ROUTING PROTOCOLS

MODULE 5

IP ROUTING

Static Routing

Dynamic Routing

Interior Gateway Protocol (IGP)

Exterior Gateway Protocol (EGP)

E.g BGP

Link state Routing Protocol

E.g OSPF, ISIS

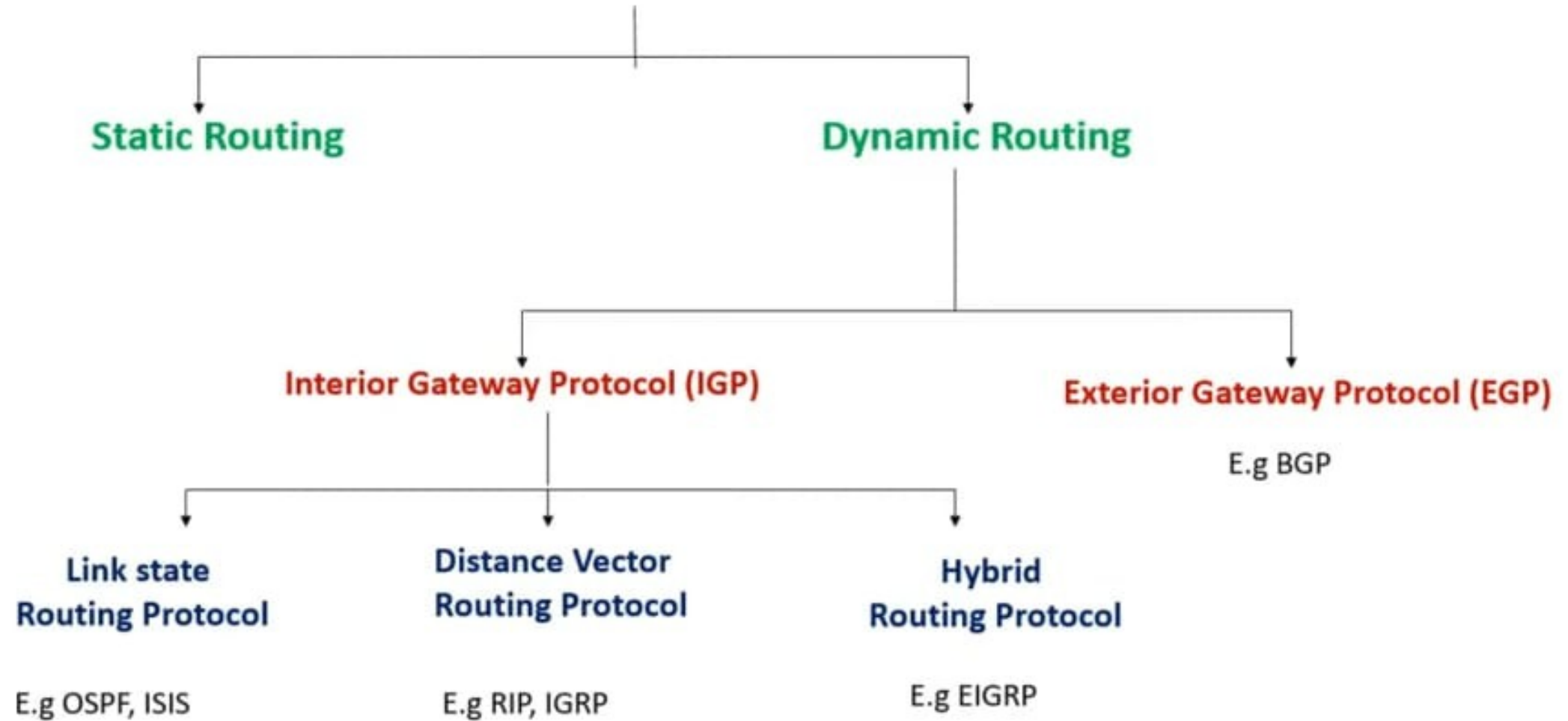**Distance Vector Routing Protocol**

E.g RIP, IGRP

Hybrid Routing Protocol

E.g EIGRP

Static Routing

Advantages Of Static Routing

Following are the advantages of Static Routing:

- **No Overhead:** It has ho overhead on the CPU usage of the router. Therefore, the cheaper router can be used to obtain static routing.
- **Bandwidth:** It has not bandwidth usage between the routers.
- **Security:** It provides security as the system administrator is allowed only to have control over the routing to a particular network.

