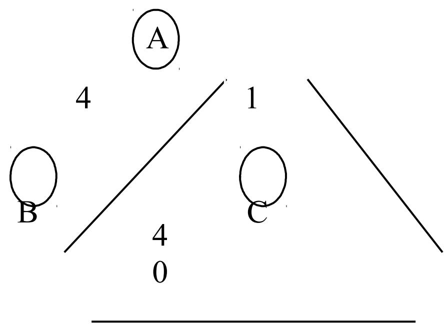
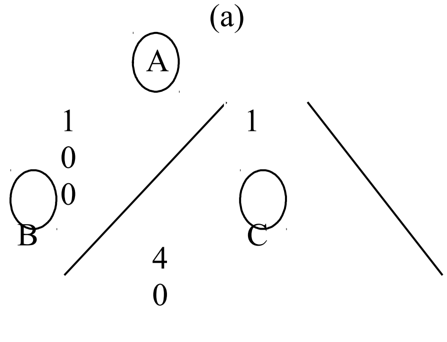
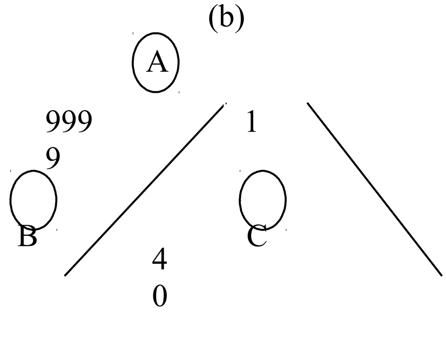
1. Suppose we are using Distance Vector Routing protocol.



(a)



(b)



(c)

(1) Please give the distance vector of A, B, and C as follows for scenario (a):

A to B: **4**	A to C: **1**
B to A: **4**	B to C: **5**
C to A: **1**	C to B: **5**

(2) What happens if link cost of AB becomes 100, as shown in scenario (b)?

A to B: **41**	A to C: **1**
B to A: **41**	B to C: **40**
C to A: **1**	C to B: **40**

(3) How many rounds it takes for the protocol to converge to the above actual minimum distance?

It will need at least two hops (rounds)

(4) What happens if link cost of AB changes from 100 to 9999, as shown in scenario (c)?

The shortest distant vector will remain unchanged after two hops. Although with one hop, the routing table will include the cost of AB

(5) What happens if link AB breaks, will the protocol be able to converge to some final minimum distance? If not, why?

By breaking link AB, it may cause a *count to infinity* problem. It depends how much the cost is for link CB. If the cost is relatively large, then it will not converge quickly. The reason being that from node C's routing table, it has the information that the least cost to node B is through node A. Node C will send a packet to node A, node A will send it to node B, but link AB is broken so then node A sends information back to node C. This transition will update the routing table of A and C by incrementing it by 1. Now we're back to node C. Node C will repeat the same process until infinity or until it reaches a relatively large cost.