COMP90007 Internet Technology

Week8

Yiran (Scott) Ruan

Email: yrrua@unimelb.edu.au

Web Page: https://yiranruan.github.io

Question 1

- A router has just received the following IP addresses:

57.6.96.0/21

57.6.104.0/21

57.6.112.0/21

57.6.120.0/21

- If all of them use the same outgoing line, can they be aggregated? If so, to what? If not, why not?

CIDR

- Before we talk about ’aggregate’, how about we introduce CIDR first.
- Problem: table explosion
- CIDR: Classless Inter-Domain Routing

University	First address	Last address	How many	Prefix
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
(Available)	194.24.12.0	194.24.15.255	1024	194.24.12.0/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

Figure 5-50. A set of IP address assignments.

Aggregation

- Allocation

Supernet -> subnet

- Aggregation

Subnet -> Supernet

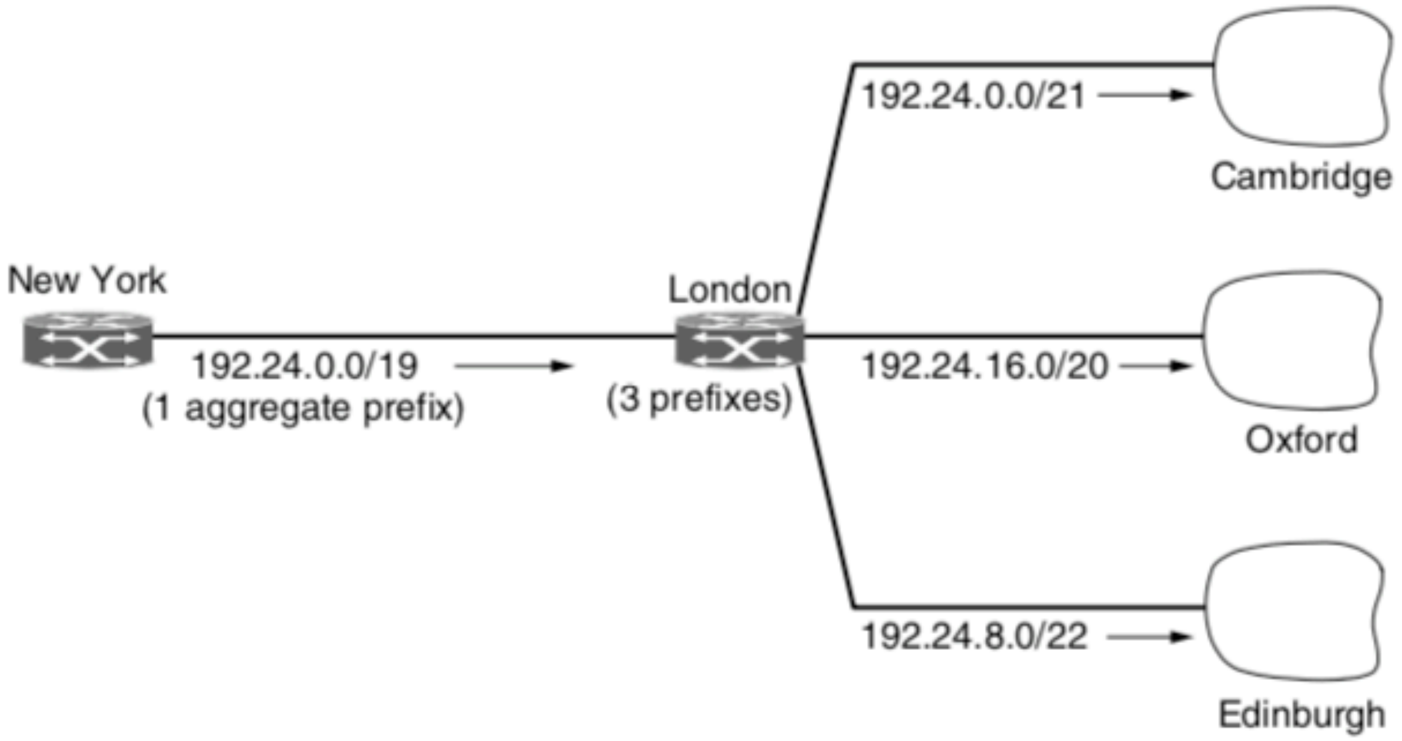


Figure 5-51. Aggregation of IP prefixes.

Aggregation

- The routing process in London combines the three prefixes into a single aggregate entry for the prefix.
- Reduce the prefixes in New York router and the routing table entries in the New York router.

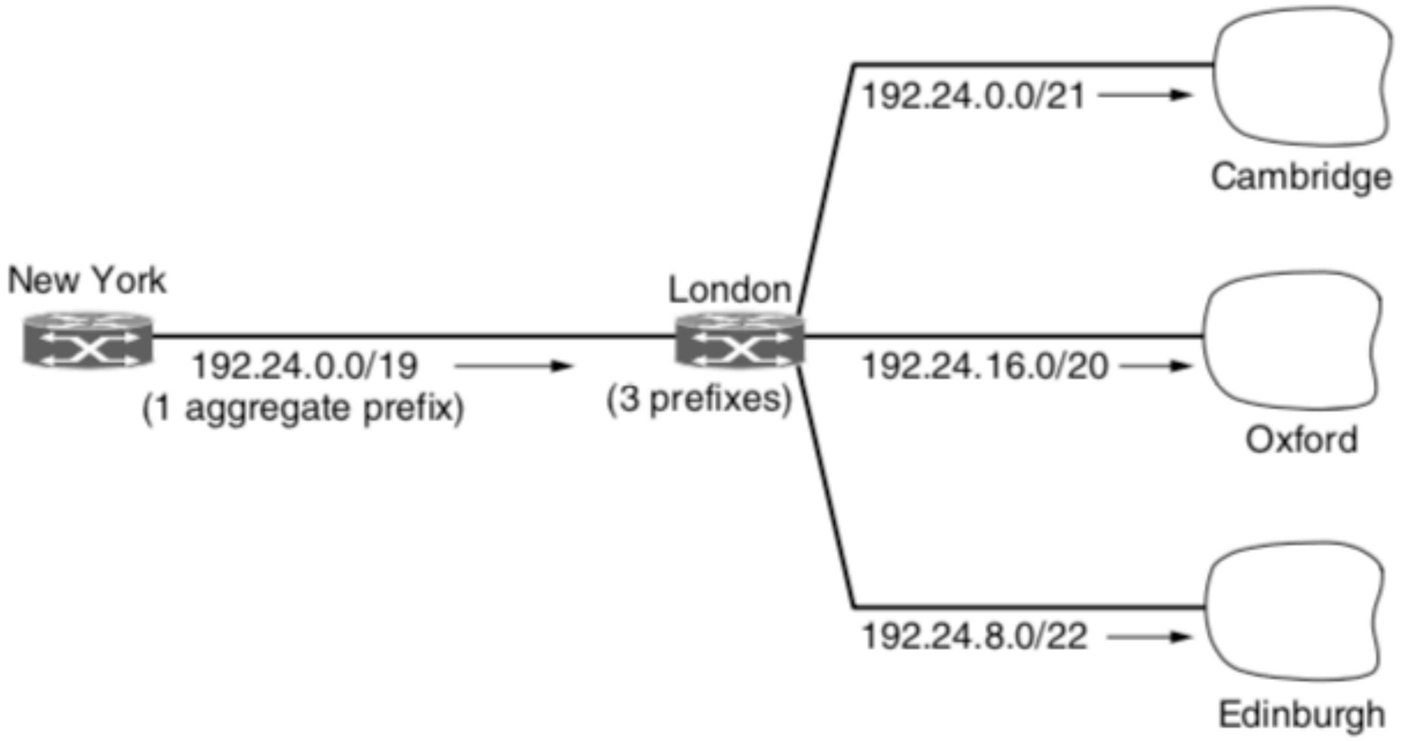


Figure 5-51. Aggregation of IP prefixes.

