CPSC 317 COMPUTER NETWORKING

Module 5: Network Layer – Day 5 – Distance Vector Routing

1

RECAP: LINK STATE

- Each router tells every other router about the state of all of its directly connected links
- This gives every router complete information about the entire network
- Every so often, each router uses Dijkstra's algorithm to find the shortest path to all the other routers
- It then updates its forwarding table

LINK STATE COST

- N is the number of routers in the network
- L is the number of links in the network
- Communication cost:
  - O(N^2) messages to broadcast link state information from every router to every other router
- Routing update cost (Dijkstra's algorithm):
  - Depends on the representation of the graph, datastructures for minimum search
  - O(N^2) with graph as adjacency matrix and linear search
  - O(L + N log N) with graph as adjacency list of edges and using Fibonacci heap

LEARNING GOALS

Intra-domain routing II

- Explain what distance vector routing is
- Execute the distance vector routing algorithm on a small graph of routers
- Contrast link state and distance vector routing

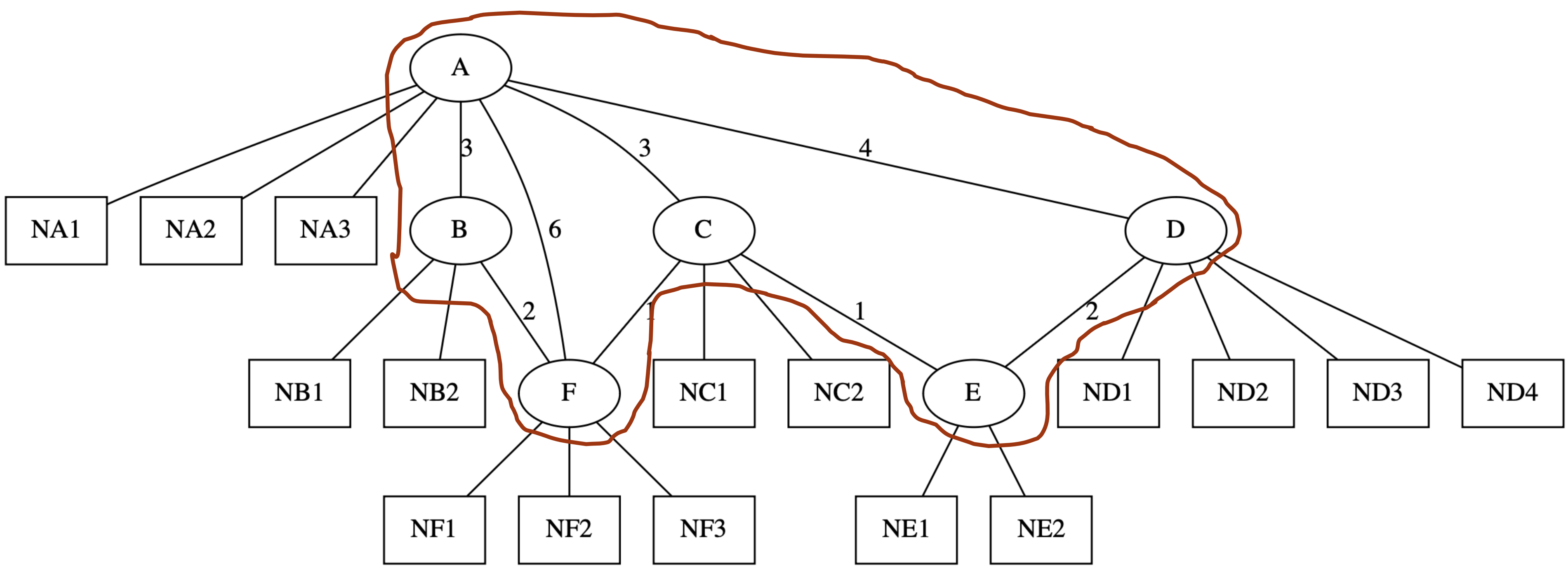
READING

- Reading: 5.1, 5.2, 5.2.1, 5.2.2

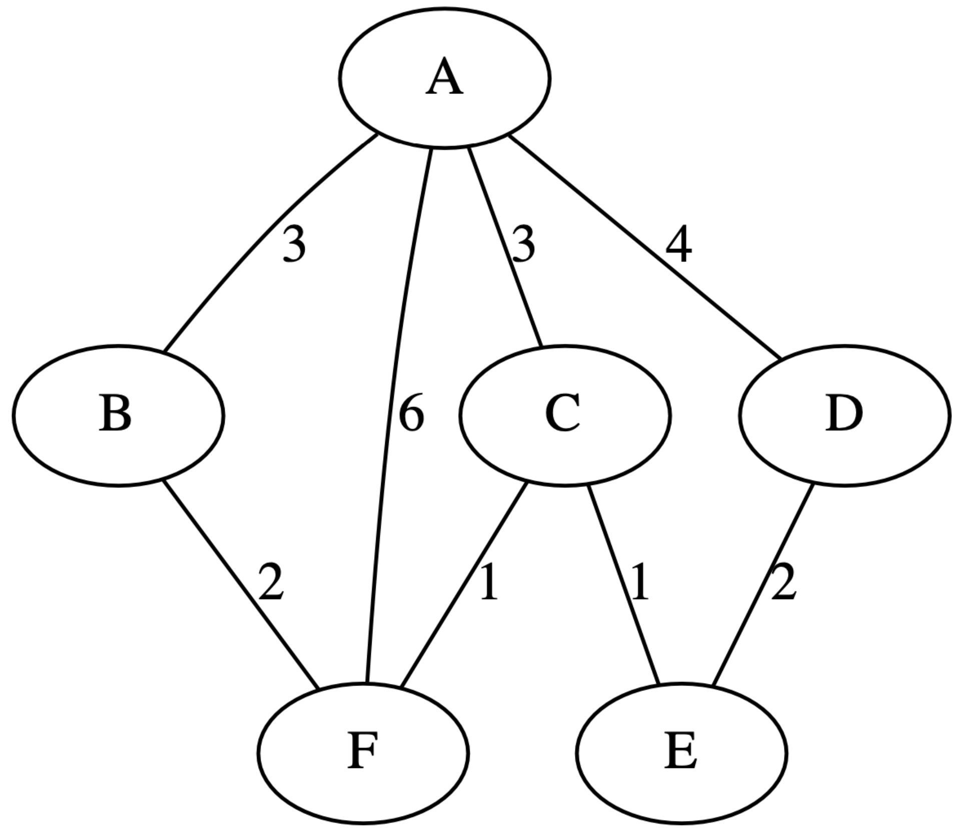
DISTANCE VECTOR

- Every so often, each router (A) tells its neighbours about the cost of its best routes to the networks it knows about
- A receiving router (B) checks to see if any of these routes would shorten their path to the destination
- If so, it updates its forwarding table to forward through A
- Like previous class, we'll use the router as a proxy for the "stub" networks that are connected to it

