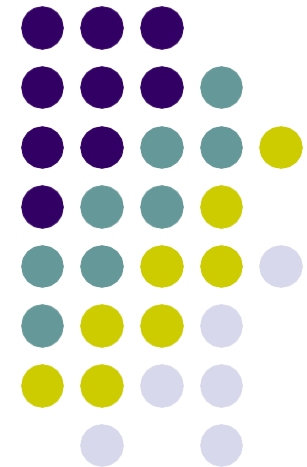


Chương 6: Chọn đường - Routing

Giảng viên: Nguyễn Đức Toàn

Bộ môn Truyền thông và Mạng máy tính
Viện CNTT&TT - ĐHBK Hà Nội



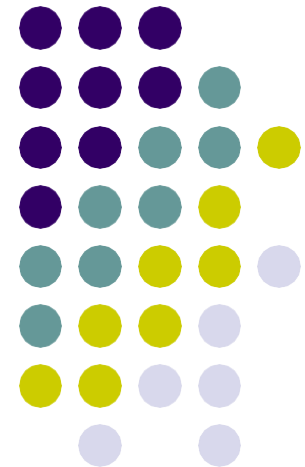


Tổng quan

- Tuần trước
 - Giao thức IP
 - địa chỉ IP và cấu trúc gói tin IP
 - Giao thức ICMP
- Tuần này: Tiếp tục về tầng mạng
 - Thế nào là chọn đường?
 - Chọn đường tĩnh và chọn đường động
 - Giải thuật và giao thức chọn đường

Chọn đường là gì?

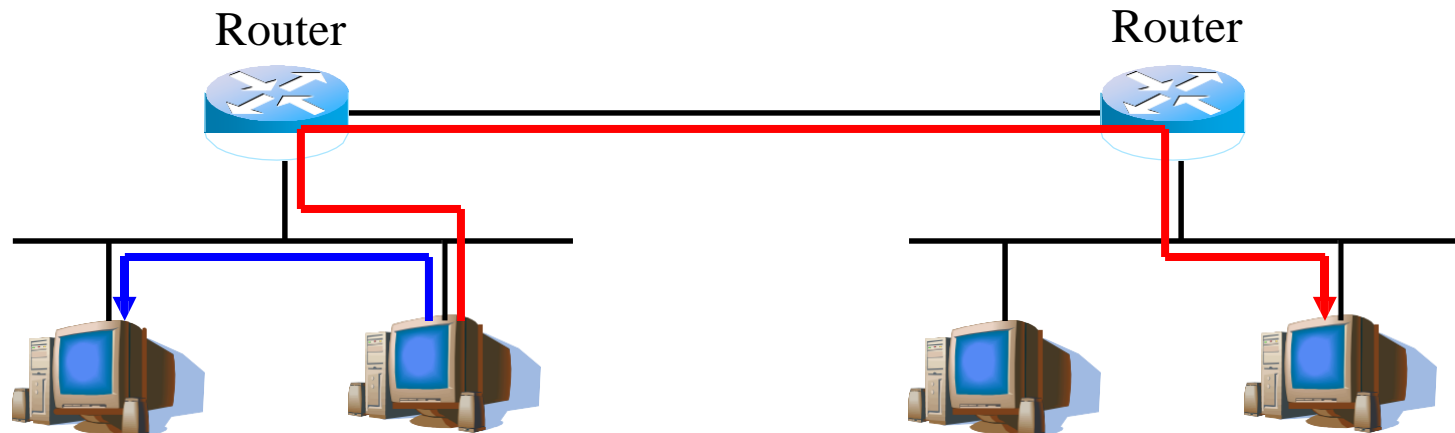
Các nguyên lý chọn đường
Cơ chế chuyển tiếp gói tin
Quy tắc “Longest matching”



