Problem

- Here problem is when a router receives an IP packet with destination address then how can it decide to interface send this packet.

- This decision at the router, is taken with the help of a routing table.

- Actually the process of designing a routing table is called routing. Taking a packet and sending it to some path is actually switching.

- One question, is it possible that a packet reaches its destination without routing table, actually yes, the process is called flooding.

- That is instead of trying to identify the shortest path, we can send it to all possible way and then we can be sure that at least one packet will reach the destination.

- Flooding Advantage
  - No Routing Algorithm is required
  - Shortest Path is guaranteed
  - Highly Reliable

- Flooding Disadvantage
  - Duplicate packets will arrive at destination and intermediate router
  - Traffic is high

- Note: In Security network mostly, flooding is used

Break

ROUTING

- A routing table contains information about the network, and it helps deciding to which interface the incoming packet should be sent inorder to reach destination.

- Routing table can be either static or dynamic.

- A static table is one with manual entries, i.e. if someone has information about all the routers in the network and can compute the shortest distance from one router to another and can upload the routing information in a table for each router then it is called Static Routing.
  - Now-a days as we know the internet is so complex that no one can have complete information about the entire internet
  - Internet keeps on changing some new routers come and some old routers may go down, means topology and traffic keeps on changing.
  - Conclusion Static routing is not possible.

- A dynamic table, on the other hand, is one that is updated automatically, without human intervention, when there is a change somewhere in the internet either in topology or traffic.

Break

UNICAST ROUTING PROTOCOLS

- Routing protocols have been created in response to the demand for dynamic routing tables. A routing protocol is a combination of rules and procedures that lets routers in the internet inform each other of changes.

- It allows routers to share whatever they know about the internet or their neighbourhood. The sharing of information allows a router in Delhi to know about the failure of a network in Singapore.

- The routing protocols also include procedures for combining information received from other routers.

- Router to have several routing tables based on the required type of service.

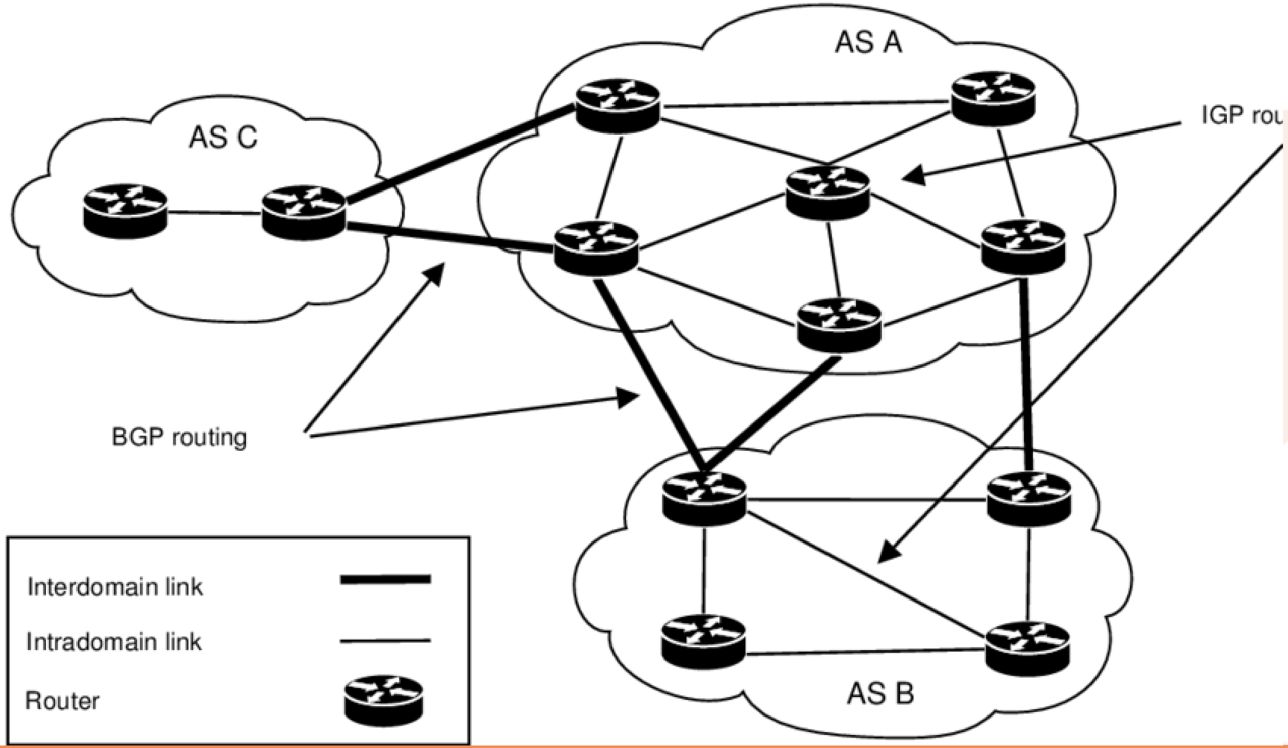
```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0         10.0.0.1       10.0.0.75     35
         10.0.0.0    255.255.255.0         On-link        10.0.0.75    291
        10.0.0.75  255.255.255.255         On-link        10.0.0.75    291
       10.0.0.255  255.255.255.255         On-link        10.0.0.75    291
        127.0.0.0        255.0.0.0         On-link        127.0.0.1    331
        127.0.0.1  255.255.255.255         On-link        127.0.0.1    331
  127.255.255.255  255.255.255.255         On-link        127.0.0.1    331
     192.168.56.0    255.255.255.0         On-link     192.168.56.1    281
     192.168.56.1  255.255.255.255         On-link     192.168.56.1    281
   192.168.56.255  255.255.255.255         On-link     192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link        127.0.0.1    331
        224.0.0.0        240.0.0.0         On-link     192.168.56.1    281
        224.0.0.0        240.0.0.0         On-link        10.0.0.75    291
  255.255.255.255  255.255.255.255         On-link        127.0.0.1    331
```