ROUTERS AS PROXIES FOR NETWORKS



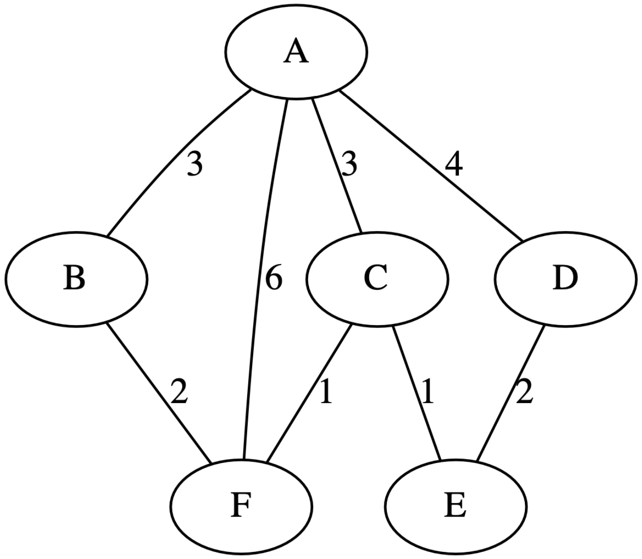
AN EXAMPLE NETWORK GRAPH



DISTANCE VECTOR – A'S VIEW

	A	B	C	D	E	F
A	0	3	3	4	-	6
B	-	-	-	-	-	-
C	-	-	-	-	-	-
D	-	-	-	-	-	-
E	-	-	-	-	-	-
F	-	-	-	-	-	-

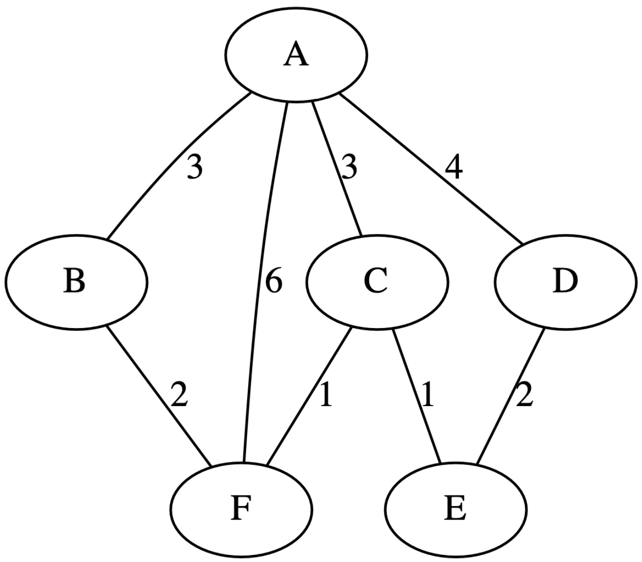
- Each row represents the view of the corresponding router
- Each column indicates least-cost path of the node from the router of the row



DISTANCE VECTOR – C'S VIEW

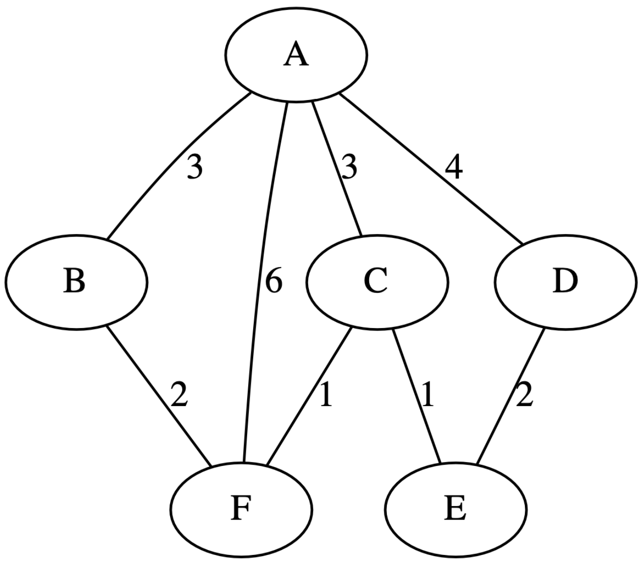
	A	B	C	D	E	F
A	-	-	-	-	-	-
B	-	-	-	-	-	-
C	3	-	0	-	1	1
D	-	-	-	-	-	-
E	-	-	-	-	-	-
F	-	-	-	-	-	-

- Each row represents the view of the corresponding router
- Each column indicates least-cost path of the node from the router of the row