Disadvantages of Static Routing:

Following are the disadvantages of Static Routing:

- For a large network, it becomes a very difficult task to add each route manually to the routing table.
- The system administrator should have a good knowledge of a topology as he has to add each route manually.

Dynamic Routing

- It is also known as Adaptive Routing.

- It is a technique in which a router adds a new route in the routing table for each packet in response to the changes in the condition or topology of the network.

- Dynamic protocols are used to discover the new routes to reach the destination.

- In Dynamic Routing, RIP and OSPF are the protocols used to discover the new routes.

- If any route goes down, then the automatic adjustment will be made to reach the destination.
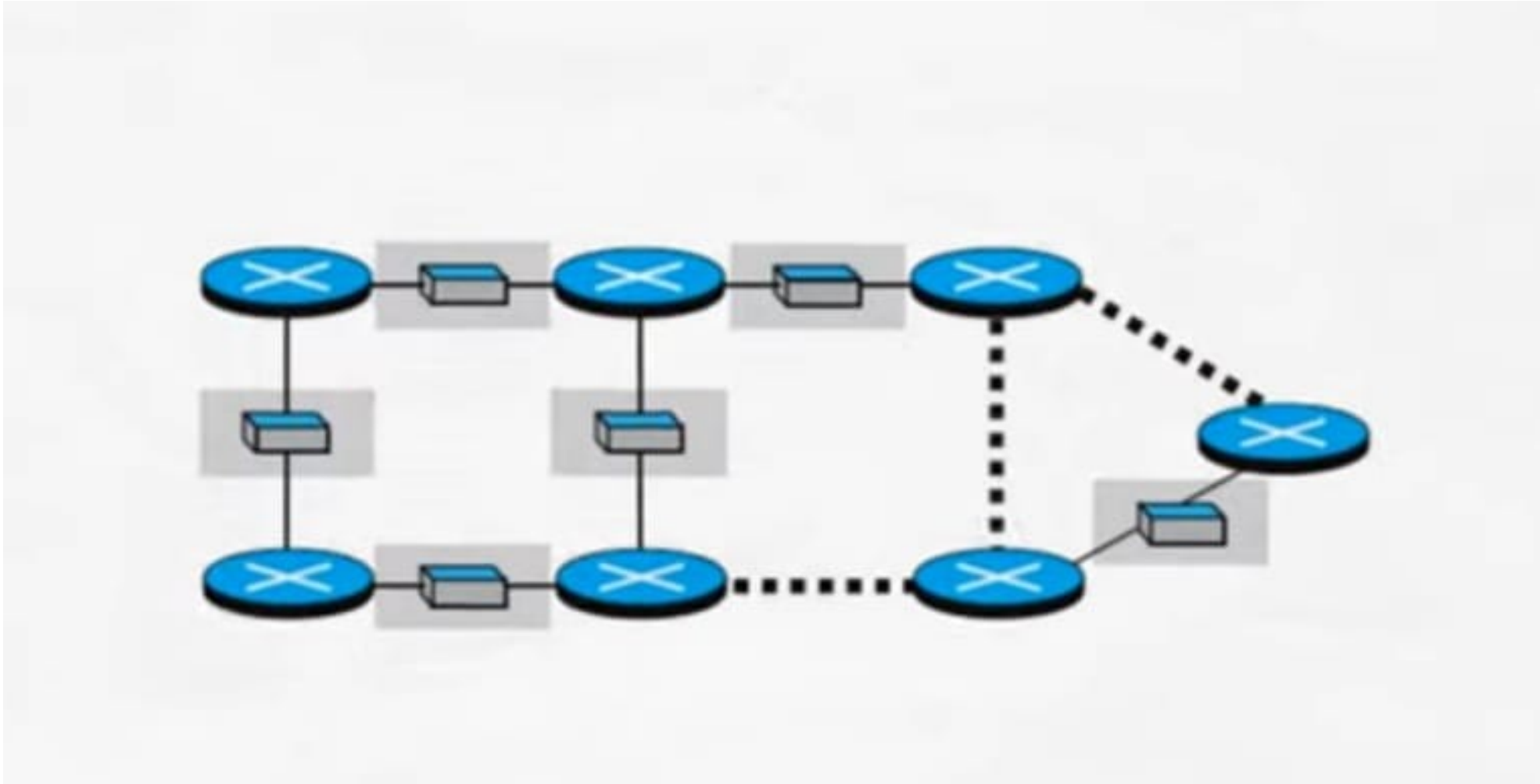
Dynamic Routing