Break

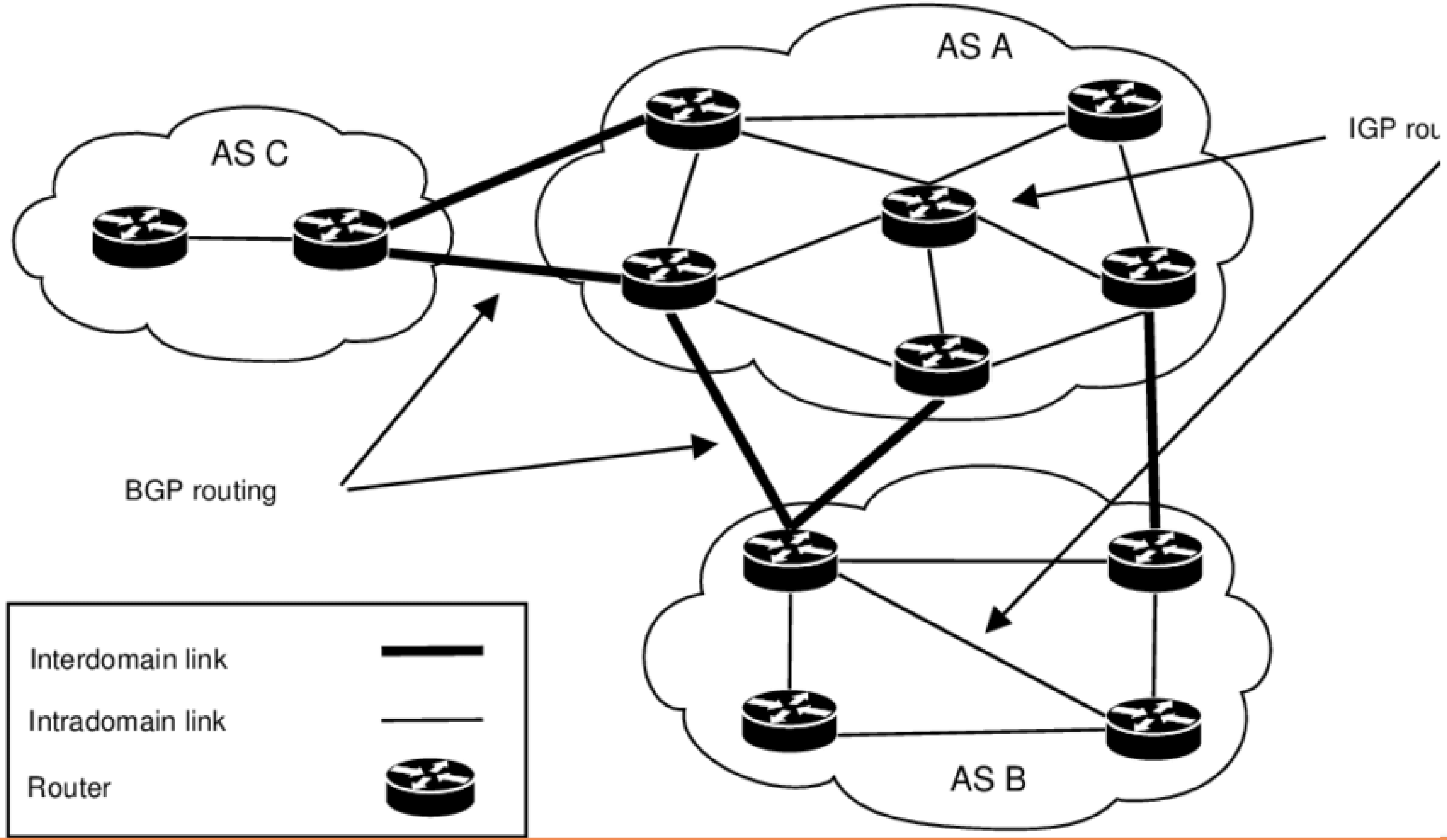
Intra-domain and Interdomain Routing

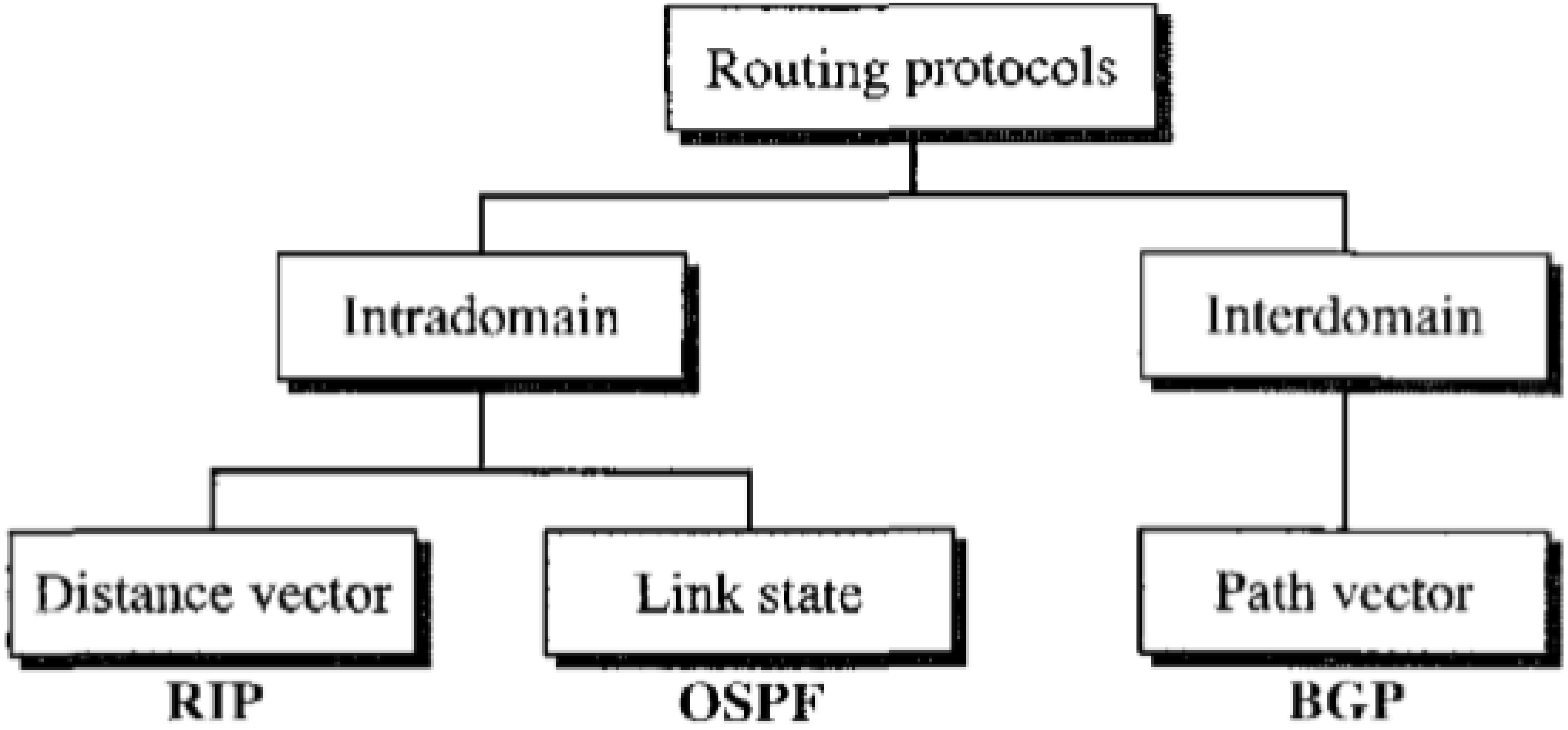
- Today, an internet can be so large that one routing protocol cannot handle the task of updating the routing tables of all routers. For this reason, an internet is divided into autonomous systems.

- An autonomous system (AS) is a group of networks and routers under the authority of a single administration. Routing inside an autonomous system is referred to as intradomain routing.



- Routing between autonomous systems is referred to as interdomain routing.
- Each autonomous system can choose one or more intradomain routing protocols to handle routing inside the autonomous system. However, only one interdomain routing protocol handles routing between autonomous systems.




Routing protocols
Intradomain
Interdomain
Distance vector
Link state
Path vector
RIP
OSPF
BGP

- Which of the available pathways is the optimum pathway? What is the definition of the term optimum? And What is the definition of the term cost

- However, the metric assigned to each network depends on the type of protocol. Some simple protocols, such as the Routing Information Protocol (RIP), treat all networks as equals. The cost of passing through a network is the same; it is one hop count. So if a packet passes through 10 networks to reach the destination, the total cost is 10 hop counts.

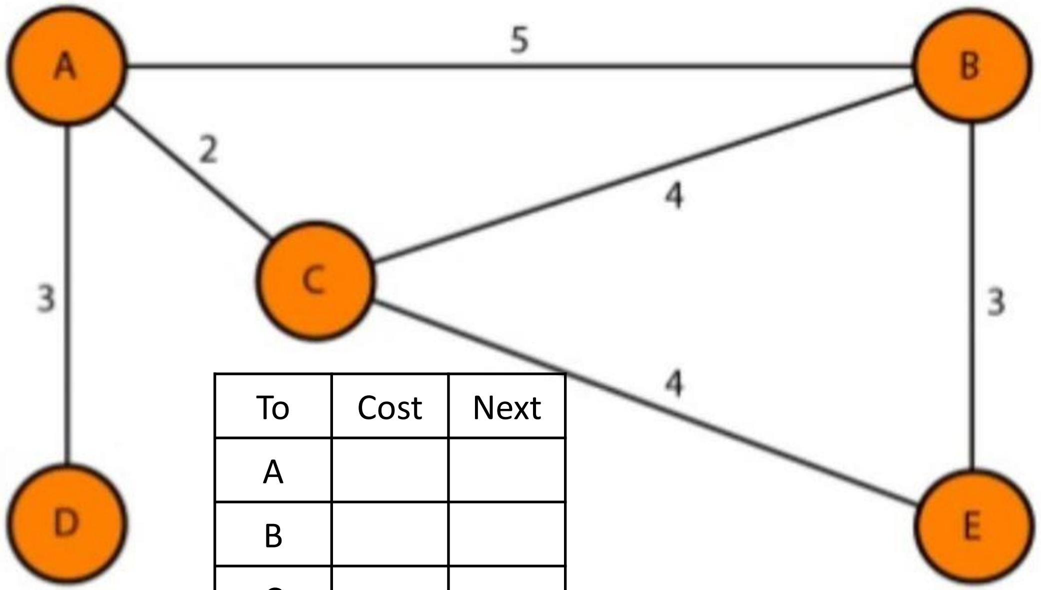
- Other protocols, such as Open Shortest Path First (OSPF), allow the administrator to assign a cost for passing through a network based on the type of service required. A route through a network can have different costs (metrics).

Break

Distance Vector Routing

- In distance vector routing, the least-cost route between any two nodes is the route with minimum distance.
- In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node. The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing).

To	Cost	Next
A		
B		
C		
D		
E		



To	Cost	Next
A		
B		
C		
D		
E		

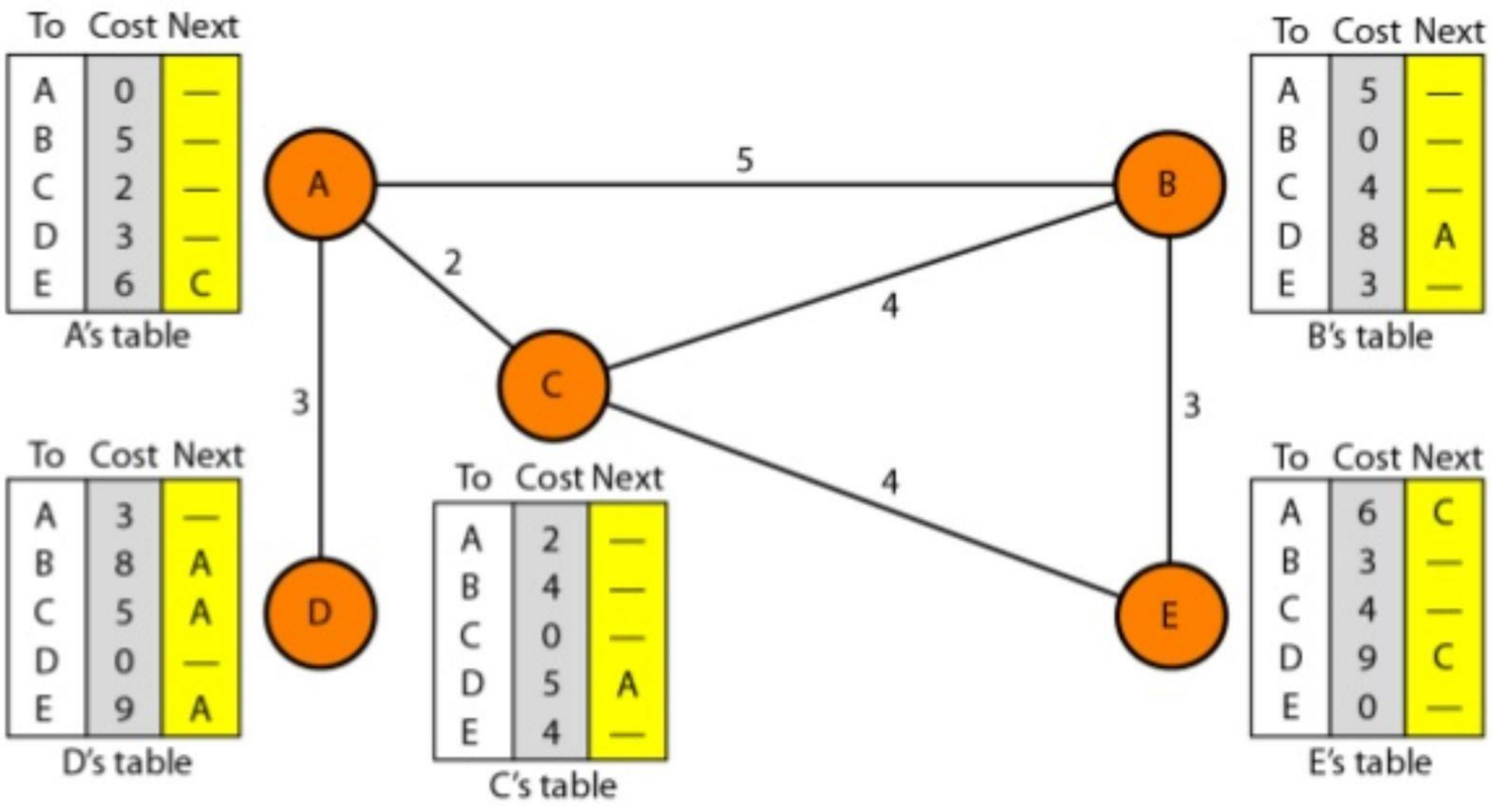
To	Cost	Next
A		
B		
C		
D		
E		

To	Cost	Next
A		
B		
C		
D		
E		

To	Cost	Next
A		
B		
C		
D		
E		

- The whole idea of distance vector routing is the sharing of information between neighbours. Although node A does not know about node E, node C does. So, if node C shares its routing table with A, node A can also know how to reach node E.

- On the other hand, node C does not know how to reach node D, but node A does. If node A shares its routing table with node C, node C also knows how to reach node D. In other words, nodes A and C, as immediate neighbours, can improve their routing tables if they help each other.



- There is only one problem. How much of the table must be shared with each neighbour? A node is not aware of a neighbour’s table. The best solution for each node is to send its entire table to the neighbour and let the neighbour decide what part to use and what part to discard.

- However, the third column of a table (next stop) is not useful for the neighbour. When the neighbour receives a table, this column needs to be replaced with the sender's name. If any of the rows can be used, the next node is the sender of the table.

- A node therefore can send only the first two columns of its table to any neighbour. In other words, sharing here means sharing only the first two columns.

Break

- The question now is, when does a node send its partial routing table (only two columns) to all its immediate neighbours? The table is sent both periodically and when there is a change in the table.