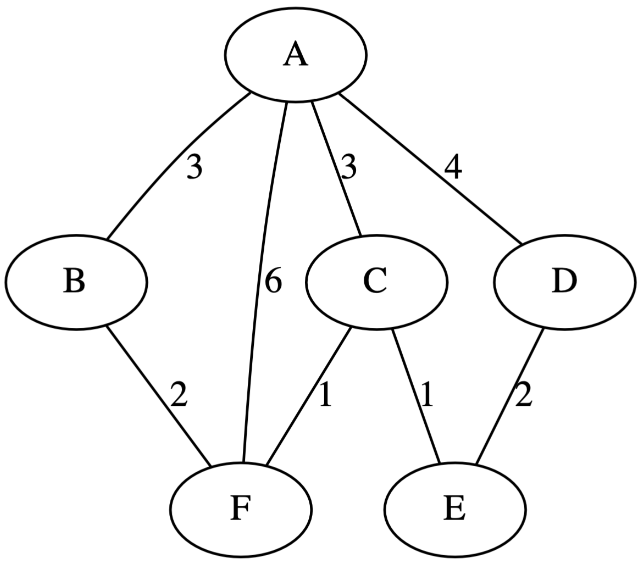
DISTANCE VECTOR – C SENDS TO A

	A	B	C	D	E	F
A	0	3	3	4	-	6
B	-	-	-	-	-	-
C	3	-	0	-	1	1
D	-	-	-	-	-	-
E	-	-	-	-	-	-
F	-	-	-	-	-	-



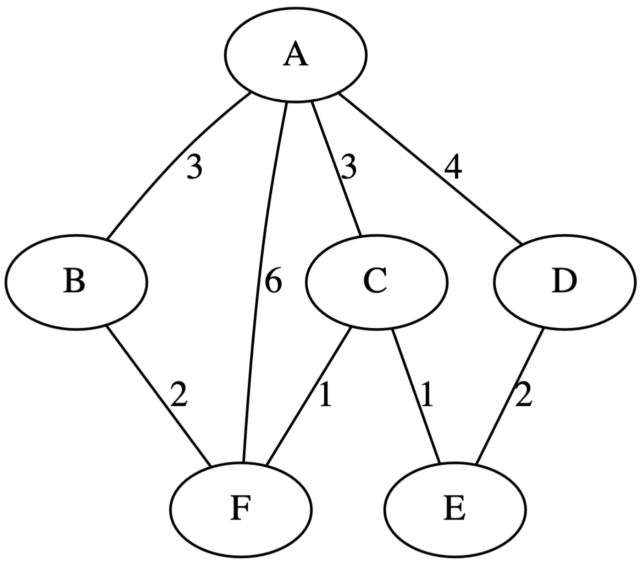
DISTANCE VECTOR – C SENDS TO A

	A	B	C	D	E	F
A	0	3	3	4	4 - C	6
B	-	-	-	-	-	-
C	3	-	0	-	1	1
D	-	-	-	-	-	-
E	-	-	-	-	-	-
F	-	-	-	-	-	-



DISTANCE VECTOR – C SENDS TO A

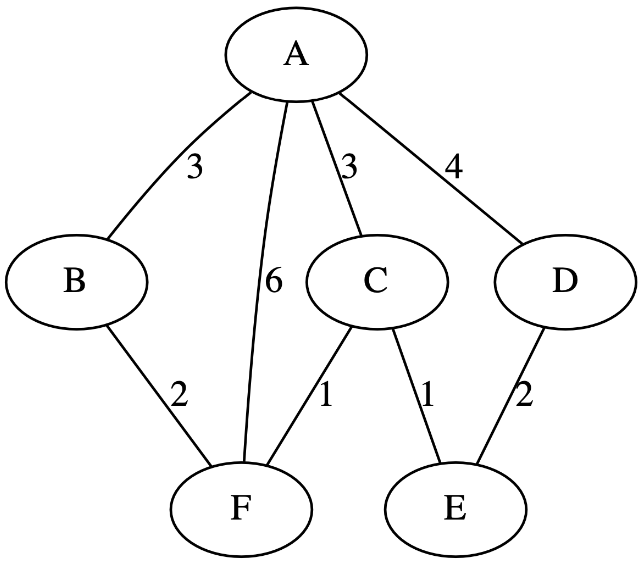
	A	B	C	D	E	F
A	0	3	3	4	4 - C	4 - C
B	-	-	-	-	-	-
C	3	-	0	-	1	1
D	-	-	-	-	-	-
E	-	-	-	-	-	-
F	-	-	-	-	-	-