Question 1

- A router has just received the following IP addresses:

57.6.96.0/21

57.6.104.0/21

57.6.112.0/21

57.6.120.0/21

- If all of them use the same outgoing line, can they be aggregated? If so, to what? If not, why not?

Solution 1

- 57 6 96 0 / 21
- 00111001 00000110 01100000 00000000
- 57 6 104 0 / 21
- 00111001 00000110 01101000 00000000
- 57 6 112 0 / 21
- 00111001 00000110 01110000 00000000
- 57 6 120 0 / 21
- 00111001 00000110 01111000 00000000

Solution 1

Prefix

- 57 6 96 0 / 21
- 00111001 00000110 01100000 00000000
- 57 6 104 0 / 21
- 00111001 00000110 01101000 00000000
- 57 6 112 0 / 21
- 00111001 00000110 01110000 00000000
- 57 6 120 0 / 21
- 00111001 00000110 01111000 00000000

Solution 1

Network portion

- 57 6 96 0 / 21
- 00111001 00000110 01100|000 00000000
- 57 6 104 0 / 21
- 00111001 00000110 01101|000 00000000
- 57 6 112 0 / 21
- 00111001 00000110 01110|000 00000000
- 57 6 120 0 / 21
- 00111001 00000110 01111|000 00000000

Solution 1

- 0011100100000110011 | 0000000000000
- 0011100100000110011 | 0100000000000
- 0011100100000110011 | 1000000000000
- 0011100100000110011 | 1100000000000

👆

19

- 00111001 00000110 01100000 00000000 -> 57.6.96.0/19

They can be aggregated to 57.6.96.0/19

Question 2

- A router has the following entries in its routing table:

Address/mask	Next hop
`151.46.184.0/22`	`Interface 0`
`151.46.188.0/22`	`Interface 1`
`151.53.40.0/23`	`Router 1`
`default`	`Router 2`

- For each of the following IP addresses, what does the router do If a packet with that address arrives?
- (a) `151.46.191.10`
- (b) `151.46.187.2`

Longest matching prefix

University	First address	Last address	How many	Prefix
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
San Francisco	194.24.12.0	194.24.15.255	1024	194.24.12.0/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

Figure 5-50. A set of IP address assignments.

Longest matching prefix

- Hold 4 specific prefixes in routing (Non-aggregation)
- =========> Hold only 2 entries (After aggregation)

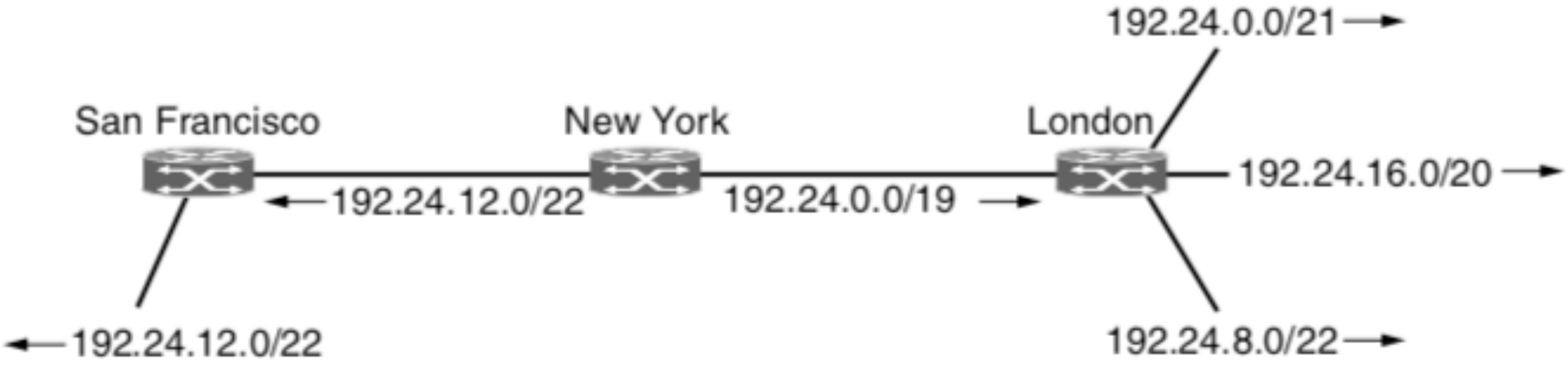


Figure 5-52. Longest matching prefix routing at the New York router.

Longest matching prefix

- One overall prefix is used to direct traffic for the entire block to London. One more specific prefix is also used to direct a portion of the larger prefix to San Francisco.

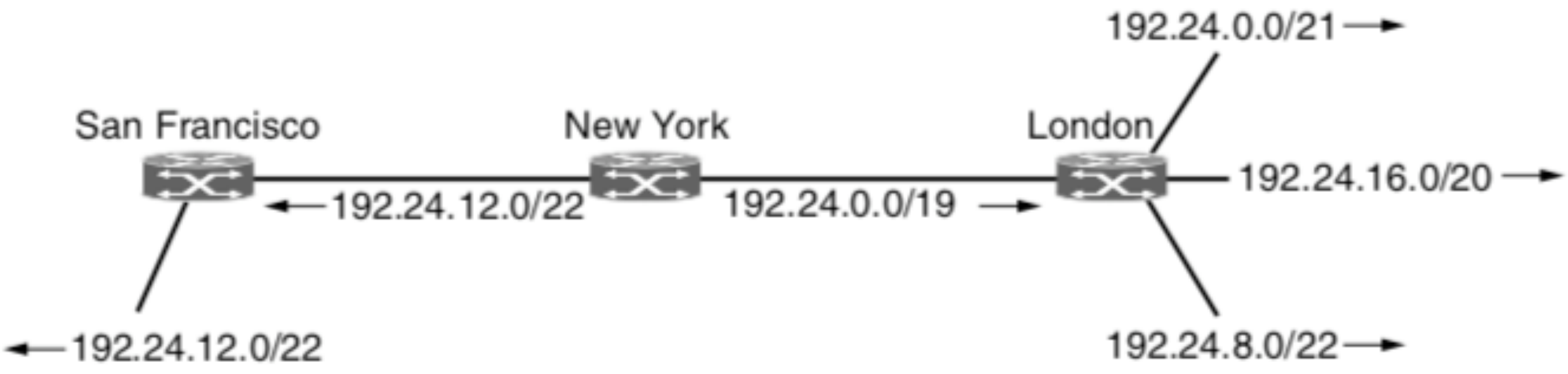


Figure 5-52. Longest matching prefix routing at the New York router.