- **Periodic Update** A node sends its routing table, normally every 30 s, in a periodic update. The period depends on the protocol that is using distance vector routing.

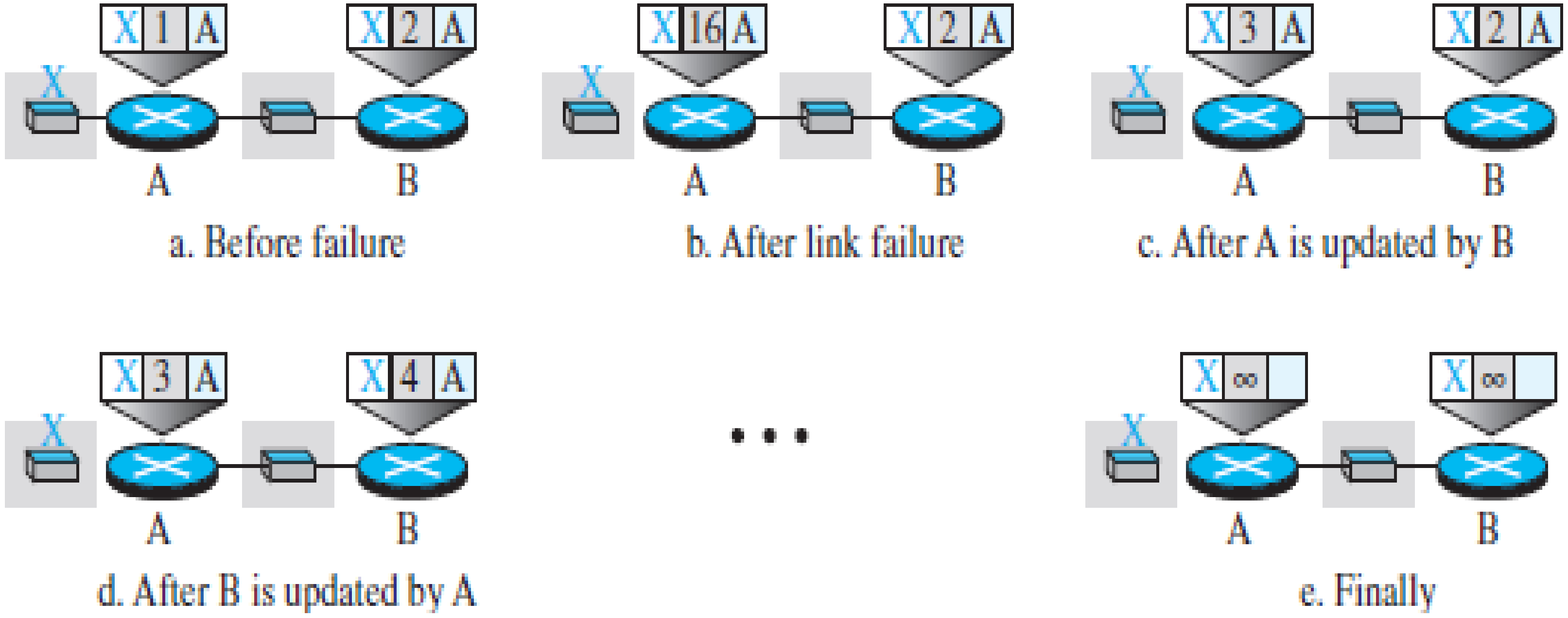
- **Triggered Updat**e A node sends its two-column routing table to its neighbours anytime there is a change in its routing table. This is called a triggered update. The change can result from the following.
  - A node receives a table from a neighbour, resulting in changes in its own table after updating.
  - A node detects some failure in the neighbouring links which results in a distance change to infinity.

Two-Node Loop Instability

- If a link is broken (cost becomes infinity), every other router should be aware of it immediately, but in distance-vector routing, this takes some time as the algorithms is designed in such a way that it reports the minimum first.

- The problem is referred to as *count to infinity*. It sometimes takes several updates before the cost for a broken link is recorded as infinity by all routers.

- Initially in fig.(a) nodes A and B know how to reach node X.
- But suddenly, the link between A and X fails. Node A changes its table.
- If A can send its table to B immediately, everything is fine.
- The system becomes unstable if B sends its forwarding table to A before receiving A's forwarding table.
- Node A receives the update assumes that B has found a way to reach X, it updates its forwarding table.
- Now A sends its new update to B. Now B thinks that something has been changed around A and updates its forwarding table.
- The cost of reaching X increases gradually until it reaches infinity.
- At this moment, both A and B know that X cannot be reached.



- During this time the system is not stable. Node A thinks that the route to X is via B; node B thinks that the route to X is via A.
- If A receives a packet destined for X, the packet goes to B and then comes back to A and vice versa. Packets bounce between A and B, creating a two-node loop problem.

Split Horizon

- In this strategy, instead of flooding the table through each interface, each node sends only part of its table through each interface.

- If, according to its table, node B thinks that the optimum route to reach X is via A, it does not need to advertise this piece of information to A; the information has come from A (A already knows).

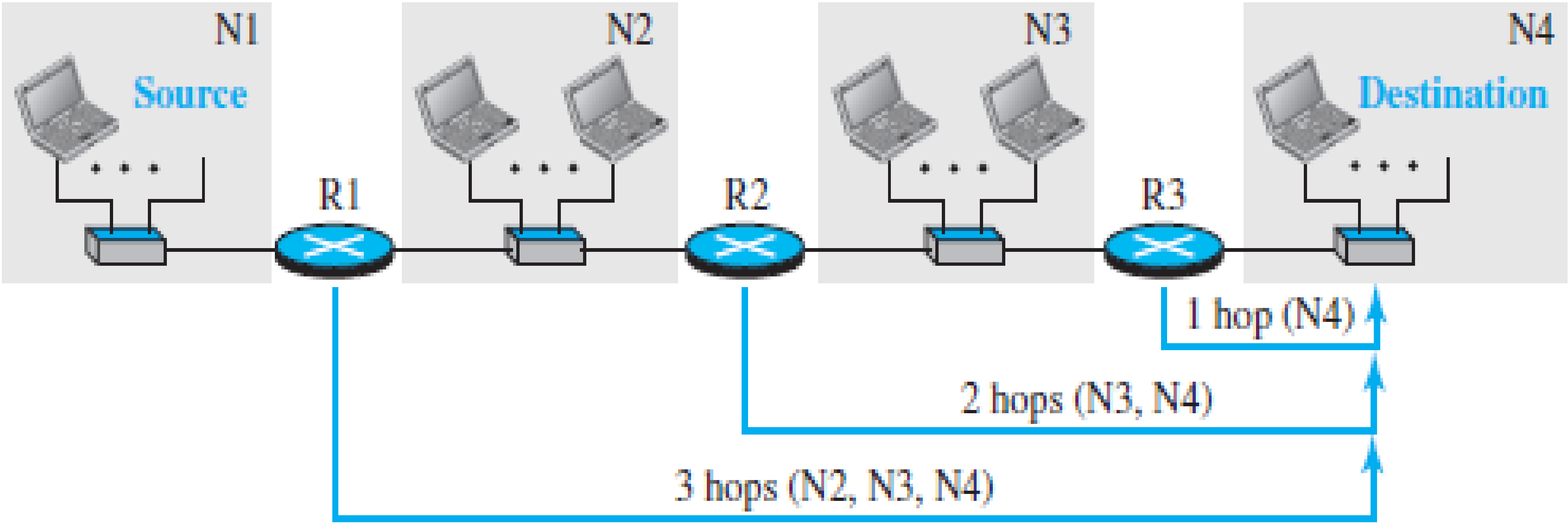
- Taking information from node A, modifying it, and sending it back to node A creates the confusion. In our scenario, node B eliminates the last line of its routing table before it sends it to A.

- In this case, node A keeps the value of infinity as the distance to X. Later when node A sends its routing table to B, node B also corrects its routing table. The system becomes stable after the first update: both node A and B know that X is not reachable

Break

Routing Information Protocol (RIP)

- The **Routing Information Protocol** (**RIP**) is one of the most widely used intradomain routing protocols based on the distance-vector routing algorithm
- **Metric**
  - **Hop Count:** To make the implementation of the cost simpler, the cost is defined as the number of hops, which means the number of networks (subnets) a packet needs to travel through from the source router to the final destination host.
  - The network in which the source host is connected is not counted in this calculation because the source host does not use a forwarding table; the packet is delivered to the default router.
  - ***In RIP, the maximum cost of a path can be 15, which means 16 is considered as infinity.***



RIP Implementation

- RIP is implemented as a process that uses the service of UDP on the well-known port number 520.
- RIP runs at the application layer, but creates forwarding tables for IP at the network later.

RIP Algorithm

- RIP implements the same algorithm as the distance-vector routing algorithm, but some changes need to be made to the algorithm to enable a router to update its forwarding table:
  - Instead of sending only distance vectors, a router needs to send the whole contents of its forwarding table in a response message.
  - The receiver adds one hop to each cost and changes the next router field to the address of the sending router.
  - The received router selects the old routes as the new ones except in the following three cases:
    - If the received route does not exist in the old forwarding table, it should be added to the route.
    - If the cost of the received route is lower than the cost of the old one, the received route should be selected as the new one.
    - If the cost of the received route is higher than the cost of the old one, but the value of the next router is the same in both routes, the received route should be selected as the new one.
  - The new forwarding table needs to be sorted according to the destination route (mostly using the longest prefix first).

Timers in RIP

RIP uses three timers to support its operation:

- The ***periodic timer*** controls the advertising of regular update messages. Generally used to prevent all routers sending their messages at the same time and creating excess traffic.

- The ***expiration timer*** governs the validity of a route.
  - When a router receives update information for a route, the expiration timer is set to 180 seconds for that particular route.
  - Every time a new update for the route is received, the timer is reset. If there is a problem on an internet and no update is received within the allotted 180 seconds, the route is considered expired and the hop count of the route is set to 16.

