Question 1: Dijkstra’s Algorithm
Dijkstra’s algorithm in a nutshell:

1. Pick the unvisited vertex with the lowest distance.
2. Calculate the distance through it to each neighbor, and update the neighbor’s distance if smaller.
3. Mark the vertex as visited when done with all neighbors.
Question 1
Question 1
Question 1
Question 1
Question 1
Question 1
Question 1
Question 2: Distance-Vector Routing
Setting of distance-vector routing:

- Nodes know only the cost to their neighbor; not the topology.
- Nodes can talk only to their neighbors.
- All nodes run the same algorithm.
- Nodes and links may fail; messages may be lost.
The distance-vector routing algorithm:

Each node maintains a vector of distances (and next hops) to all destinations.

1. Initialize vector with 0 to self, $\infty$ to others.
2. Periodically send vector to neighbors.
3. Update vector for each destination by selecting the shortest distance heard, after adding the cost of the neighbor link.
Question 2
**Question 2**

**Time 0**

- **A to D**: distance 3
- **B to D**: distance 3
- **C to D**: distance 2
*The link fails*

**Time 1**
- A to D: distance 3
- B to D: distance 3
- C to D: **distance 4**
Time 2

A to D: distance 5
B to D: distance 5
C to D: distance 4
Question 2

Time 96
A to D: distance 99
B to D: distance 99
C to D: distance 98
Time 97
A to D: distance 99
B to D: distance 99
C to D: distance 100
Time 98

A to D: distance 101 (over C)
B to D: distance 100 (directly)
C to D: distance 100
Time 99
A to D: distance 101 (over C)
B to D: distance 100 (directly)
C to D: distance 101 (over B)
Question 2

Time 100: Stable state
A to D: distance 102 (over C)
B to D: distance 100 (directly)
C to D: distance 101 (over B)

Eventually, the problem is resolved and the system reaches a stable state.