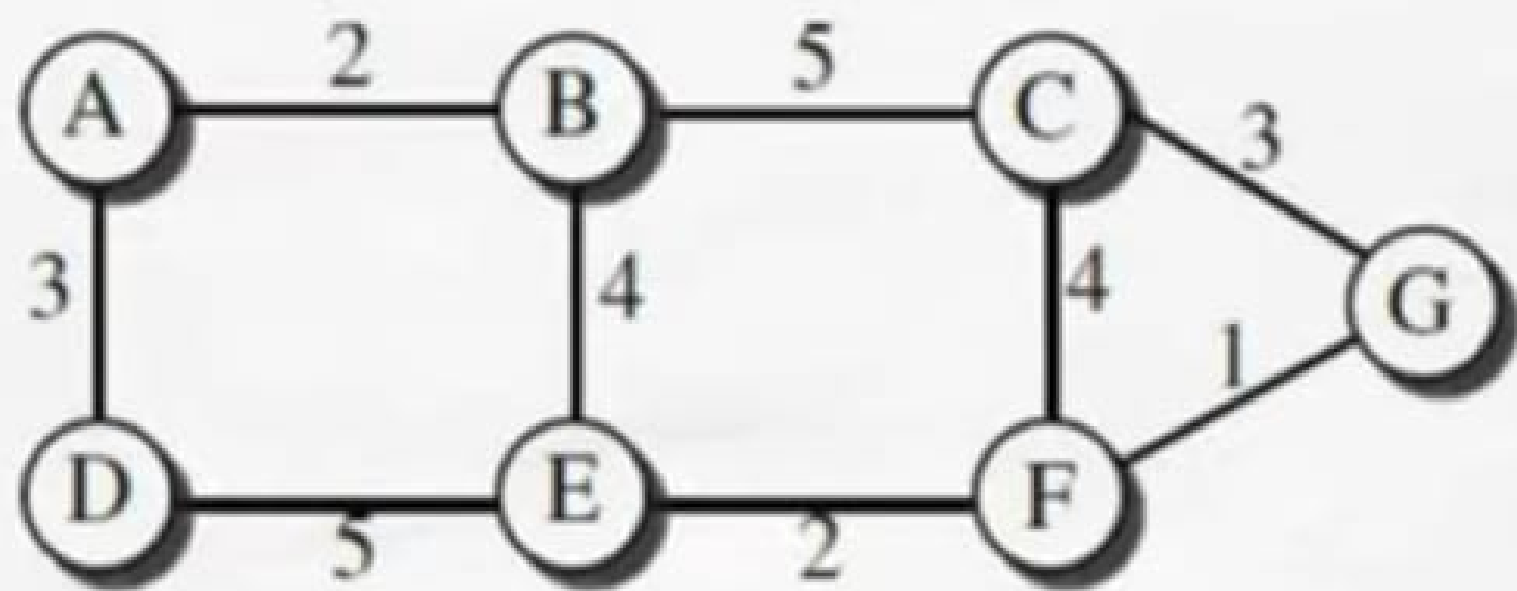**Advantages of Dynamic Routing:**

- It is easier to configure.

- It is more effective in selecting the best route in response to the changes in the condition or topology.

