-B			**1**	**A**	**4**	**A-E**					**3**	**A**	5	A-E				
5	{A,E,B,I,F}	**0**	**A**	**2**	**A**	6	A-B			**1**	**A**	**4**	**A-E**					**3**	**A**	**5**	**A-E**	9	A-E-F		
6	{A,E,B,I,F,J}	**0**	**A**	**2**	**A**	**6**	**A-B**			**1**	**A**	**4**	**A-E**					**3**	**A**	**5**	**A-E**	7	A-E-J		
7	{A,E,B,I,F,J,C}	**0**	**A**	**2**	**A**	**6**	**A-B**	9	A-B-C	**1**	**A**	**4**	**A-E**	12	A-B-C			**3**	**A**	**5**	**A-E**	**7**	**A-E-J**		
8	{A,E,B,I,F,J,C,K}	**0**	**A**	**2**	**A**	**6**	**A-B**	**9**	**A-B-C**	**1**	**A**	**4**	**A-E**	9	A-E-J-K			**3**	**A**	**5**	**A-E**	**7**	**A-E-J**	12	A-E-J-K
9	{A,E,B,I,F,J,C,,K,D}	**0**	**A**	**2**	**A**	**6**	**A-B**	**9**	**A-B-C**	**1**	**A**	**4**	**A-E**	**9**	**A-E-J-K**	12	A-B-C-D	**3**	**A**	**5**	**A-E**	**7**	**A-E-J**	11	A-B-C-D
10	{A,E,B,I,F,J,CK,,D,G}	**0**	**A**	**2**	**A**	**6**	**A-B**	**9**	**A-B-C**	**1**	**A**	**4**	**A-E**	**9**	**A-E-J-K**	12	A-B-C-D	**3**	**A**	**5**	**A-E**	**7**	**A-E-J**	**11**	**A-B-C-D**
11	{A,E,B,I,F,J,C,,K,D,G,L}	**0**	**A**	**2**	**A**	**6**	**A-B**	**9**	**A-B-C**	**1**	**A**	**4**	**A-E**	**9**	**A-E-J-K**	**12**	**A-B-C-D**	**3**	**A**	**5**	**A-E**	**7**	**A-E-J**	**11**	**A-B-C-D**
12	{A,E,B,I,F,J,C,,K,D,G,L,H}	**0**	**A**	**2**	**A**	**6**	**A-B**	**9**	**A-B-C**	**1**	**A**	**4**	**A-E**	**9**	**A-E-J-K**	**12**	**A-B-C-D**	**3**	**A**	**5**	**A-E**	**7**	**A-E-J**	**11**	**A-B-C-D**

7. Consider the CSMA MAC protocol

a) [10 points] What is a contention based protocol? Name two contention based protocols. Is CSMA a contention based protocol? What does the acronym CSMA stand for?

b) [10 points] Give a step by step description of how non persistent CSMA operates.

c) [10 points] Give a step by step description of how P-persistent CSMA operates.

a) ***CSMA (Carrier Sense Multiple Access) is a Medium Access Control Protocol (MAC) designed to share the transmission medium between multiple hosts connected to a single transmission medium (LAN). It is a contention based protocol. Multiple stations may transmit at the same time. If two stations transmit at the same time the transmission medium is in a state of contention and neither message can be expected to arrive correctly. The sending stations must recognize that this collision or contention has occurred and retransmit the data that was unable to successfully reach its destination.***

The Carrier Sense part of CSMA refers to watching or sensing the transmission medium to which the host is connected. If the medium is busy when the host is ready to transmit a packet it must wait until it senses the medium is free before its packet is transmitted.

b) ***A host using non-persistent CSMA will wait a length of time after determining the transmission medium is busy then sense the medium again to determine if it is still busy. If the medium is now free the packet waiting to be transmitted will be sent. Otherwise, it will wait another length of time and try again. The length of time between each sensing of the medium is a random duration chosen from a distribution of waiting times with a particular mean. The steps used for non-persistent CSMA are as follows:***

1. ***Sense the medium and determine if it is in use***
2. ***If the medium is not in use transmit the packet immediately***
3. ***If the medium is busy select a random waiting time from distribution with mean B.***
4. ***Wait for that length of time and sense the medium again***
5. ***If the medium is free transmit the packet***
6. ***If the medium is still busy double the mean of the distribution of waiting times, return to step 3.***
7. ***If the packet is transmitted without contention/collision we are done***

8. ***If a collision occurs during the transmission then double the mean of the distribution, select a random waiting time from the waiting time distribution, and return to step 1***

c) ***A host using p-persistent CSMA will continue to sense the busy transmission medium until it becomes free. As soon as the transmission medium becomes free the host will transmit its packet with probability p. The steps used for P-persistent CSMA are as follows:***

1. ***Sense the medium and determine if it is in use***
2. ***If the medium is not in use transmit the packet with probability P***
   a) ***If you do not transmit (probability 1-P) choose a random waiting time from the distribution of waiting times. After this waiting time return to step 1***
3. ***If the medium is busy continue sensing the medium until it is free***
4. ***When the medium is free transmit the packet with probability P***
   a) ***If you do not transmit immediately (probability 1-P), double the mean of the distribution of waiting times, choose a random waiting time from the distribution of waiting times. After this waiting time return to step 1***
5. ***If the packet is transmitted without contention/collision we are done***
6. ***If a collision occurs during the transmission than double the mean of the distribution, select a random waiting time from the waiting time distribution, and return to step 1.***

The doubling of waiting time occurs for the first 10 tries, after 16 tries the algorithm gives up and indicates the message cannot be delivered