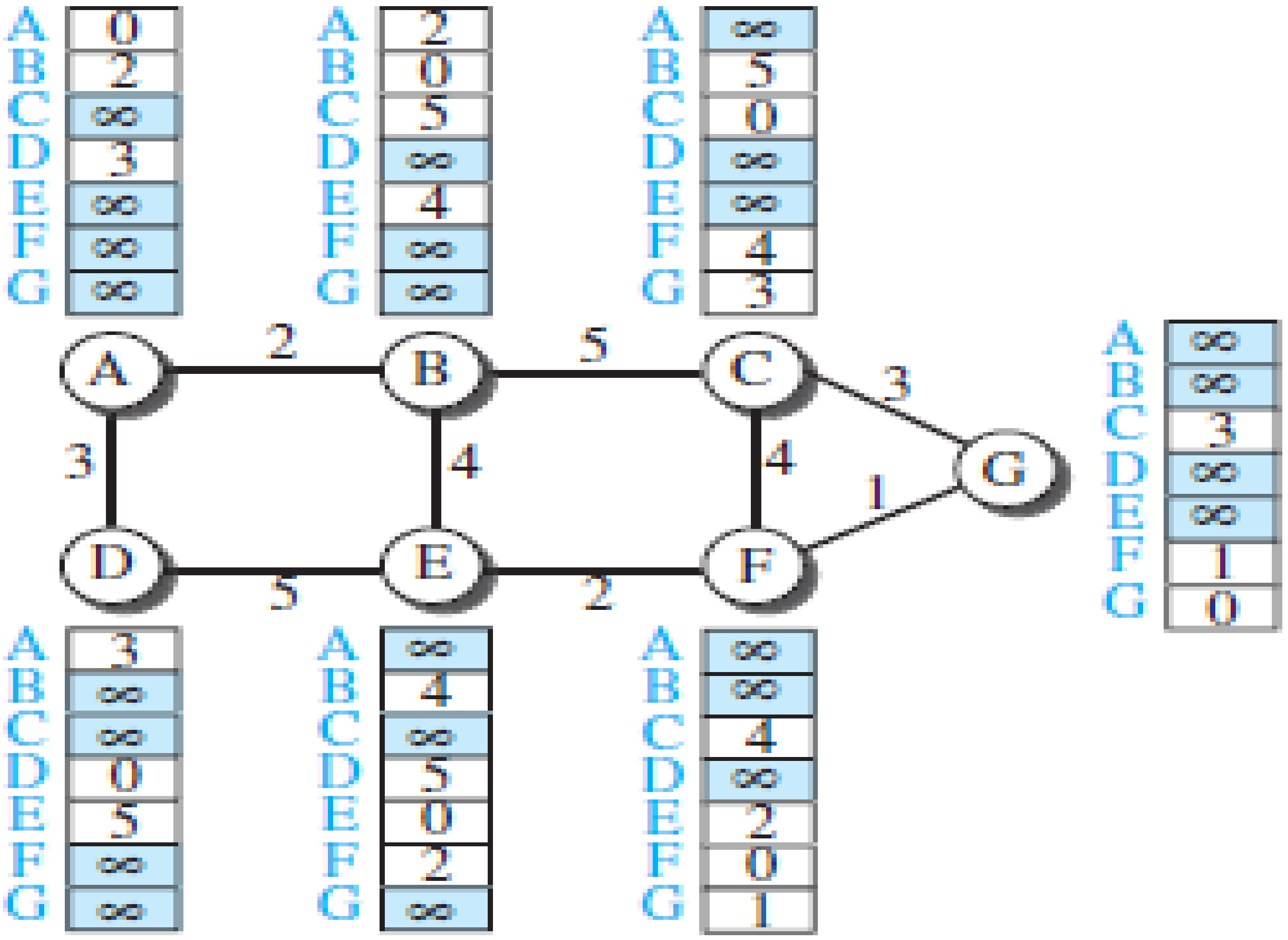
- The ***garbage collection timer*** is used to purge a route from the forwarding table.
  - When the information about a route becomes invalid, the router does not immediately purge that route from its table.
  - Instead, it continues to advertise the route with a metric value of 16.
  - At the same time, a garbage collection timer is set to 120 seconds for that route. When the count reaches zero, the route is purged from the table.

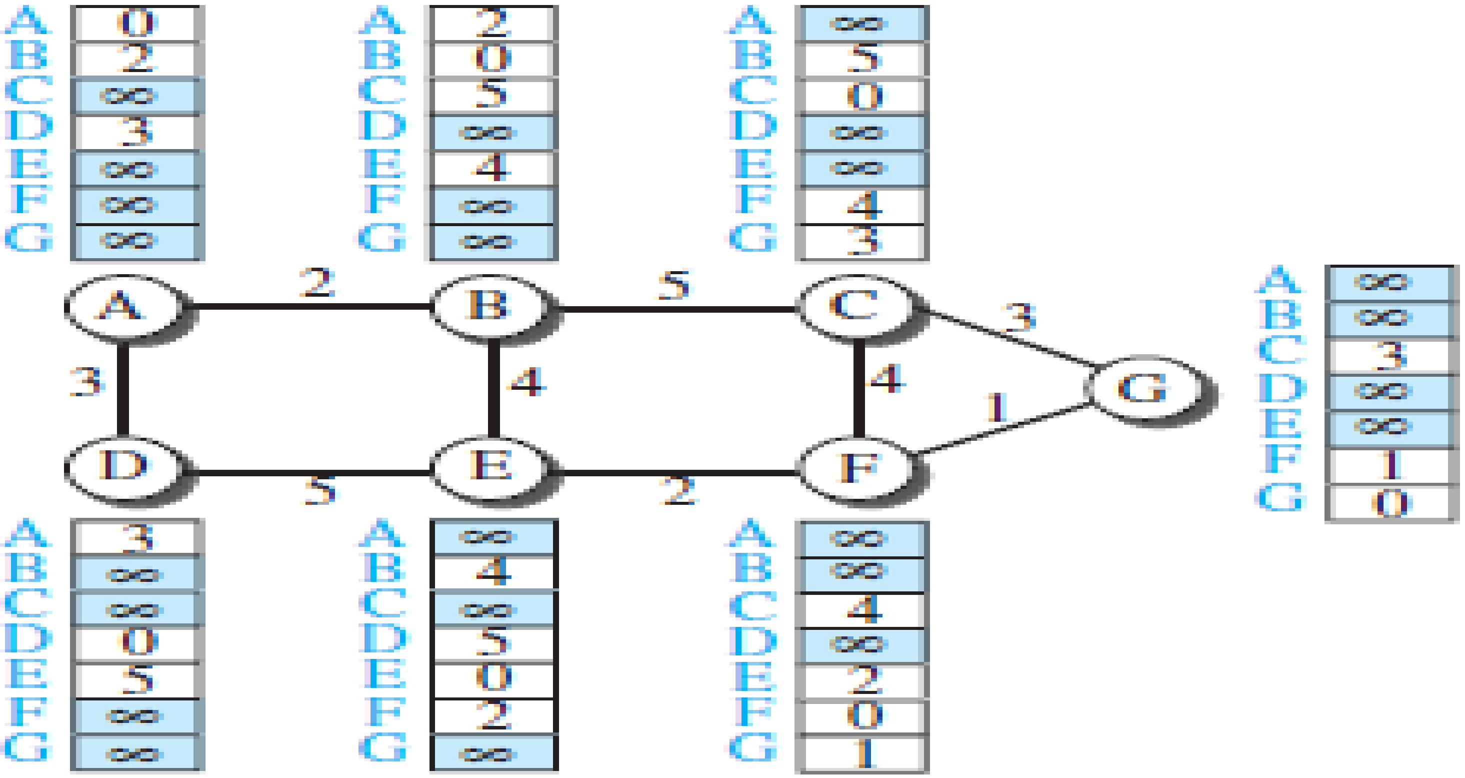
Performance of RIP

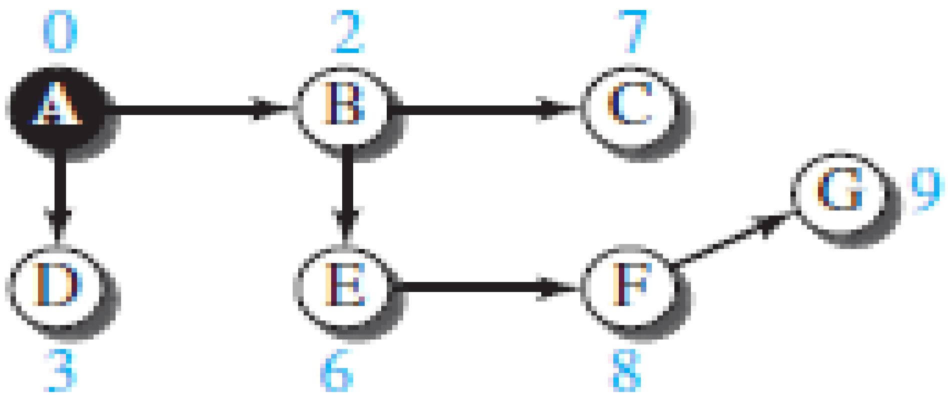
- ***Less Traffic:*** The update messages in RIP have a very simple format and are sent only to neighbours. They do not normally create traffic because the routers try to avoid sending them at the same time.

- ***Convergence of Forwarding Tables:*** since RIP allows only 15 hops in a domain (16 is considered as infinity), there is normally no problem in convergence. The only problems that may slow down convergence are count-to-infinity and loops created in the domain.

- ***Robustness:*** If there is a failure or corruption in one router, the problem will be propagated to all routers and the forwarding in each router will be affected, as it works on distance vector routing.


A B C D E F G
A 0 2 ∞ 3 ∞ ∞ ∞
B 2 0 5 ∞ 4 ∞ ∞
C ∞ 5 0 ∞ ∞ 4 3
A B C D E F G
2 5 3 3 4 4 1 5 2
G ∞ ∞ 3 ∞ ∞ 1 0
D 3 ∞ ∞ 0 5 ∞ ∞
E ∞ 4 ∞ 5 0 2 ∞
F ∞ ∞ 4 ∞ 2 0 1



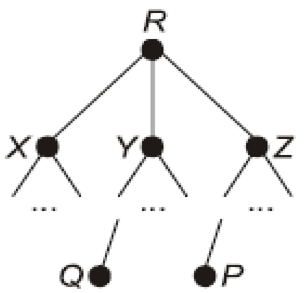


A	0
B	2
C	7
D	3
E	6
F	8
G	9

Break

(GATE- 2021)

Q.43 Consider a computer network using the distance vector routing algorithm in its network layer. The partial topology of the network is as shown below.



The objective is to find the shortest-cost path from the router R to routers P and Q. Assume that R does not initially know the shortest routes to P and Q. Assume that R has three neighbouring routers denoted as X, Y and Z. During one iteration, R measures its distance to its neighbours X, Y and Z as 3, 2 and 5, respectively. Router R gets routing vectors from its neighbours that indicate that the distance to router P from routers X, Y and Z are 7, 6 and 5, respectively. The routing vector also indicates that the distance to router Q from routers X, Y and Z are 4, 6 and 8, respectively. Which of the following statement(s) is/are correct with respect to the new routing table of R, after updation during this iteration?

(a) The distance from R to Q will be stored as 7.
(b) The distance from R to P will be stored as 10.
(c) The next hop router for a packet from R to Q is Z.
(d) The next hop router for a packet from R to P is Y.

Q Consider a network with five nodes, N_1 to N_5, as shown below. The network uses a Distance Vector Routing protocol. Once the routes have stabilized, the distance vectors at different nodes are as following.