Longest matching prefix

- With the longest matching prefix rule, IP addresses within the San Francisco network will be sent on the outgoing line to San Francisco, and all other IP addresses **in the larger prefix** will be sent to London.

Figure 5-52. Longest matching prefix routing at the New York router.

Question 2

- A router has the following entries in its routing table:

Address/mask	Next hop
`151.46.184.0/22`	`Interface 0`
`151.46.188.0/22`	`Interface 1`
`151.53.40.0/23`	`Router 1`
`default`	`Router 2`

- For each of the following IP addresses, what does the router do If a packet with that address arrives?
- (a) `151.46.191.10`
- (b) `151.46.187.2`

Solution 2

- 151.46.191.10
- ======>10010111 00101110 10111111 00001010
- 151.46.184.0/22
- ======>10010111 00101110 10111000 00000000
- 151.46.188.0/22
- ======>10010111 00101110 10111100 00000000
- 151.53.40.0/23
- ======>10010111 00110101 00101000 00000000

Solution 2

- 151.46.191.10
- ======>10010111 00101110 10111111 00001010
- 151.46.184.0/22
- ======>10010111 00101110 10111000 00000000
- 151.46.188.0/22
- ======>10010111 00101110 10111100 00000000
- 151.53.40.0/23
- ======>10010111 00110101 00101000 00000000

Solution 2

- 151.46.187.2
- ======>10010111 00101110 10111011 00000010
- 151.46.184.0/22
- ======>10010111 00101110 10111000 00000000
- 151.46.188.0/22
- ======>10010111 00101110 10111100 00000000
- 151.53.40.0/23
- ======>10010111 00110101 00101000 00000000

Solution 2

- 151.46.187.2
- ======>10010111 00101110 10111011 00000010
- 151.46.184.0/22
- ======>10010111 00101110 10111000 00000000
- 151.46.188.0/22
- ======>10010111 00101110 10111100 00000000
- 151.53.40.0/23
- ======>10010111 00110101 00101000 00000000

Solution 2

- A router has the following entries in its routing table:

Address/mask	Next hop
151.46.184.0/22	Interface 0
151.46.188.0/22	Interface 1
151.53.40.0/23	Router 1
default	Router 2

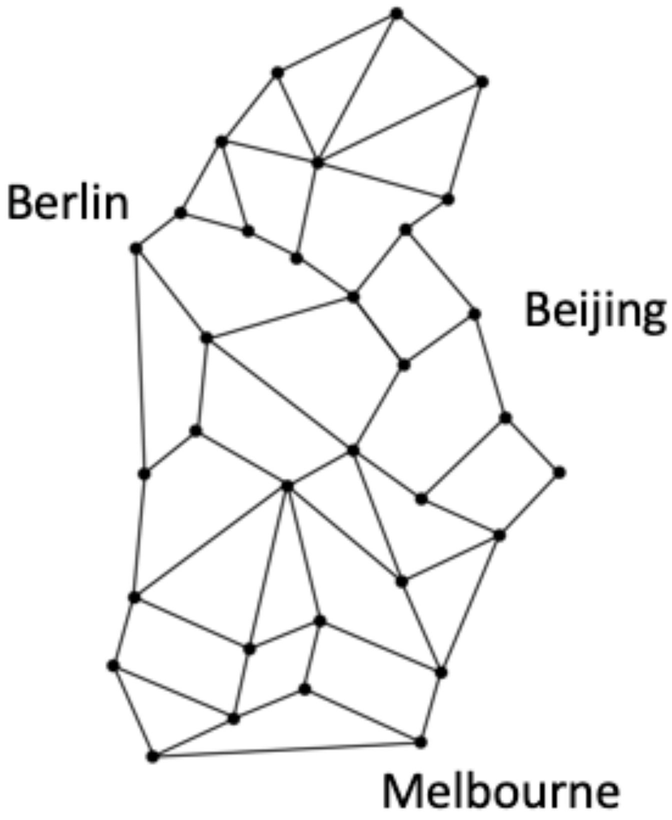
- For each of the following IP addresses, what does the router do If a packet with that address arrives?
- (a) 151.46.191.10 ---------------> Interface 1
- (b) 151.46.187.2 ---------------> Interface 0

Question 3

- Why do we need routing algorithms in the Network layer? What are the key categories of routing algorithms?

Solution 3

- Routing algorithms are needed to help decide on which output line an incoming packet should be transmitted.



Routing algorithms

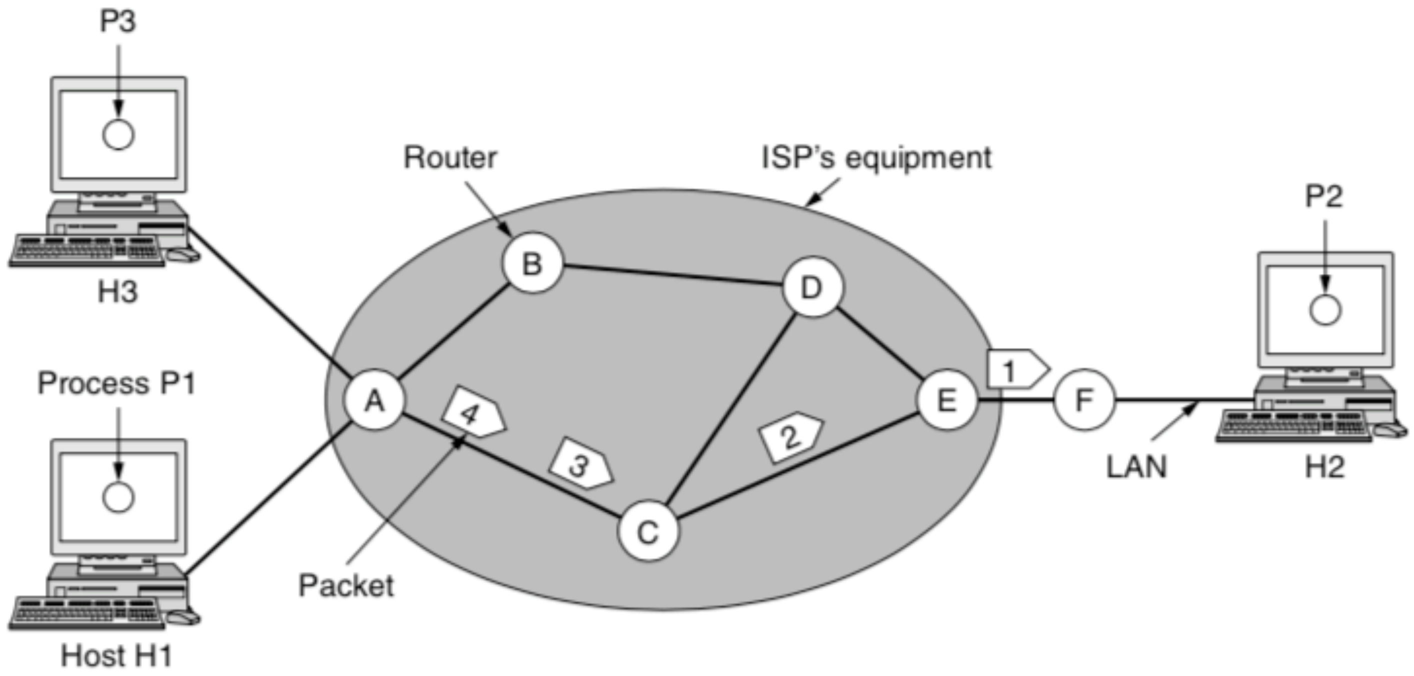
- Routing algorithms can be grouped into two major classes: *non-adaptive* and *adaptive*.
- Key Categories:
  - Non-Adaptive Algorithms
  - Adaptive Algorithms