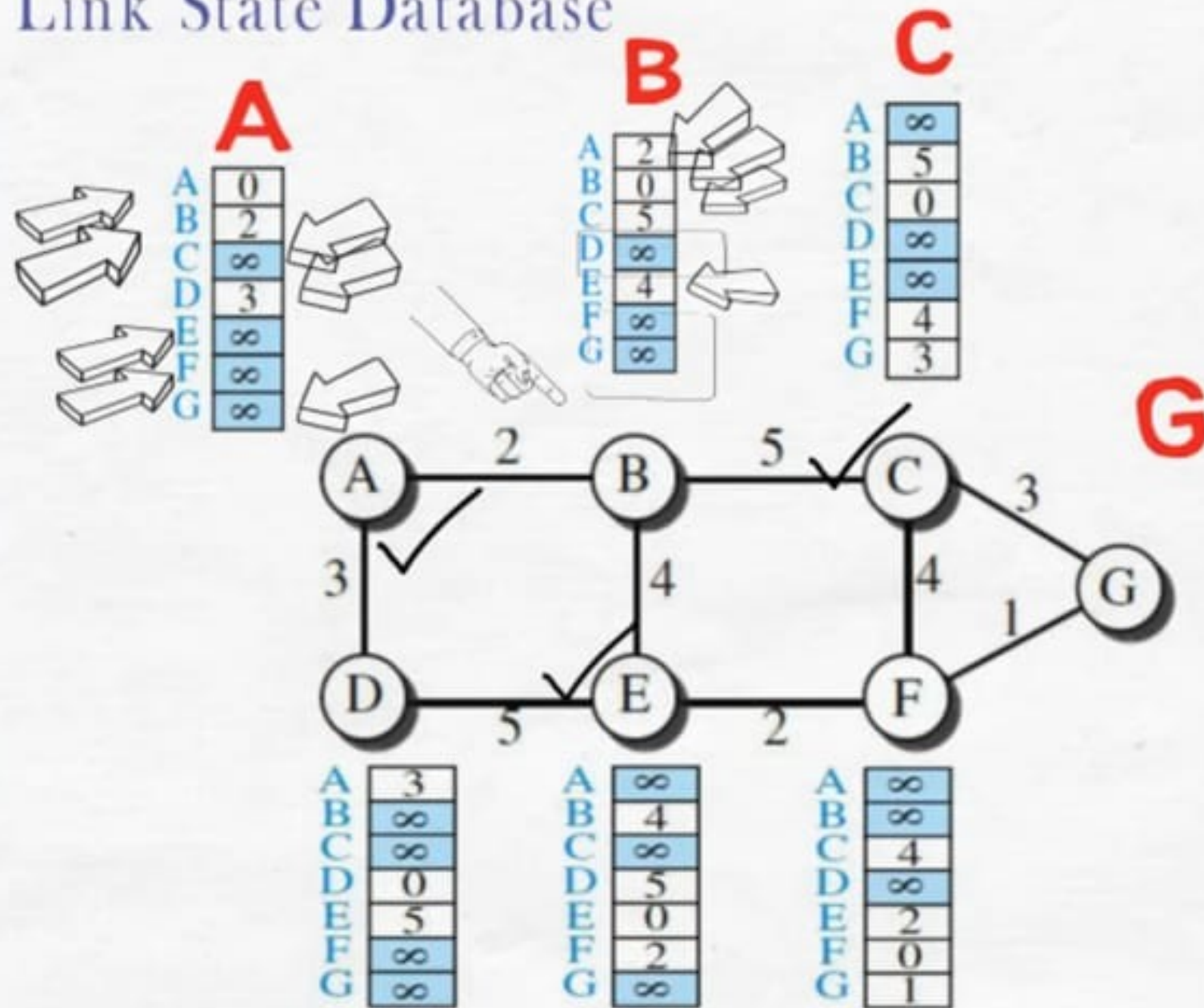**Disadvantages of Dynamic Routing:**

- It is more expensive in terms of CPU and bandwidth usage.