DISTANCE VECTOR – INITIAL STATE

	A	B	C	D	E	F
A	0	3	3	4	-	6
B	3	0	-	-	-	2
C	3	-	0	-	1	1
D	4	-	-	0	2	-
E	-	-	1	2	0	-
F	6	2	1	-	-	0

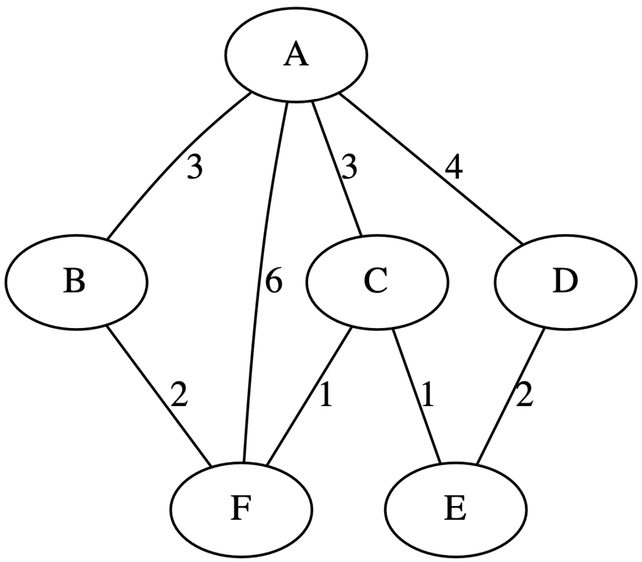
Each router sends (a subset) of its table to its neighbors



DISTANCE VECTOR – ROUND 1

	A	B	C	D	E	F
A	0	3	3	4	4 – C	4 – C
B	3	0	3 – F	7 – A	-	2
C	3	3 – F	0	3 – E	1	1
D	4	7 – A	3 – E	0	2	10-A
E	4 – C	-	1	2	0	2 - C
F	4 - C	2	1	10-A	2 - C	0

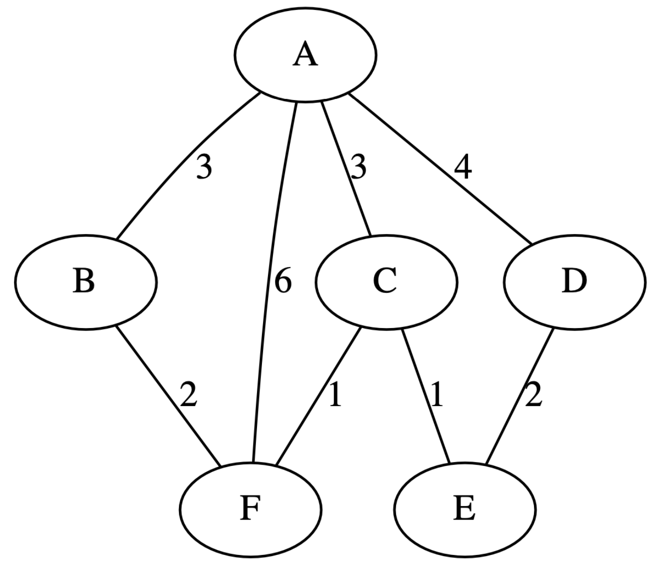
Each router updates its table based on least-cost paths from neighbors, then resend their tables