Non-adaptive algorithms

- Do not base their routing decisions on any measurements or estimates of the current topology and traffic. Instead, the choice of the route to use to get from *I* to *J* (for all *I* and *J*) is computed in advance, offline, and downloaded to the routers when the network is booted.
- Advantage:
  - it does not respond to failures, non-adaptive routing is mostly useful for situations in which the routing choice is clear.

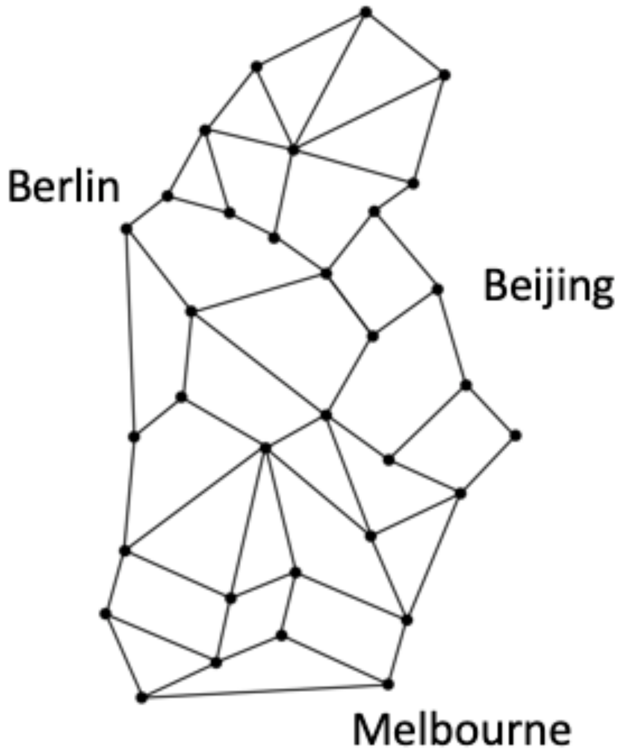
Non-adaptive algorithms

- router *F* should send packets headed into the network to router *E* regardless of the ultimate destination.



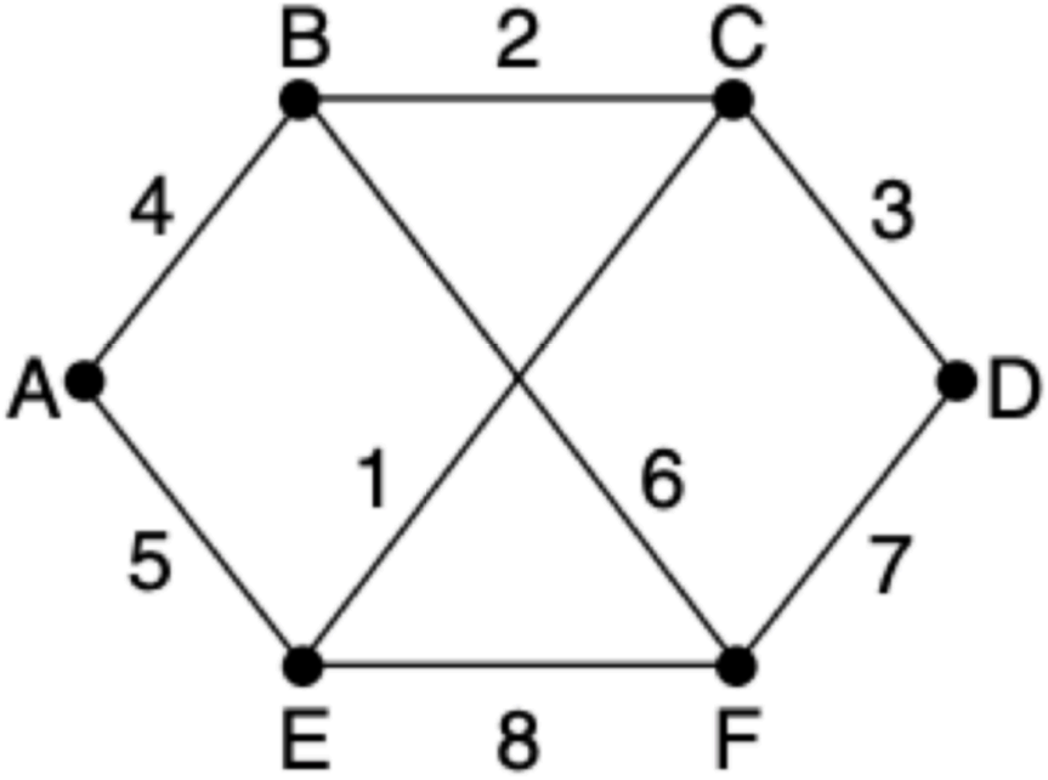
Adaptive algorithms

- In contrast, adaptive algorithms change their routing decisions to reflect changes in the topology, and sometimes changes in the traffic as well.
- Differ in:
  - where they get their information (e.g., locally, from adjacent routers, or from all routers)
  - when they change the routes (e.g., when the topology changes, or every ΔT seconds as the load changes)
  - what metric is used for optimization (e.g., distance, number of hops, or estimated transit time).