- It is less secure as compared to default and static routing.
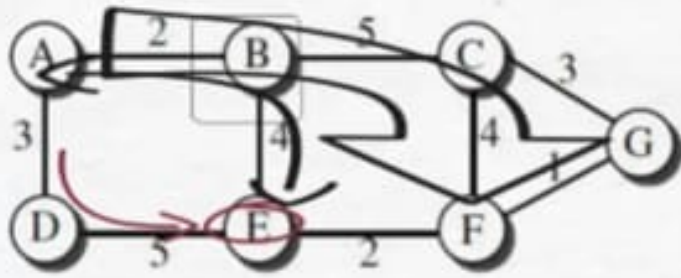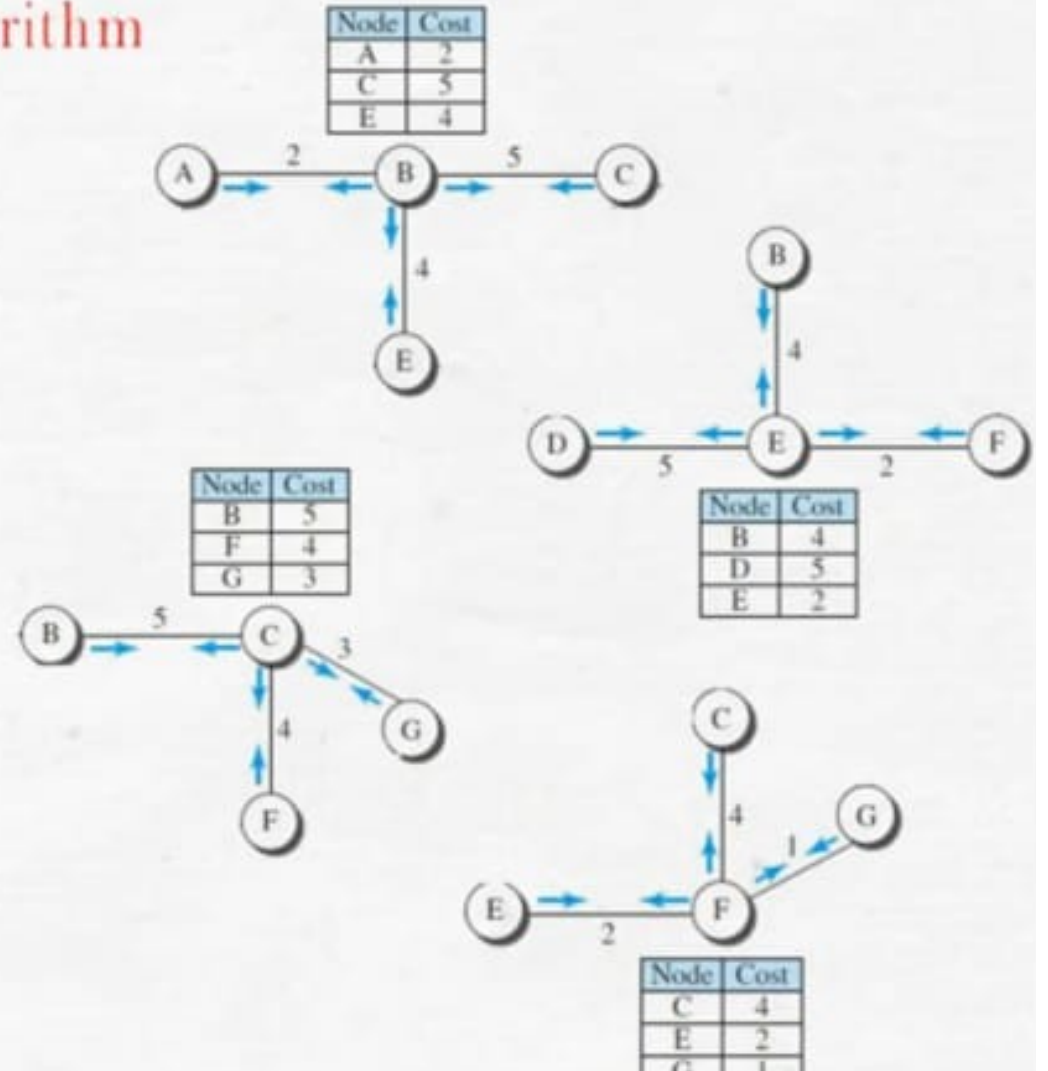
LINK STATE ROUTING PROTOCOL-

Link State Database

Graph with nodes A, B, C, D, E, F, G and edge weights: A–B = 2, A–D = 3, B–C = 5, B–E = 4, C–F = 4, C–G = 3, D–E = 5, E–F = 2, F–G = 1, G–C = 3.

Node A vector:

A	0
B	2
C	∞
D	3
E	∞
F	∞
G	∞

Node B vector:

A	2
B	0
C	5
D	∞
E	4
F	∞
G	∞

Node C vector:

A	∞
B	5
C	0
D	∞
E	∞
F	4
G	3

Node D vector:

A	3
B	∞
C	∞
D	0
E	5
F	∞
G	∞

Node E vector:

A	∞
B	4
C	∞
D	5
E	0
F	2
G	∞

Node F vector:

A	∞
B	∞
C	4
D	∞
E	2
F	0
G	1

Node G vector:

A	∞
B	∞
C	3
D	∞
E	∞
F	1
G	0

Full Link State Database matrix:

	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

Formation of Least-Cost Trees

Using the shared **LSDB**, each node needs to run the Dijkstra Algorithm

Permanent	Tentative
–	A(0)
A(0)	B(2) D(3)
A(0) B(2)	D(3) C(7)E(6)
A(0) B(2) D(3)	C(7)E(6)
A(0) B(2) D(3) E(6)	C(7) F(8)
A(0) B(2) C(7) D(3) E(6)	F(8) G(9)
A(0) B(2) C(7) D(3) E(6) F(8)	G(9)

Node	Cost
A	2
C	5
E	4

Node	Cost
B	5
F	4
G	3

Node	Cost
B	4
D	5
E	2

Node	Cost
C	4
E	2
G	1

Formation of Least-Cost Trees

Permanent	Tentative
-	A (0)
A (0)	B (2) D (3)
A (0) B (2)	D (3) C (7) E (6)
A (0) B (2) D (3)	C (7) E (6)
A (0) B (2) D (3) E (6)	C (7) F (8)
A (0) B (2) C (7) D (3) E (6)	F (8) G (9)
A (0) B (2) C (7) D (3) E (6) F (8)	G (9)