N_1:(0, 1, 7, 8, 4)

N_2:(1, 0, 6, 7, 3)

N_3:(7, 6, 0, 2, 6)

N_4:(8, 7, 2, 0, 4)

N_5:(4, 3, 6, 4, 0)

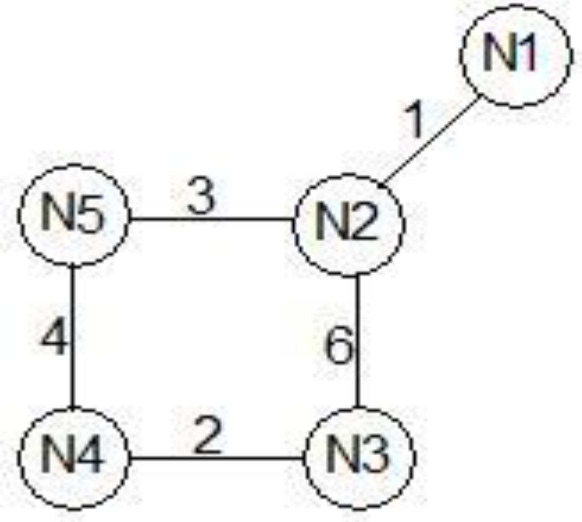
Each distance vector is the distance of the best known path at that instance to nodes, N_1 to N_5, where the distance to itself is 0. Also, all links are symmetric and the cost is identical in both directions. In each round, all nodes exchange their distance vectors with their respective neighbours. Then all nodes update their distance vectors. In between two rounds, any change in cost of a link will cause the two incident nodes to change only that entry in their distance vectors. The cost of link N_2-N_3 reduces to 2 (in both directions). After the next round of update what will be the new distance vector at node, N3? **(GATE-2011) (2 Marks)**

(A) (3, 2, 0, 2, 5)

(B) (3, 2, 0, 2, 6)

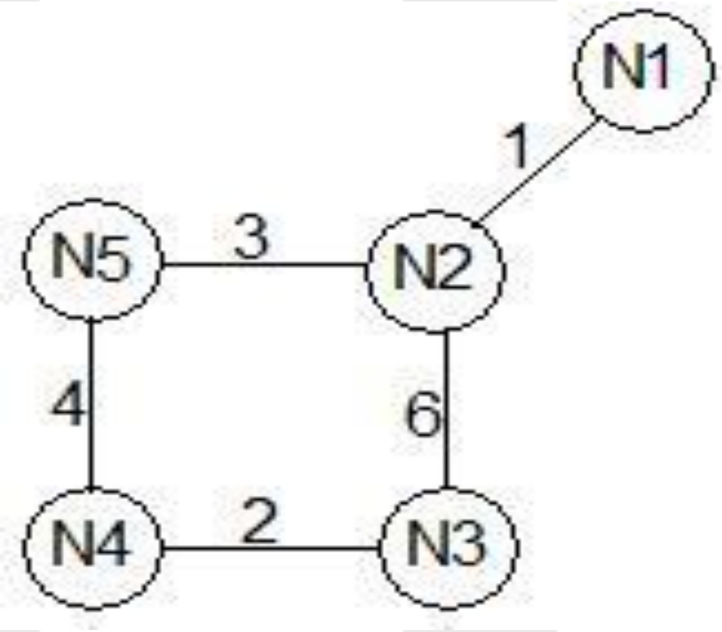
(C) (7, 2, 0, 2, 5)

(D) (7, 2, 0, 2, 6)

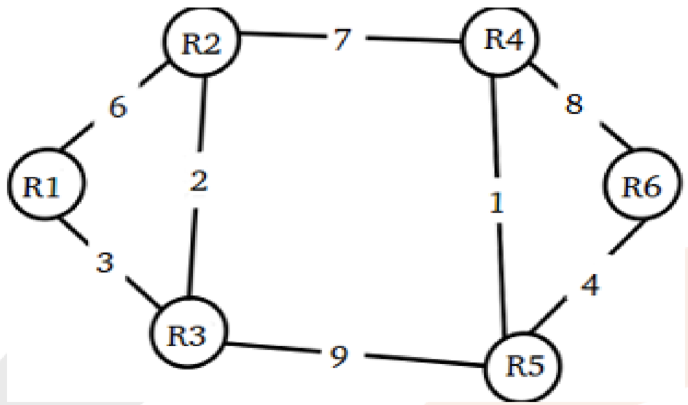


Q After the update in the previous question, the link N_1-N_2 goes down. N_2 will reflect this change immediately in its distance vector as cost, ∞. After the NEXT ROUND of update, what will be cost to N_1 in the distance vector of N_3? **(GATE-2011) (2 Marks)**

(A) 3 **(B)** 9 **(C)** 10 **(D)** ∞



Q Consider a network with 6 routers R_1 to R_6 connected with links having weights as shown in the following diagram



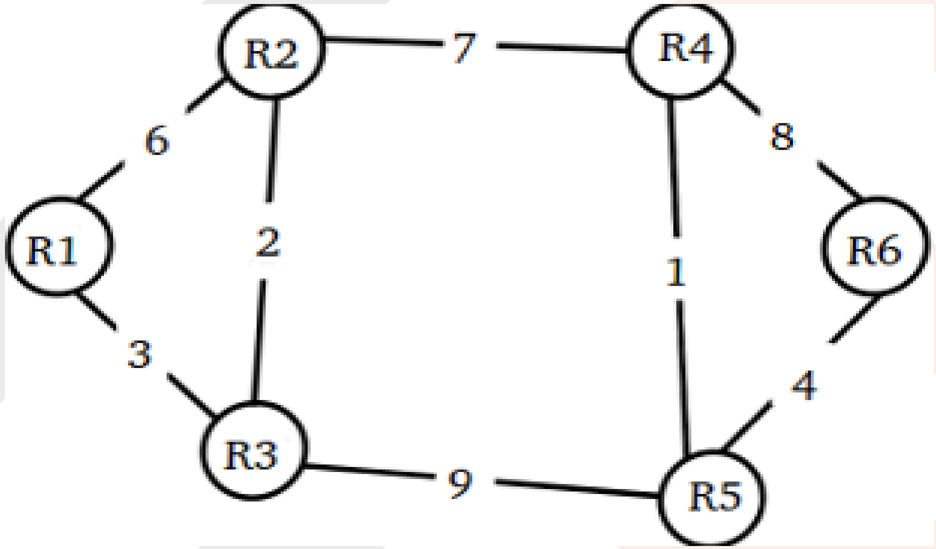
All the routers use the distance vector-based routing algorithm to update their routing tables. Each router starts with its routing table initialized to contain an entry for each neighbour with the weight of the respective connecting link. After all the routing tables stabilize, how many links in the network will never be used for carrying any data?

(GATE-2010) (2 Marks)

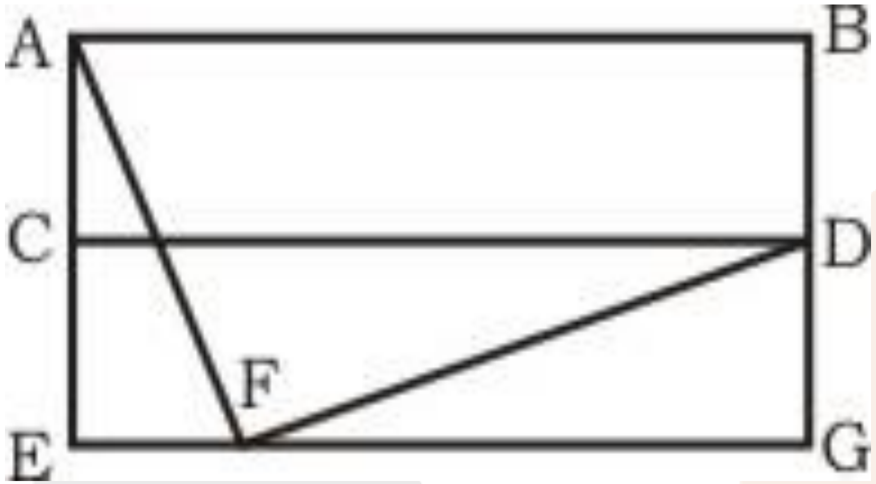
(A) 4 **(B)** 3 **(C)** 2 **(D)** 1

Q Suppose the weights of all unused links in the previous question are changed to 2 and the distance vector algorithm is used again until all routing tables stabilize. How many links will now remain unused? **(GATE-2010) (2 Marks)**

(A) 0 **(B)** 1 **(C)** 2 **(D)** 3



Q For the network given in the figure below, the routing tables of the four nodes A, E, D and G are shown. Suppose that F has estimated its delay to its neighbours, A, E, D and G as 8, 10, 12 and 6 msec respectively and updates its routing table using distance vector routing technique. **(GATE-2007) (2 Marks)**



Routing Table of A	
A	0
B	40
C	14
D	17
E	21
F	9
G	24

Routing Table of D	
A	20
B	8
C	30
D	0
E	14
F	7
G	22

Routing Table of E	
A	24
B	27
C	7
D	20
E	0
F	11
G	22

Routing Table of G	
A	21
B	24
C	22
D	19
E	22
F	10
G	0

A	8
B	20
C	17
D	12
E	10
F	0
G	6

A	21
B	8
C	7
D	19
E	14
F	0
G	22

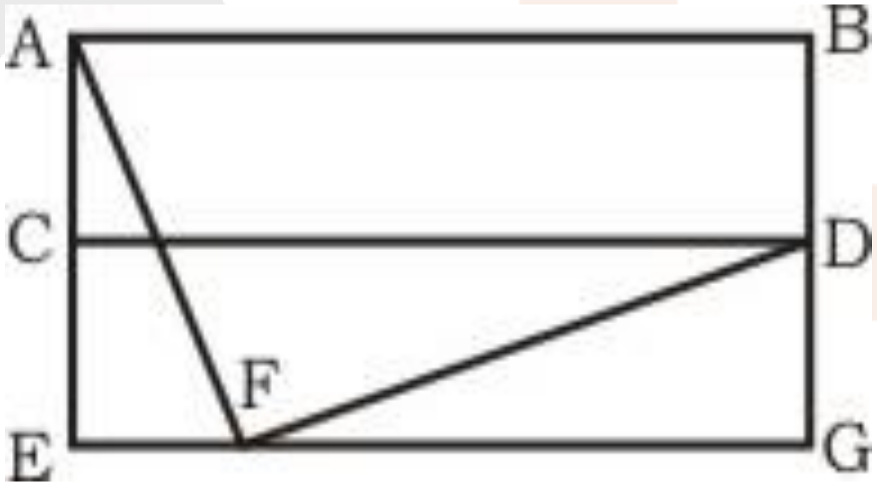
A	8
B	20
C	17
D	12
E	10
F	16
G	6

A	8
B	8
C	7
D	12
E	10
F	0
G	6

Break

Link State Routing

- Link state routing has a different philosophy from that of distance vector routing. In link state routing, if each node in the domain has the entire topology of the domain the list of nodes and links, how they are connected including the type, cost (metric), and condition of the links (up or down)-the node can use Dijkstra's algorithm to build a routing table.
- Each node uses the same topology to create a routing table, but the routing table for each node is unique because the calculations are based on different interpretations of the topology.
- The topology must be dynamic, representing the latest state of each node and each link. If there are changes in any point in the network (a link is down, for example), the topology must be updated for each node.