Question 4

- Compute the sink tree for Node F in the graph below:

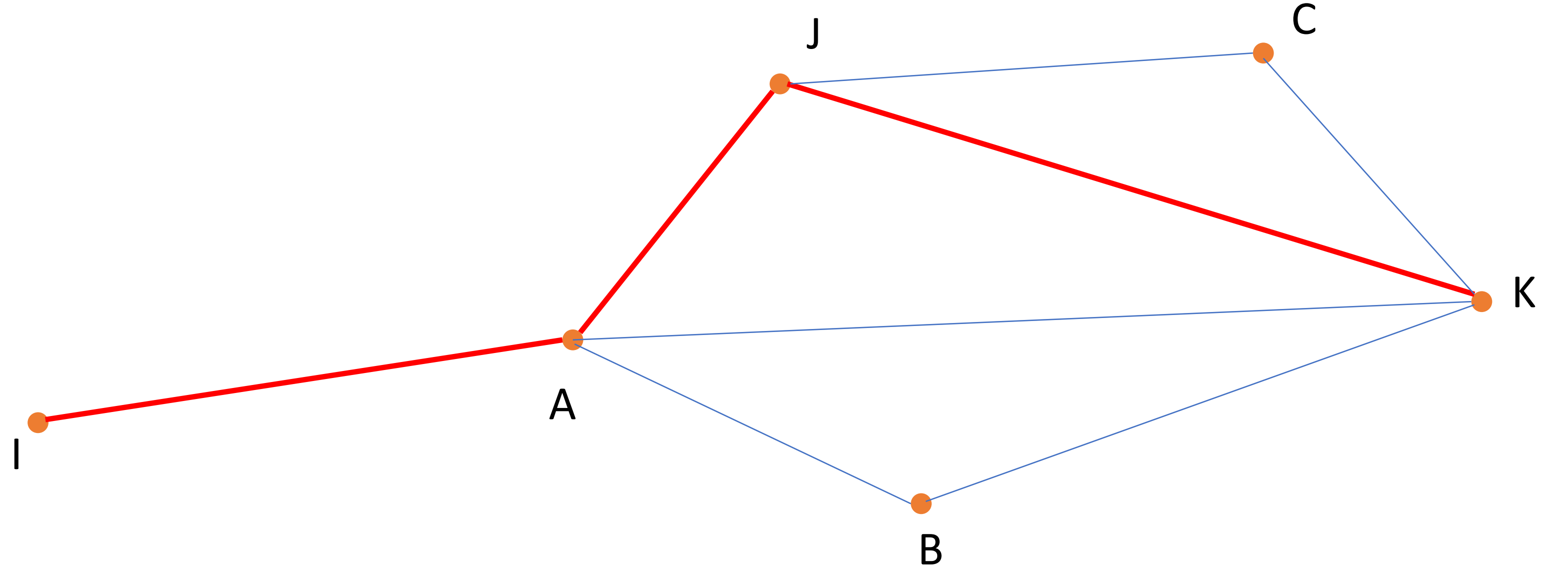


Routing Algorithms

- Shortest Path Algorithm
- Flooding
- Distance Vector Routing
- Link State Routing
- Hierarchical Routing
- Broadcast Routing
- Anycast Routing
- …….

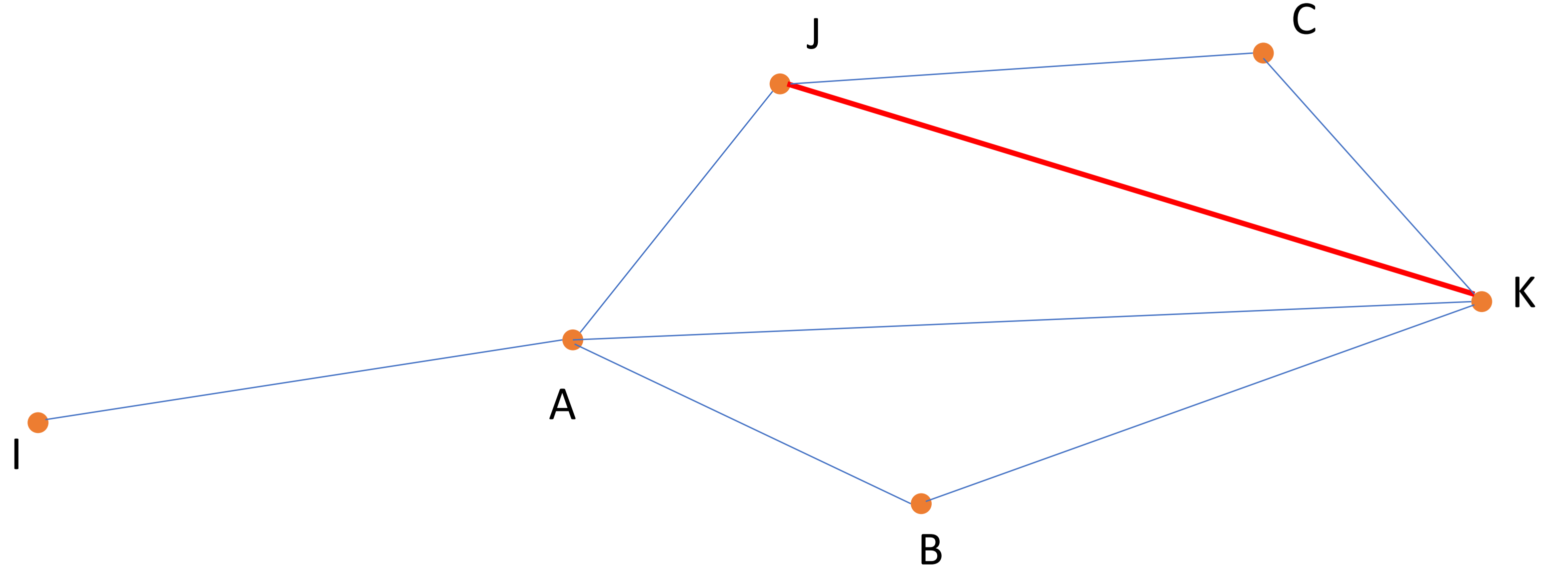
Optimality Principle

- a general statement about optimal routes without regard to network topology or traffic.
- It states that if router ***J*** is on the optimal path from router ***I*** to router ***K***, then the optimal path from ***J*** to ***K*** also falls along the same route



Optimality Principle

- a general statement about optimal routes without regard to network topology or traffic.
- It states that if router ***J*** is on the optimal path from router ***I*** to router ***K***, then the optimal path from ***J*** to ***K*** also falls along the same route



Shortest Path: Dijkstra

- Algorithm:

