

**1. KR, #R4**

Link state algorithms: Computes the least-cost path between source and destination using complete, global knowledge about the network. Distance-vector routing: The calculation of the least-cost path is carried out in an iterative, distributed manner. A node only knows the neighbor to which it should forward a packet in order to reach given destination along the least-cost path, and the cost of that path from itself to the destination.

**2. KR, #R5**

The count-to-infinity problem refers to a problem of distance vector routing. The problem means that it takes a long time for a distance vector routing algorithm to converge when there is a link cost increase. For example, consider a network of three nodes x, y, and z. Suppose initially the link costs are  $c(x,y)=4$ ,  $c(x,z)=50$ , and  $c(y,z)=1$ . The result of distance-vector routing algorithm says that z's path to x is  $z \rightarrow y \rightarrow x$  and the cost is  $5(=4+1)$ . When the cost of link (x,y) increases from 4 to 60, it will take 44 iterations of running the distance-vector routing algorithm for node z to realize that its new least-cost path to x is via its direct link to x, and hence y will also realize its least-cost path to x is via z.

**3. KR, #R6**

No. Each AS has administrative autonomy for routing within an AS.

**4. KR, #P3**

Step	N'	D(t),p(t)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	x	$\infty$	$\infty$	3,x	6,x	6,x	8,x
1	xv	7,v	6,v	3,x	6,x	6,x	8,x
2	xvu	7,v	6,v	3,x	6,x	6,x	8,x
3	xvuw	7,v	6,v	3,x	6,x	6,x	8,x
4	xvuwyt	7,v	6,v	3,x	6,x	6,x	8,x
5	xvuwytz	7,v	6,v	3,x	6,x	6,x	8,x
6	xvuwytz	7,v	6,v	3,x	6,x	6,x	8,x

**5. KR, #P7**

$D_x(w) = 2$ ,  $D_x(y) = 4$ ,  $D_x(u) = 7$

**6. KR, #P14**

a) eBGP

- b) iBGP**
- c) eBGP**
- d) iBGP**

**7. KR, #P15**

- a) I1 because this interface begins the least cost path from 1d towards the gateway router 1c.**
- b) I2. Both routes have equal AS-PATH length but I2 begins the path that has the closest NEXT-HOP router.**
- c) I1. I1 begins the path that has the shortest AS-PATH.**