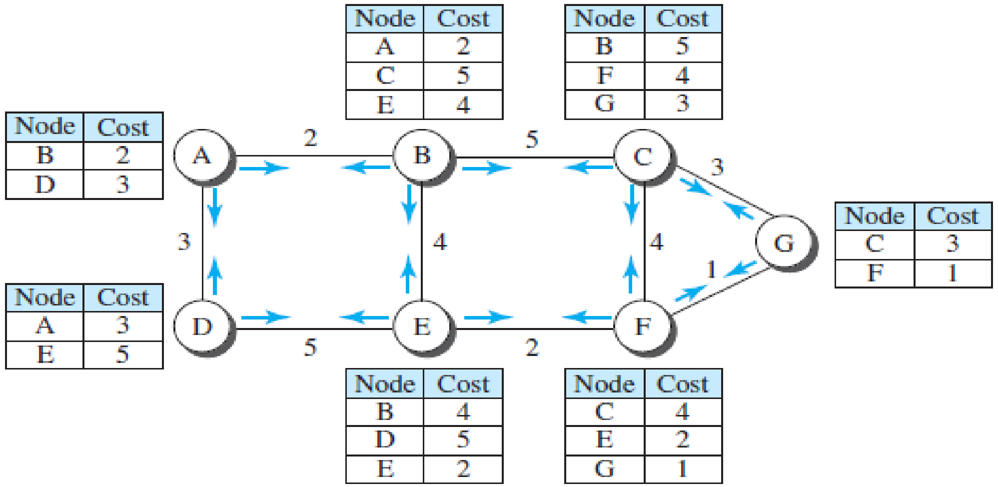


- How can a common topology be dynamic and stored in each node?
- Link state routing is based on the assumption that, although the global knowledge about the topology is not clear, each node has partial knowledge.
- it knows the state (type, condition, and cost) of its links. In other words, the whole topology can be compiled from the partial knowledge of each node.
- In this algorithm the cost associated with an edge defines the state of the link.

Break

Building Routing Tables

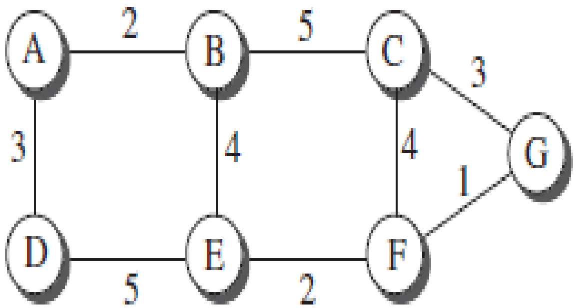
- In link state routing, four sets of actions are required to ensure that each node has the routing table showing the least-cost node to every other node.
  1. Creation of the states of the links by each node, called the link state packet (LSP).
  2. Dissemination of LSPs to every other router, called flooding, in an efficient and reliable way.
  3. Formation of a shortest path tree for each node.
  4. Calculation of a routing table based on the shortest path tree



- Formation of Shortest Path Tree: Dijkstra Algorithm After receiving all LSPs, each node will have a copy of the whole topology. However, the topology is not sufficient to find the shortest path to every other node; a shortest path tree is needed.

- A tree is a graph of nodes and links; one node is called the root. All other nodes can be reached from the root through only one single route.

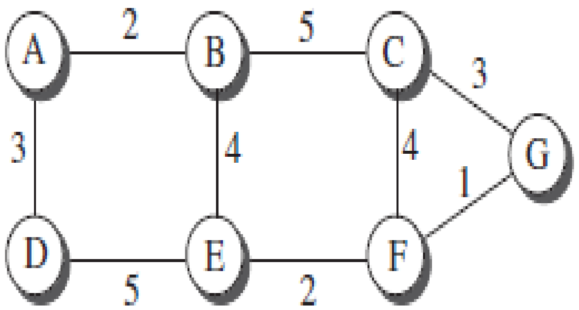
- A shortest path tree is a tree in which the path between the root and every other node is the shortest. What we need for each node is a shortest path tree with that node as the root.



a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

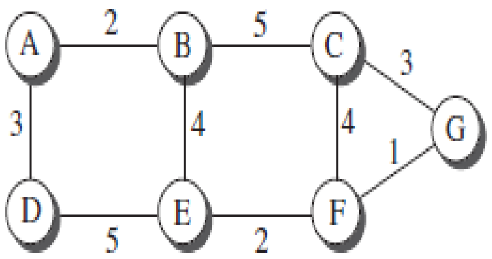
b. Link state database


a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

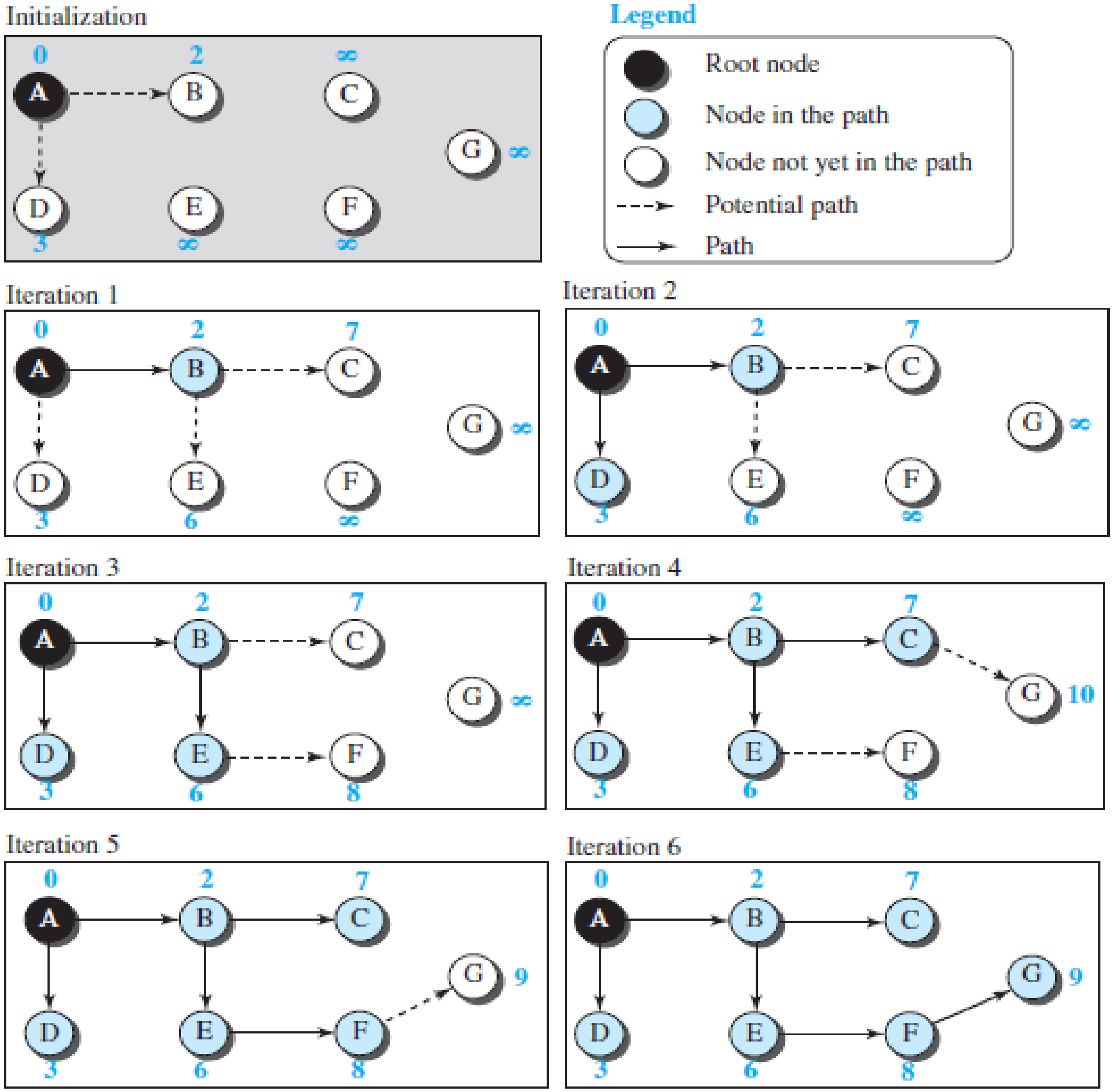
- The Dijkstra algorithm creates a shortest path tree from a graph. The algorithm divides the nodes into two sets: tentative and permanent. It finds the neighbours of a current node, makes them tentative, examines them, and if they pass the criteria, makes them permanent.
- To find the shortest path in each step, we need the cumulative cost from the root to each node, which is shown next to the node. Nodes and lists with the cumulative costs.



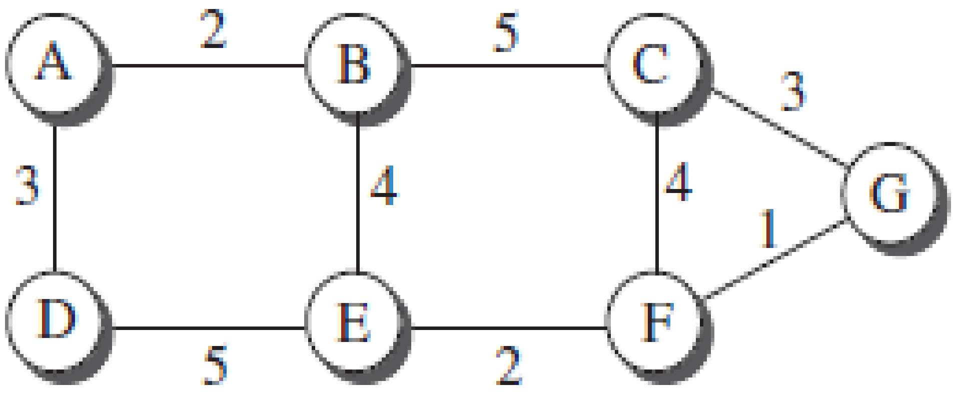
a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database



- The collection of state for all links is called the ***link-state database (LSDB).***
- There is only one LSDB for the whole internet; each node needs to have a duplicate of it to be able to create the least-cost tree.
- The LSDB can be represented as a two-dimensional array (matrix) in which the value of each cell defines the cost of the corresponding link.