  1. Create **a set *P, tracking* the nodes added in the tree**. Initialize it as empty.

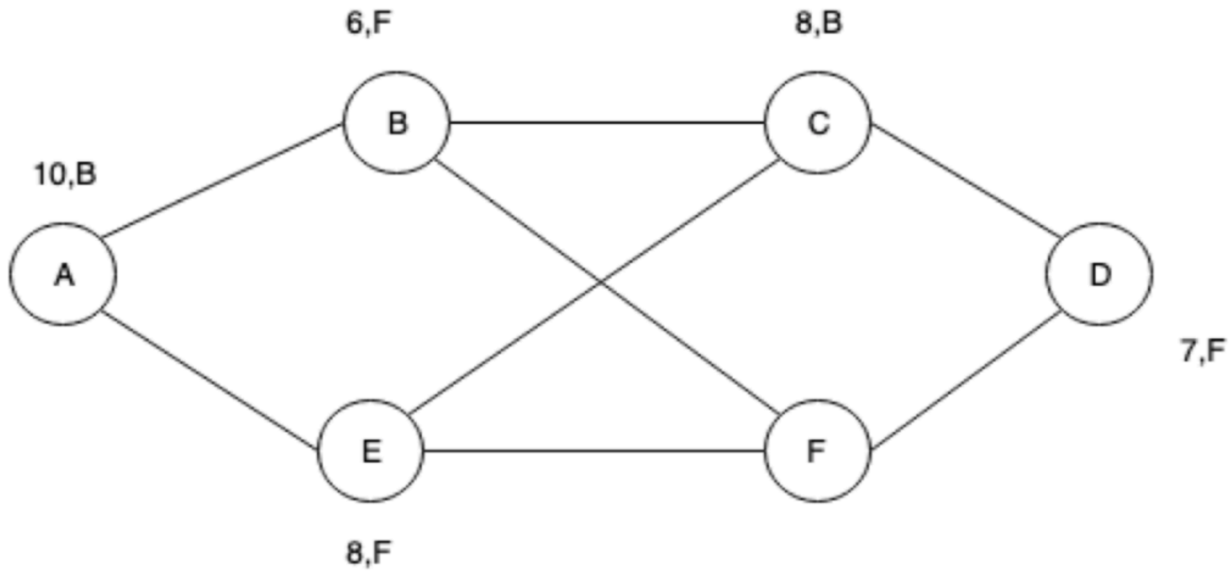
  2. For each node, assign a **distance value *d* from the node to sink**. Initialize the distance for all nodes as infinity.

  3. Start from the sink node, assign distance as 0.

  4. Repeat when *P* doesn't include all nodes:
     - For all the nodes not in *P, c*ompare distance *d*
     - Pick a node v with min distance and add it to P
     - Update *d* for all the adjacent nodes of *v* (newly added node)

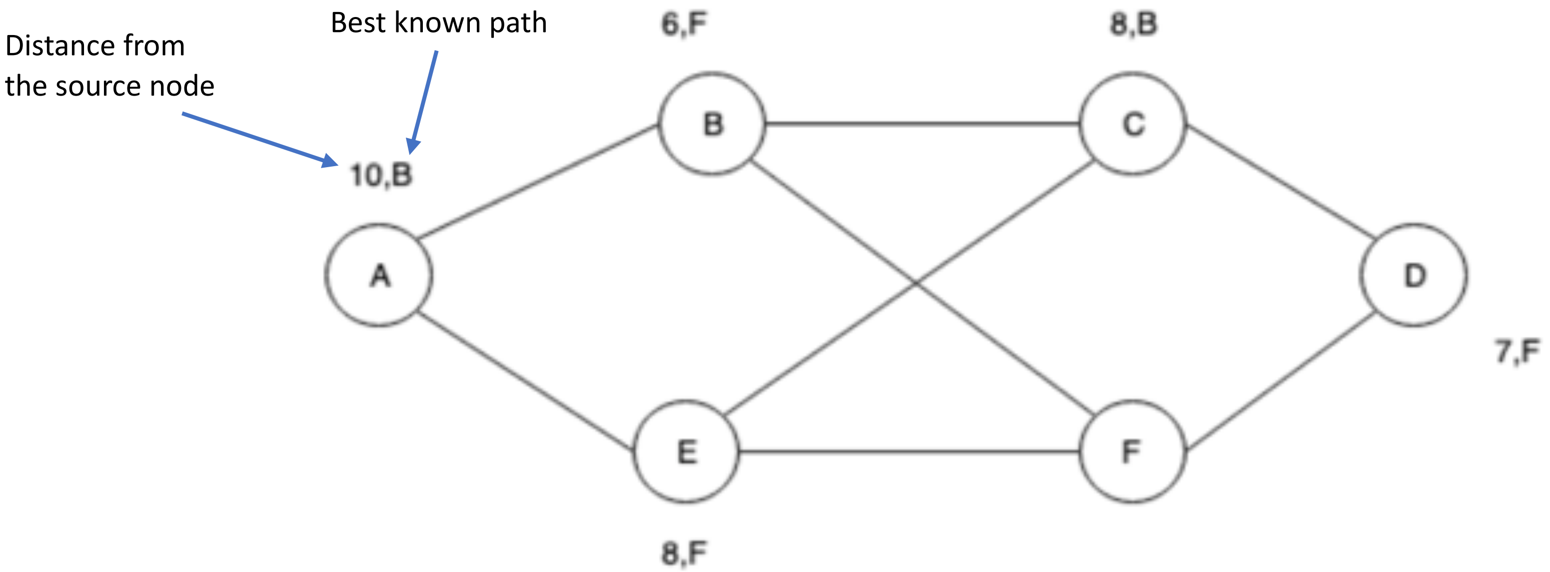
Solution 4

- Refer to Week 6 - Network Layer (3) of the Lecture



Routing Algorithms

- **Shortest Path Algorithm**: finds the shortest paths between a source and all destinations in the network (Dijkstra)



Initially,
labels: Tentative
distance: Infinity

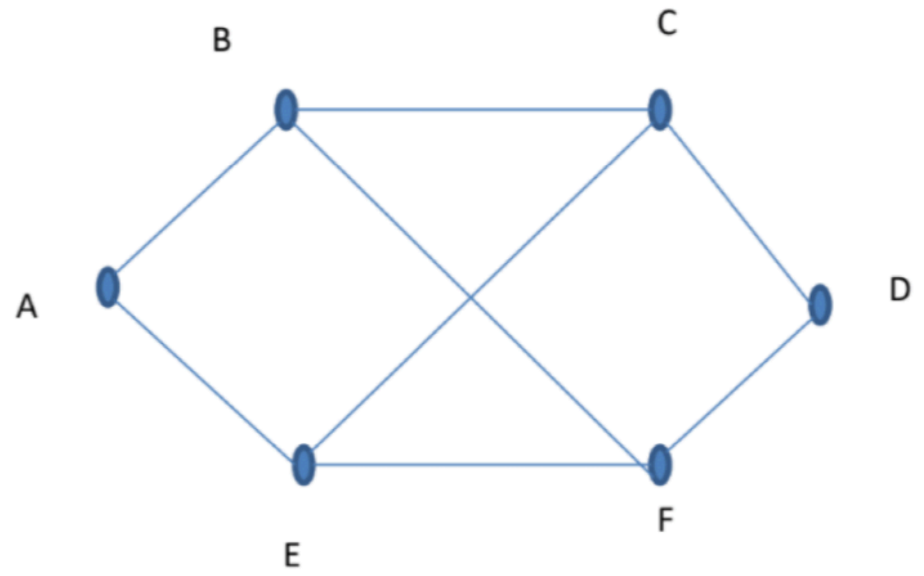
After Dijkstra,
labels: Permanent
distance: Shortest

Routing Algorithms

- **Distance Vector Routing Algorithm**
  - Dynamic algorithm
  - More efficient than flooding algorithm
  - each router maintain a table (i.e., a vector) giving the best known distance to each destination and which link to use to get there. These tables are updated by *exchanging information with the neighbours*. (why?) Eventually, every router knows the best link to reach each destination.