DISTANCE VECTOR – ROUND 2

	A	B	C	D	E	F
A	0	3	3	4	4 – C	4 – C
B	3	0	3 – F	7 – A	4 - F	2
C	3	3 – F	0	3 – E	1	1
D	4	7 – A	3 – E	0	2	4 - E
E	4 – C	4 - C	1	2	0	2 - C
F	4 - C	2	1	4 - C	2 - C	0

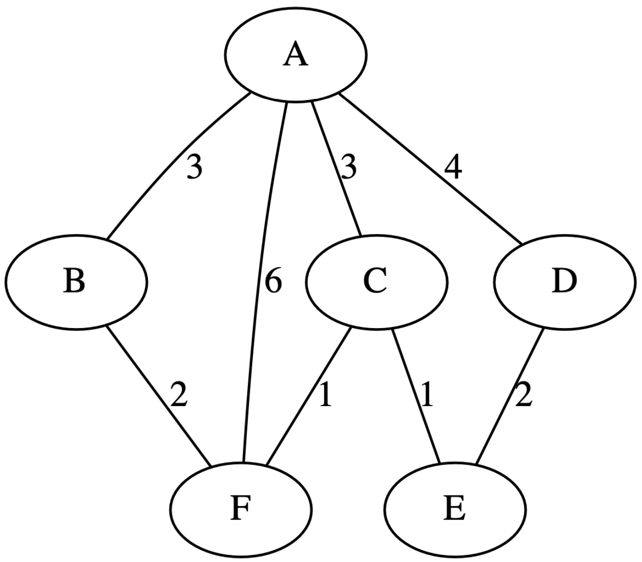
Each router updates its table based on least-cost paths from neighbors, then resend their tables



DISTANCE VECTOR – ROUND 3

	A	B	C	D	E	F
A	0	3	3	4	4 – C	4 – C
B	3	0	3 – F	6 - F	4 - F	2
C	3	3 – F	0	3 – E	1	1
D	4	6 - E	3 – E	0	2	4 - E
E	4 – C	4 - C	1	2	0	2 - C
F	4 - C	2	1	4 - C	2 - C	0

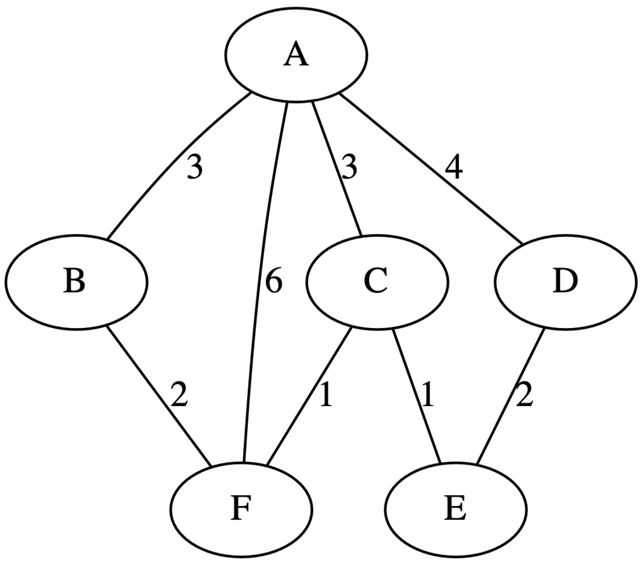
Each router updates its table based on least-cost paths from neighbors, then resend their tables



DISTANCE VECTOR – ROUND 4

	A	B	C	D	E	F
A	0	3	3	4	4 – C	4 – C
B	3	0	3 – F	6 - F	4 - F	2
C	3	3 – F	0	3 – E	1	1
D	4	6 - E	3 – E	0	2	4 - E
E	4 – C	4 - C	1	2	0	2 - C
F	4 - C	2	1	4 - C	2 - C	0

No more updates



CLICKER QUESTION

Assuming no links change cost, how many distance vector iterations (every router sending all its routes to all of its neighbours) does it take before the routes are guaranteed to converge?