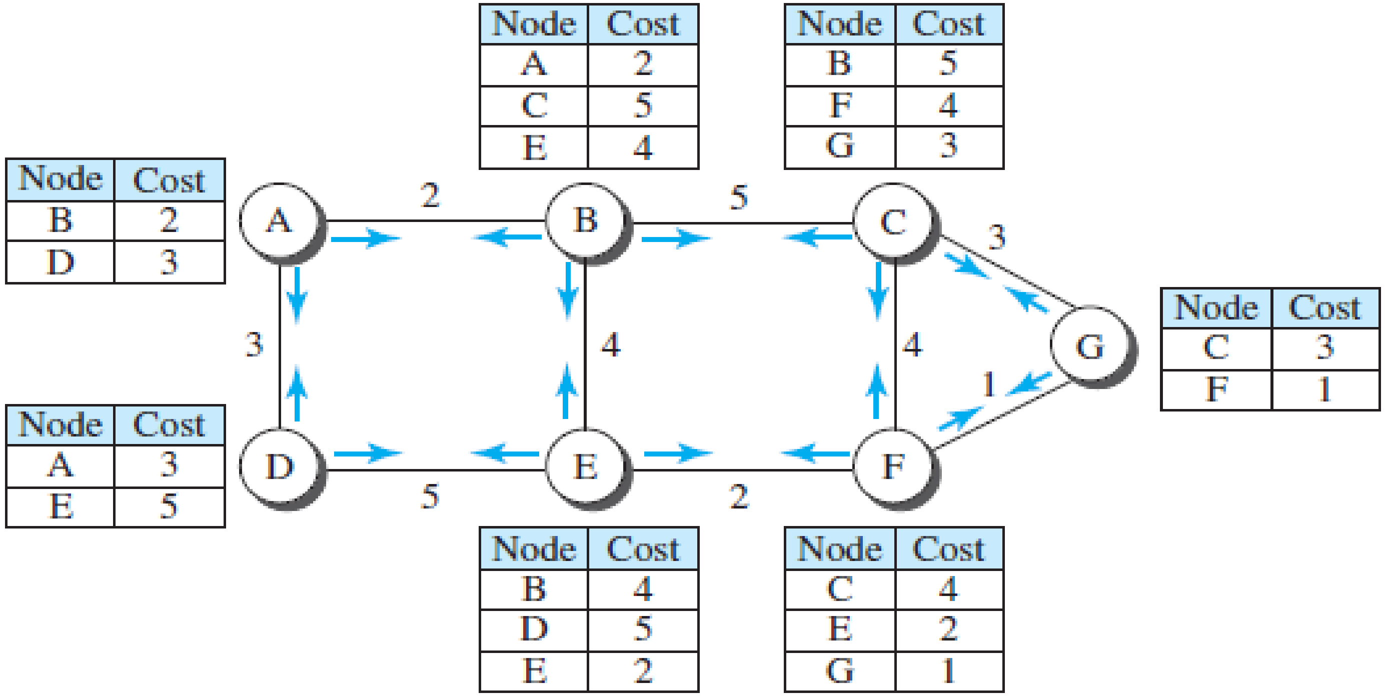


a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

- Each node can create this LSDB that contains information about the whole internet by a process called ***flooding.***
- Each node sends some greeting messages to all its immediate neighbours (those nodes to which it is connected directly) to collect two pieces of information: ***the identity of the node*** and ***the cost of the link.***
- The combination of these two pieces of information is called the *LS packet* (LSP); the LSP is sent out of each interface.



Node	Cost
B	2
D	3

Node	Cost
A	2
C	5
E	4

Node	Cost
B	5
F	4
G	3

Node	Cost
A	3
E	5

Node	Cost
B	4
D	5
E	2

Node	Cost
C	4
E	2
G	1

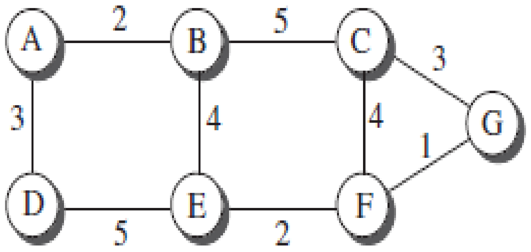
Node	Cost
C	3
F	1

- When a node receives an LSP from one of its interfaces, it compares the LSP with the copy it may already have.
  - If the newly arrived LSP is older than the one it has (found by checking the sequence number), it discards the LSP.
  - If it is newer or the first one received, the node discards the old LSP (if there is one) and keeps the received one.
  - It then sends a copy of it out of each interface except the one from which the packet arrived.

Break

Formation of Least-Cost Trees

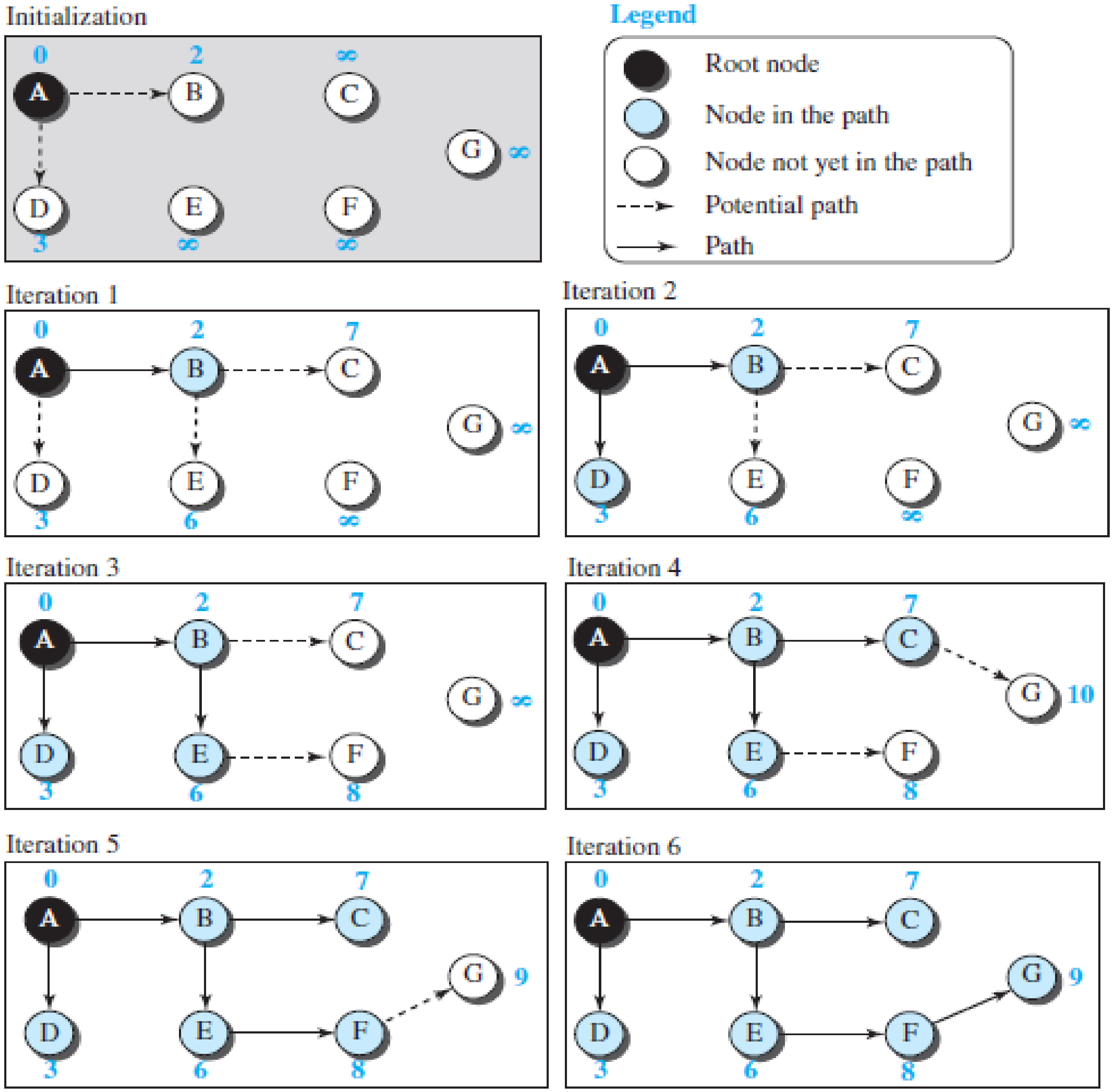
- To create a least-cost tree for itself, using the shared LSDB, each node needs to run the **Dijkstra Algorithm.** This iterative algorithm uses the following steps:
  1. The node chooses itself as the root of the tree, creating a tree with a single node, and sets the total cost of each node based on the information in the LSDB.
  2. The node selects one node, among all nodes not in the tree, which is closest to the root, and adds this to the tree. After this node is added to the tree, the cost of all other nodes not in the tree needs to be updated because the paths may have been changed.
  3. The node repeats step 2 until all nodes are added to the tree



a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database



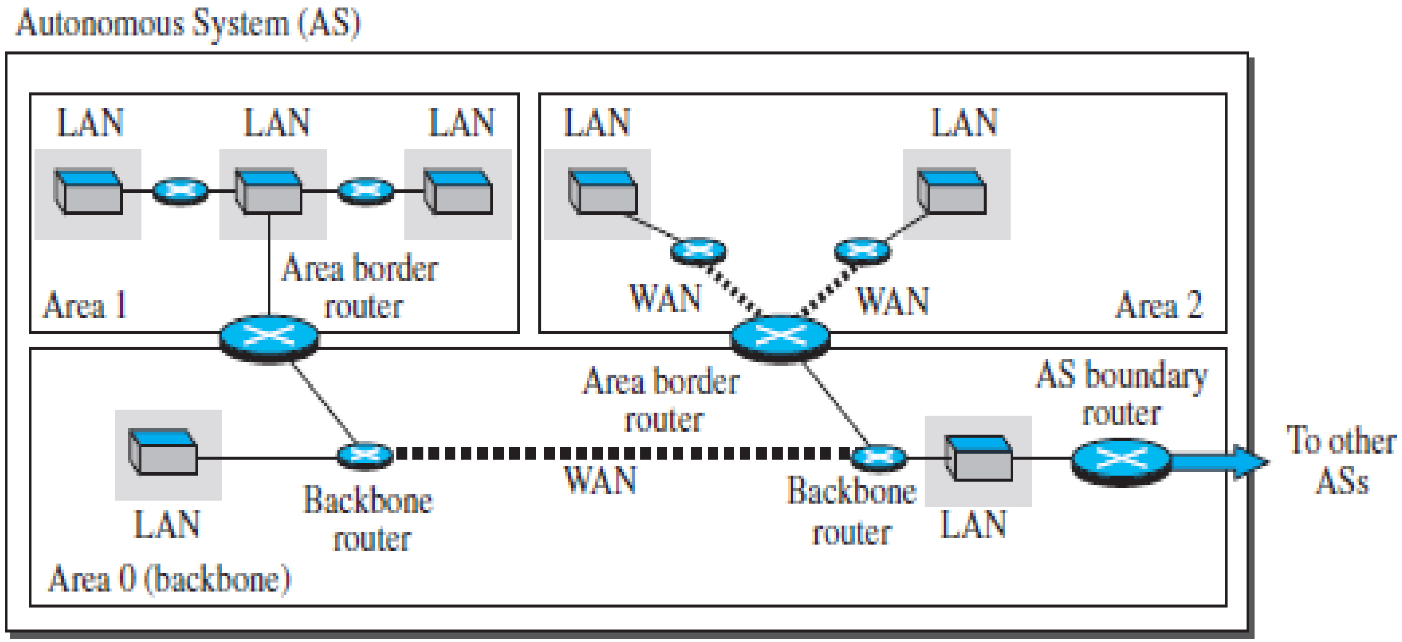
Break

Open Shortest Path First (OSPF)

- **Open Shortest Path First** (**OSPF**) is also an intradomain routing protocol like RIP, but it is based on the link-state routing protocol
- ***Metric***
  - The cost of reaching a destination from the host is calculated from the source router to the destination network.
  - An interesting point about the cost in OSPF is that different service types (TOSs) can have different weights as the cost.
  - Each link (network) can be assigned a weight based on the throughput, round-trip time, reliability, and so on.