Question 5

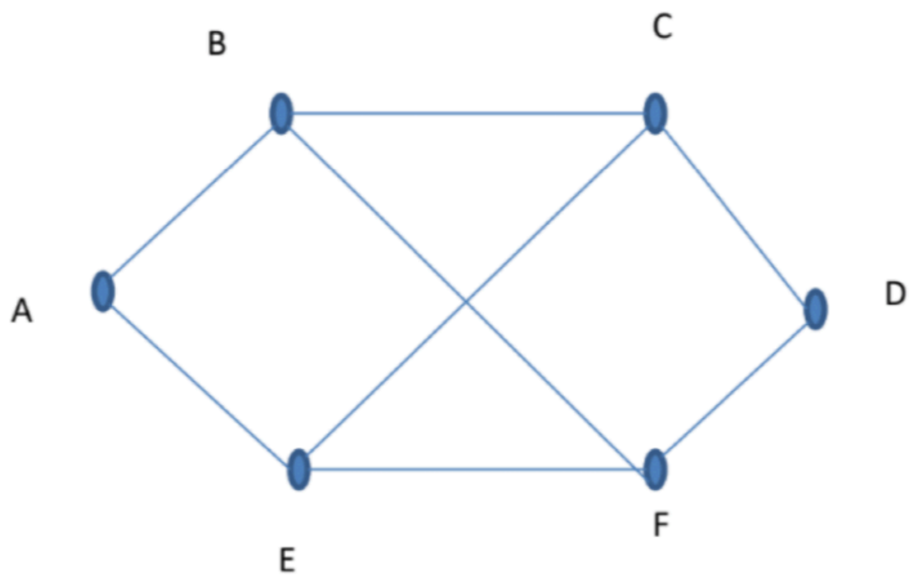
- Distance vector routing is used for the diagram shown below, and the following vectors have just come in to router C: from B: (5, 0, 8, 12, 6, 2); from D: (16, 12,6, 0, 9, 10); and from E: (7, 6, 3, 9, 0, 4). The cost of the links from C to B, D, and E, are 6, 3, and 5, respectively. What is C's new routing table? Give both the outgoing line to use and the expected delay.