A. Number of routers

B. Number of links

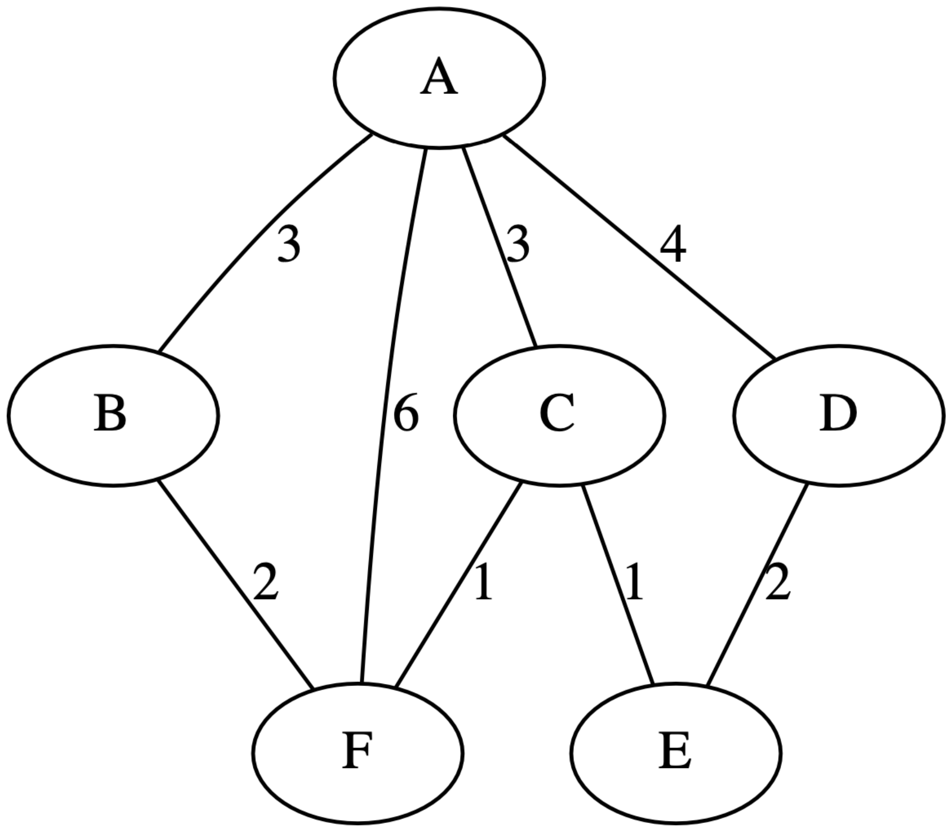
C. Diameter of the graph

D. None of the above

HOW DOES THIS COMPARE TO LINK STATE?

Done: A B C F D E

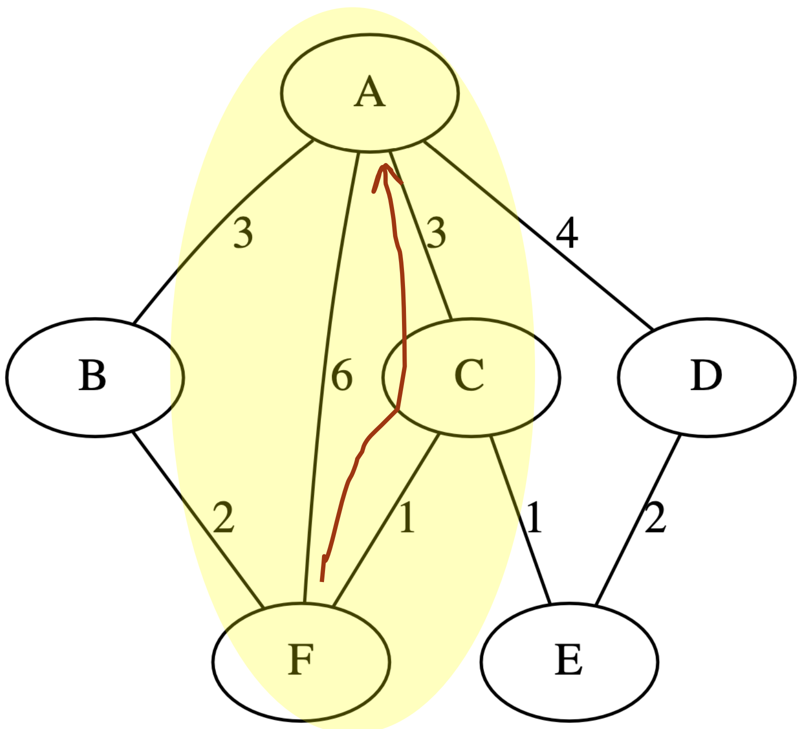
Dest	Cost	Prev
B	3	A
C	3	A
D	4	A
E	4	C
F	4	C