Areas

- RIP is normally used in small ASs, OSPF was designed to handle routing in a small or large autonomous system.
- In Large AS's, as OSPF creates a LSDB by flooding this can lead to creation of huge traffic in network, to deal with this a large AS is divided into small ***areas.***
- Each area acts as a small independent domain for flooding LSPs.



- Each router in an area needs to know the information about the link states not only in its area but also in other areas.
- For this reason, one of the areas in the AS is designated as the *backbone area,* responsible for gluing the areas together.
- The routers in the backbone area are responsible for passing the information collected by each area to all other areas

OSPF Implementation

- OSPF is implemented as a program in the network layer, using the service of the IP for propagation.

Performance

- ***Update Messages.*** The link-state messages in OSPF have a somewhat complex format. They also are flooded to the whole area. If the area is large, these messages may create heavy traffic and use a lot of bandwidth.
- ***Convergence of Forwarding Tables***. When the flooding of LSPs is completed, each router can create its own shortest-path tree and forwarding table; convergence is fairly quick
- ***Robustness.*** The OSPF protocol is more robust than RIP because Corruption or failure in one router does not affect other router as seriously as in RIP.

Break

Q Consider the following statements about the routing protocols, Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) in an IPv4 network.

I. RIP uses distance vector routing

II. RIP packets are sent using UDP

III. OSPF packets are sent using TCP

IV. OSPF operation is based on link-state routing

Which of the following above are CORRECT? **(Gate-2017) (1 Marks)**

(A) I and IV only

(B) I, II and III only

(C) I, II and IV only

(D) II, III and IV only

Q Which one of the following is TRUE about interior Gateway routing protocols – Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) **(GATE-2014) (1 Marks)**

(A) RIP uses distance vector routing and OSPF uses link state routing

(B) OSPF uses distance vector routing and RIP uses link state routing

(C) Both RIP and OSPF use link state routing

(D) Both RIP and OSPF use distance vector routing

Q Consider the following three statements about link state and distance vector routing protocols, for a large network with 500 network nodes and 4000 links.

[S1] The computational overhead in link state protocols is higher than in distance vector protocols.
[S2] A distance vector protocol (with split horizon) avoids persistent routing loops, but not a link state protocol.
[S3] After a topology change, a link state protocol will converge faster than a distance vector protocol.

Which one of the following is correct about S1, S2, and S3? **(GATE-2014) (1 Marks)**

(A) S1, S2, and S3 are all true.

(B) S1, S2, and S3 are all false.

(C) S1 and S2 are true, but S3 is false

(D) S1 and S3 are true, but S2 is false

Q Two popular routing algorithms are Distance Vector (DV) and Link State (LS) routing. Which of the following are true? **(GATE-2008) (2 Marks)**

(S_1) Count to infinity is a problem only with DV and not LS routing

(S_2) In LS, the shortest path algorithm is run only at one node

(S_3) In DV, the shortest path algorithm is run only at one node

(S_4) DV requires lesser number of network messages than LS

(A) S_1, S_2 and S_4 only

(B) S_1, S_3 and S_4 only

(C) S_2 and S_3 only

(D) S_1 and S_4 only

Break

Difference between DVR and LSR

• In the distance-vector routing algorithm, each router tells its neighbours what it knows about the whole internet.	• In the link-state routing algorithm, each router tells the whole internet what it knows about its neighbours.
• Was most popularly used around 1980's	• Was most popularly used around 1990's
• Based on the idea of Local Knowledge	• Based on the idea of Global Knowledge
• Bandwidth requirement is Less	• Bandwidth requirement is high
• Roughly based on the idea of Bellman-Ford Algo	• Directly based on the idea of Dijkstra's
• Traffic is usually less	• Traffic is Usually high
• Converge slowly	• Converge faster
• Counts to Infinity	• No Counts to infinity
• RIP	• OSPF

Routing Table

- A table is maintained by the internal router called as **Routing table**.
- It helps the internal router to decide on which interface the data packet should be forwarded.
- Routing table consists of:
    - IP Address of the destination subnet
    - Subnet mask of the subnet
    - Interface

How routing is done in subnets

- When a data packet arrives at an internal router, the following steps are followed:
  1. Router performs the ***bitwise ANDing of Destination IP Address mentioned on the data packet and all the subnet masks one by one.***
  2. Router compares each result with their corresponding IP Address of the destination subnet in the routing table.
  3. Then, following three cases may occur:
     1. If there occurs only one match, Router forwards the data packet on the corresponding interface.
     2. If there occurs more than one match, Router forwards the data packet on the interface corresponding to the longest subnet mask.
     3. If there occurs no match, Router forwards the data packet on the interface corresponding to the default entry.

Points to Note

- In fixed length subnetting, since all the subnets have the same subnet mask. Bitwise ANDing is performed only once.
- If the result matches to any of the destination subnet IP Address, Router forwards the data packet on its corresponding interface. Otherwise, it is forwarded on the default interface.
- In variable length subnetting, all the subnets have different subnet mask. So, bitwise ANDing is performed once with each subnet mask.
- Then, the above three cases are followed.

Q Classless Inter-Domain Routing (CIDR) receives a packet with address 131.23.151.76. The router's routing table has the following entries: **(Gate-2014) (2 Marks)**

Prefix	Output Interface Identifier
131.16.0.0/12	3
131.28.0.0/14	5
131.19.0.0/16	2
131.22.0.0/15	1

The identifier of the output interface on which this packet will be forwarded is ______.

Q Classless Inter-Domain Routing (CIDR) receives a packet with address 131.23.151.76. The router's routing table has the following entries: **(Gate-2014) (2 Marks)**

Prefix	Output Interface Identifier
131.16.0.0/12	3
131.28.0.0/14	5
131.19.0.0/16	2
131.22.0.0/15	1

The identifier of the output interface on which this packet will be forwarded is _______.

131.23.151.76 255.240.0.0 131.16.0.0	10000011 00010111 10010111 01001100 11111111 11110000 00000000 00000000 10000011 00010000 00000000 00000000
131.23.151.76 255.252.0.0 131.20.0.0	10000011 00010111 10010111 01001100 11111111 11111100 00000000 00000000 10000011 00010100 00000000 00000000
131.23.151.76 255.255.0.0 131.23.0.0	10000011 00010111 10010111 01001100 11111111 11111111 00000000 00000000 10000011 00010111 00000000 00000000
131.23.151.76 255.254.0.0 131.22.0.0	10000011 00010111 10010111 01001100 11111111 11111110 00000000 00000000 10000011 00010110 00000000 00000000

Q A router uses the following routing table:

Destination	Mask	Interface
144.16.0.0	255.255.0.0	eth0
144.16.64.0	255.255.224.0	eth1
144.16.68.0	255.255.255.0	eth2
144.16.68.64	255.255.255.224	eth3

A packet bearing a destination address 144.16.68.117 arrives at the router.
On which interface will it be forwarded?
(Gate-2006) (2 Marks)
(A) eth0
(B) eth1
(C) eth2
(D) eth3

Q A router uses the following routing table:

Destination	Mask	Interface
144.16.0.0	255.255.0.0	eth0
144.16.64.0	255.255.224.0	eth1
144.16.68.0	255.255.255.0	eth2
144.16.68.64	255.255.255.224	eth3

A packet bearing a destination address 144.16.68.117 arrives at the router. On which interface will it be forwarded?

(Gate-2006) (2 Marks)

(A) eth0

(B) eth1

(C) eth2

(D) eth3

144.16.68.117 255.255.0.0 144.16.0.0	10010000 00010000 01000100 01110101 11111111 11111111 00000000 00000000 10010000 00010000 00000000 00000000
144.16.68.117 255.255.224.0 144.16.64.0	10010000 00010000 01000100 01110101 11111111 11111111 11100000 00000000 10010000 00010000 01000000 00000000
144.16.68.117 255.255.255.0 144.16.68.0	10010000 00010000 01000100 01110101 11111111 11111111 11111111 00000000 10010000 00010000 01000100 00000000
144.16.68.117 255.254.255.224 144.16.68.96	10010000 00010000 01000100 01110101 11111111 11111111 11111111 11100000 10010000 00010000 10000100 01100000

Q The routing table of a router is shown below:

Destination	Sub net mask	Interface
128.75.43.0	255.255.255.0	Eth0
128.75.43.0	255.255.255.128	Eth1
192.12.17.5	255.255.255.255	Eth3
default		Eth2

On which interfaces will the router forward packets have addressed to destinations 128.75.43.16 and 192.12.17.10 respectively? **(Gate-2004) (2 Marks)**

(A) Eth1 and Eth2

(B) Eth0 and Eth2

(C) Eth0 and Eth3

(D) Eth1 and Eth3

Q The routing table of a router is shown below:

Destination	Sub net mask	Interface
128.75.43.0	255.255.255.0	Eth0
128.75.43.0	255.255.255.128	Eth1
192.12.17.5	255.255.255.255	Eth3
default		Eth2

On which interfaces will the router forward packets have addressed to destinations 128.75.43.16 and 192.12.17.10 respectively? **(Gate-2004) (2 Marks)**

(A) Eth1 and Eth2

(B) Eth0 and Eth2

(C) Eth0 and Eth3

(D) Eth1 and Eth3

128.75.43.16 255.255.255.0 128.75.43.0	10000000 01001011 00101011 00010000 11111111 11111111 11111111 00000000 10000000 01001011 00101011 00000000
128.75.43.16 255.255.255.128 128.75.43.0	10000000 01001011 00101011 00010000 11111111 11111111 11111111 10000000 10000000 01001011 00101011 00000000
128.75.43.16 255.255.255.255 128.75.43.16	10000000 01001011 00101011 00010000 11111111 11111111 11111111 11111111 10000000 01001011 00101011 00010000

192.12.17.10 255.255.255.0 192.12.17.0	11000000 00001100 00010001 00001010 11111111 11111111 11111111 00000000 11000000 00001100 00010001 00000000
192.12.17.10 255.255.255.128 192.12.17.0	11000000 00001100 00010001 00001010 11111111 11111111 11111111 10000000 11000000 00001100 00010001 00000000
192.12.17.10 255.255.255.255 192.12.17.10	11000000 00001100 00010001 00001010 11111111 11111111 11111111 11111111 11000000 00001100 00010001 00001010

Q A group of 15 routers is interconnected in a centralized complete binary tree with a router at each tree node. Router i communicates with router j by sending a message to the root of the tree. The root then sends the message back down to router j. The mean number of hops per message, assuming all possible router pairs are equally likely is **(GATE-2007) (2 Marks)**

A) 3 **B)** 4.26 **C)** 4.53 **D)** 5.26