Tree for node A

A (0) B (2) C (7) D (3) E (6) F (8) G (9)
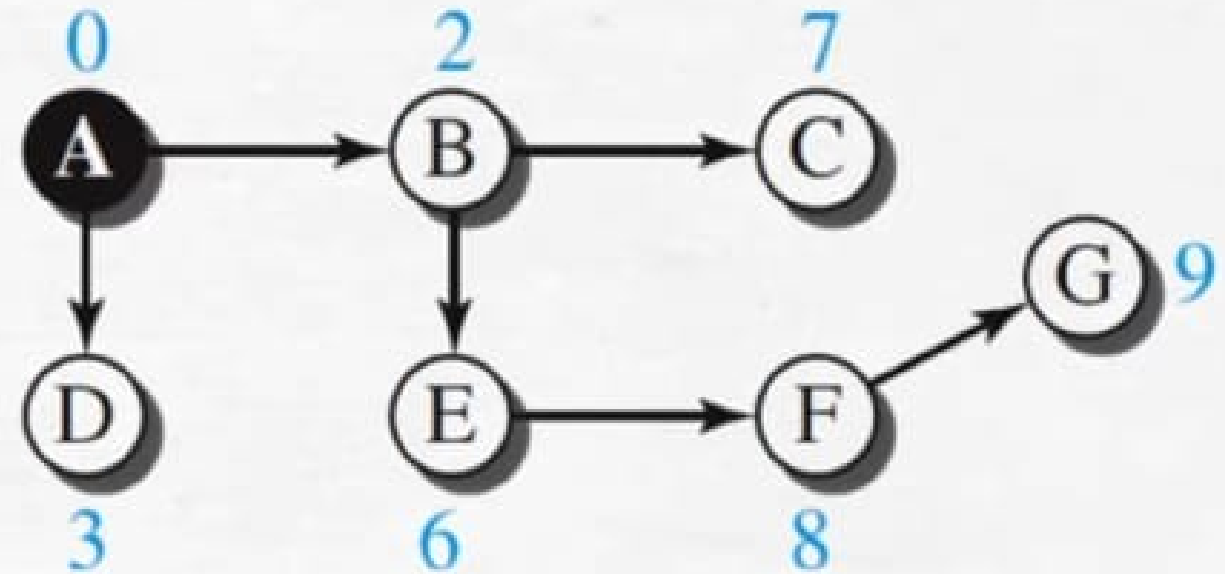
Build a routing table
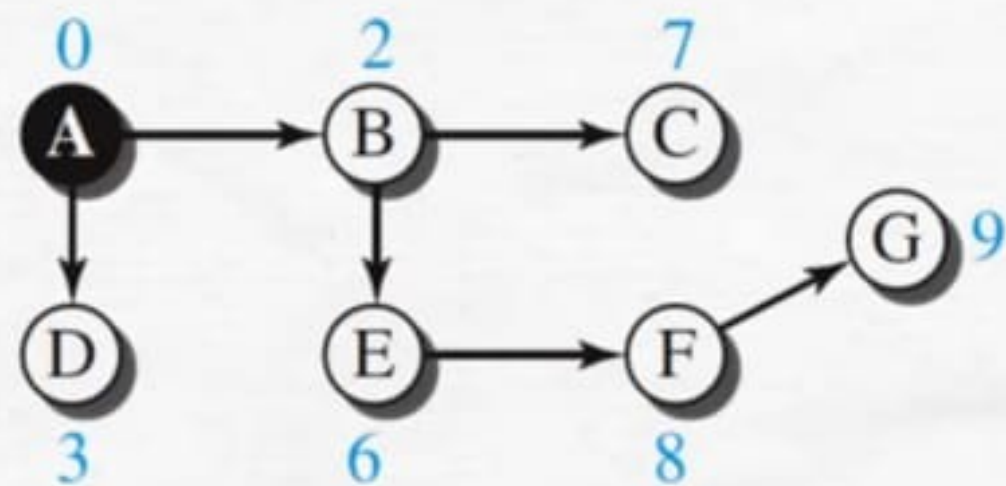
Tree for node A

Routing table of node A

DESTINATION	COST	NEXT HOP
A	0	-
B	2	-
C	7	B
D	3	-
E	6	B
F	8	B
G	9	B