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

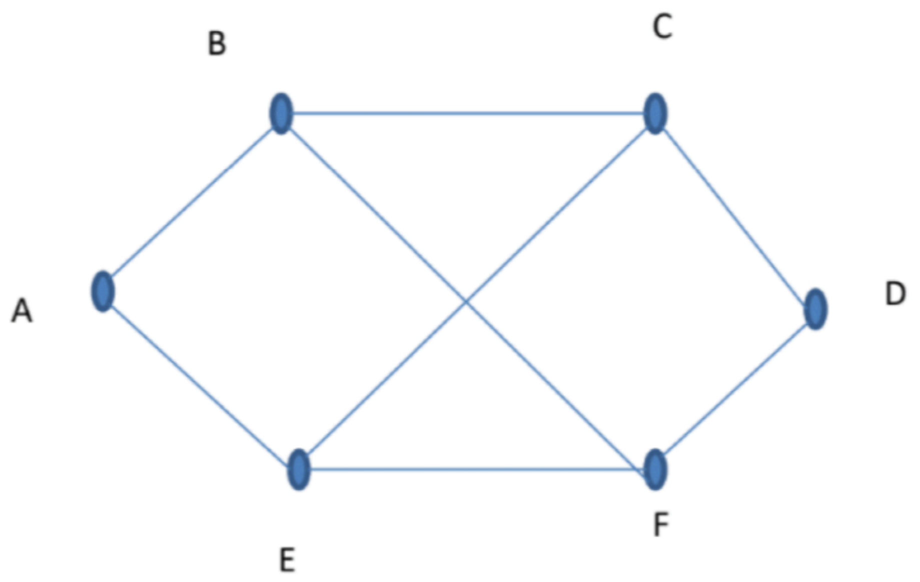
C	



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

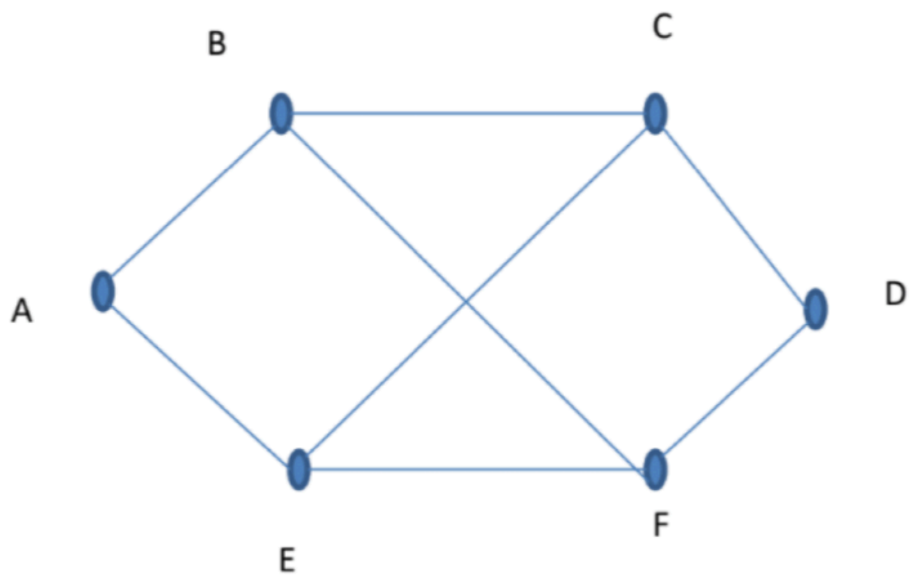
C	
11	B



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

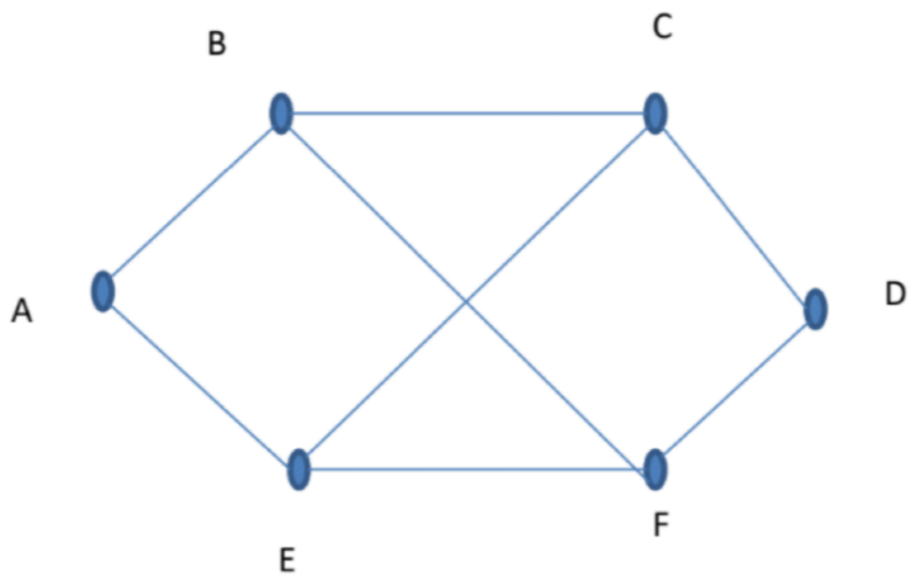
C	
11	B
6	B



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

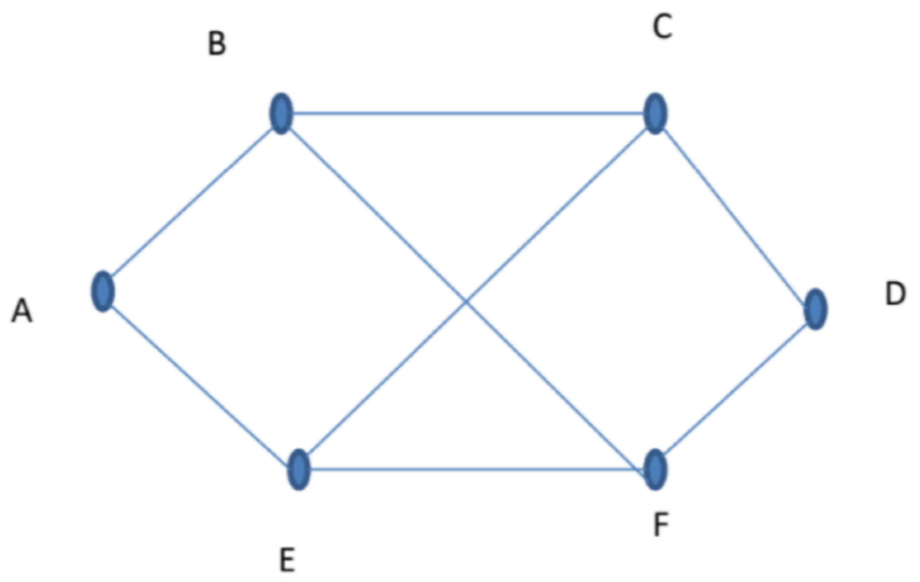
C	
11	B
6	B
0	-



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

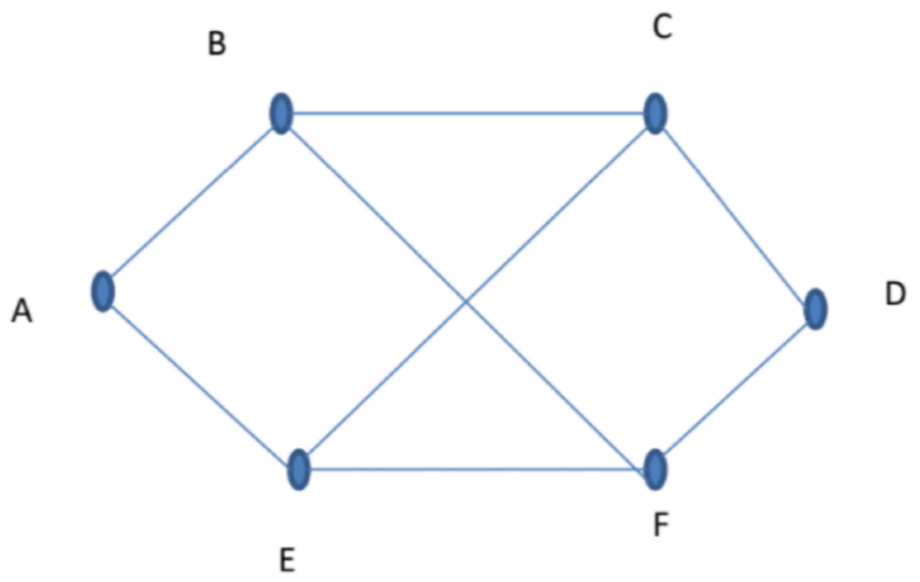
C	
11	B
6	B
0	-
3	D



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

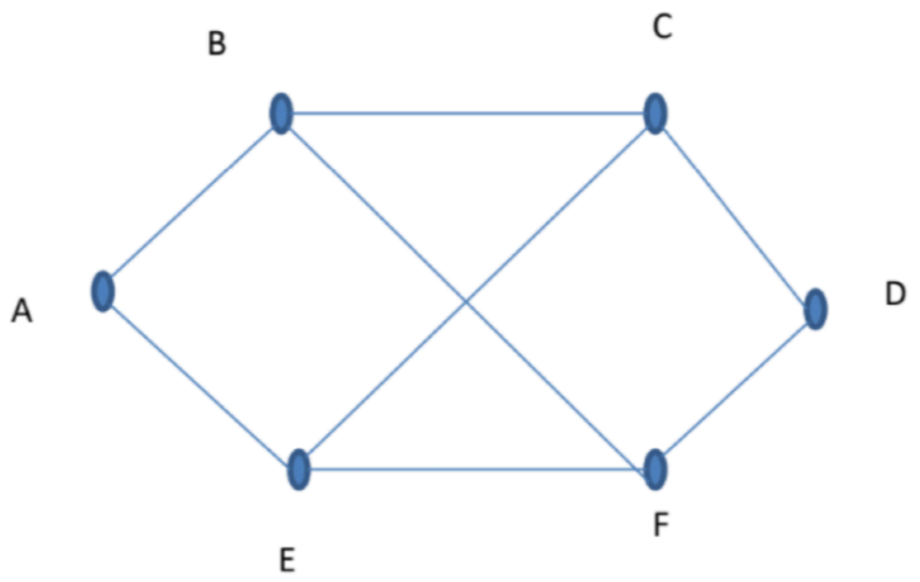
C	
11	B
6	B
0	-
3	D
5	E



Solution 5

	B	D	E
A	5	16	7
B	0	12	6
C	8	6	3
D	12	0	9
E	6	9	0
F	2	10	4
C ->	6	3	5

C	
11	B
6	B
0	-
3	D
5	E
8	B



Solution 5

- *Using the delays 6, 3, and 5 for B, D, and E, the vectors will be written as:*
- *B -> 11, 6, 14, 18, 12, 8*
- *D -> 19, 15, 9, 3, 12, 13*
- *E-> 12,11,8,14,5,9*

All Routers	Outgoing Line	Expected Delay
A	B	11
B	B	6
C	-	0
D	D	3
E	E	5
F	B	8