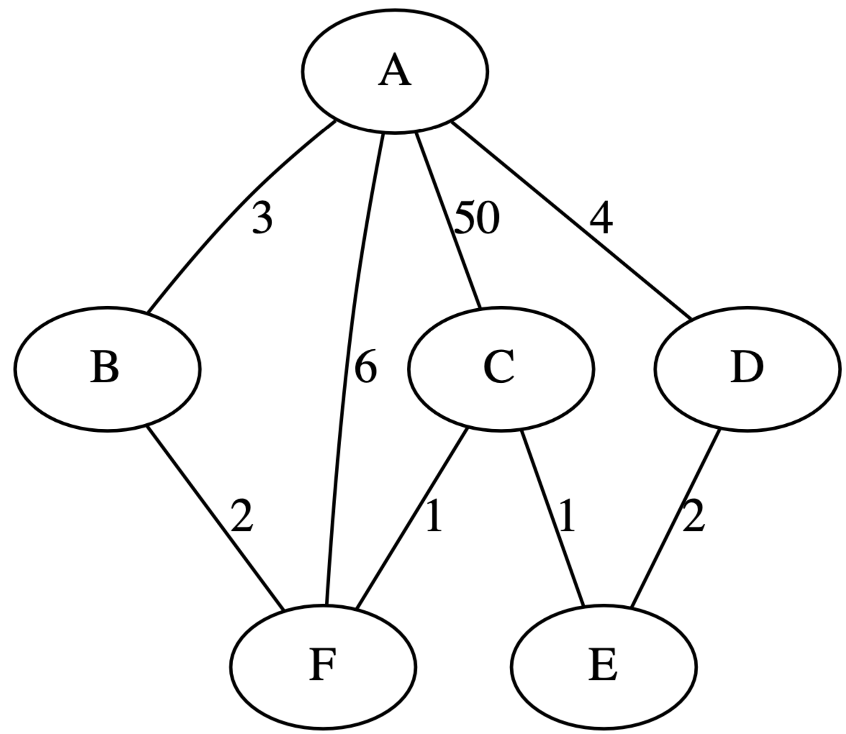


FORWARDING LOOPS

- If the cost of a link decreases, nothing bad happens
- But if the cost of a link increases, forwarding loops can be created

CREATING A FORWARDING LOOP





AVOIDING LOOPS – POISONED REVERSE

- If F routes to A through C, it advertises to C that its cost for A is infinity
- Works for loops involving 2 routers
- Doesn't work for loops involving more than 2 routers

OSPF (OPEN-SHORTEST-PATH-FIRST)

- Currently most used IGP in the Internet
- A link state protocol
- Supports a number of extensions
  - Areas – to support hierarchy (and improve scaling)
  - More scalable updates
- Runs over IP (protocol number 89)

INTERIOR GATEWAY PROTOCOLS

- Do not scale
- Do not account for administrative differences (administrative autonomy)
  - Political
  - Company
  - International boundaries
- Don’t allow policy to play a role
- Next class ...

IN-CLASS ACTIVITY

- Explore Distance Vector updates