

Due Monday 11/21 **at the beginning of class**

1. IP addressing

Consider the IPv4 address 128.239.253.6

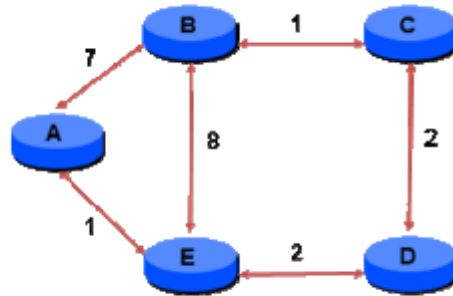
- a) Assuming we're using class-based addressing, explain i) what "class" the address is and ii) what is the address of the network part and iii) what is the address of the host part

- b) Assume that we're using classless (i.e., CIDR) addressing and the longest matching advertised route cover this address is 128.239.128.0/17. As with part "a" above explain i) what is the network part of the address and ii) what is the host part

- c) Inside the organization that owns this network (UCSD incidentally) they further subdivide the host part of the address (from part b above) into a subnet prefix and host address (let suppose the least significant 8 bits represent the host and the remainder identifies the subnet inside UCSD). How many different LANs can be supported with this subnetting architecture?

2. DV Routing

Consider the following network using distance vector routing:



Info at node	Distance to Node				
	A	B	C	D	E
A	0	6	5	3	1
B	6	0	1	3	5
C	5	1	0	2	4
D	3	3	2	0	2
E	1	5	4	2	0

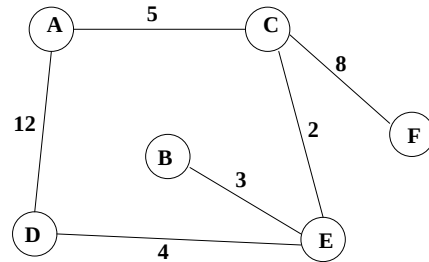
Suppose an additional link is added connecting A and D with **cost 2**. List the distance vector updates that will be sent for the system to reconverge (the first update, from A, is listed below). Assume that updates occur simultaneously in rounds (i.e., at a set time each node transmits their distance vectors to their neighbors). You only need list DV updates in a route that have changed since the last round. [also: ignore split horizon or poison reverse in this example]

Round 1:

A: 0, 6, 5, 2, 1 (reflecting only the change in the AD link)

3. Link State Routing

For the network show, complete the table below showing how the link-state algorithm builds the routing table for node D. Assume all link-state updates have been distributed (i.e., this is just the shortest path algorithm calculation) The syntax used below is as follows: (destination, distance, via which node)



	Confirmed	Tentative
1.	(D, 0, -)	(A, 12, A) (E, 4, E)
2.		
3.		
4.		
5.		
6.		
...		