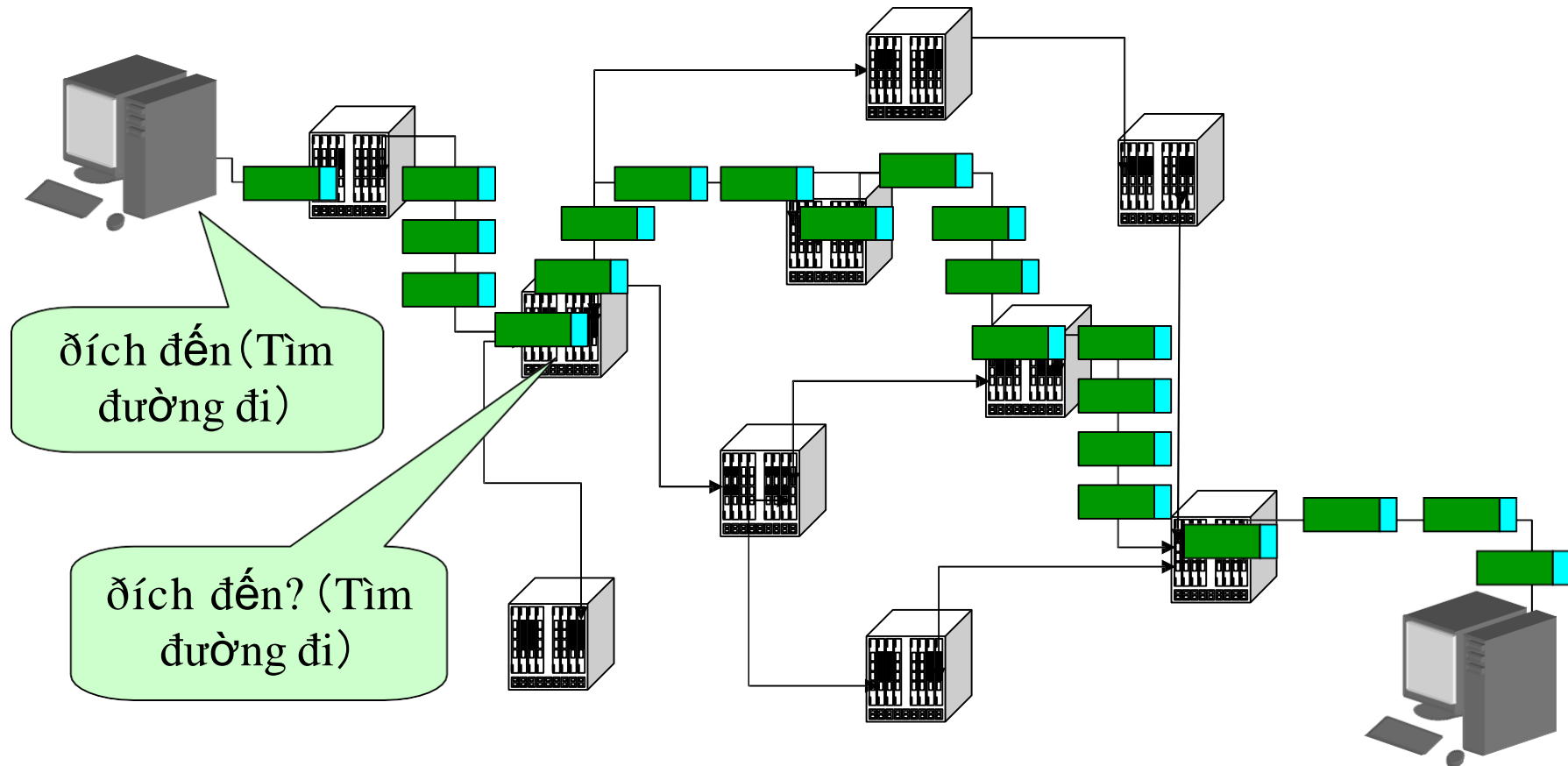


Cơ bản về chọn đường (1)

- Khi một máy trạm gửi một gói tin IP tới một máy khác
 - Nếu địa chỉ đích nằm trên cùng một đường truyền vật lý: Chuyển trực tiếp
 - Nếu địa chỉ đích nằm trên một mạng khác: Chuyển gián tiếp qua bộ định tuyến (chọn đường)



Cơ bản về chọn đường (2)





Chọn đường là gì?

- Cơ chế để máy trạm hay bộ định tuyến chuyển tiếp gói tin từ nguồn đến đích
- Các thành phần của chọn đường
 - Bảng chọn đường
 - Thông tin chọn đường
 - Giải thuật, giao thức chọn đường



Bộ định tuyến?

- Thiết bị chuyển tiếp các gói tin giữa các mạng
 - Là một máy tính, với các phần cứng chuyên dụng
 - Kết nối nhiều mạng với nhau
 - Chuyển tiếp gói tin dựa trên bảng chọn đường
- Có nhiều giao diện
- Phù hợp với nhiều dạng lưu lượng và phạm vi của mạng

Một số ví dụ...



BUFFALO
BHR-4RV



PLANEX
GW-AP54SAG



YAMAHA
RTX-1500



Cisco 2600

Router ngoại vi



Cisco CRS-1

Router mạng trục



Hitachi
GR2000-1B



Juniper M10



Foundry Networks
NetIron 800



Cisco 3700

Router cỡ trung

<http://www.cisco.com.vn>

<http://www.juniper.net/>

<http://www.buffalotech.com>



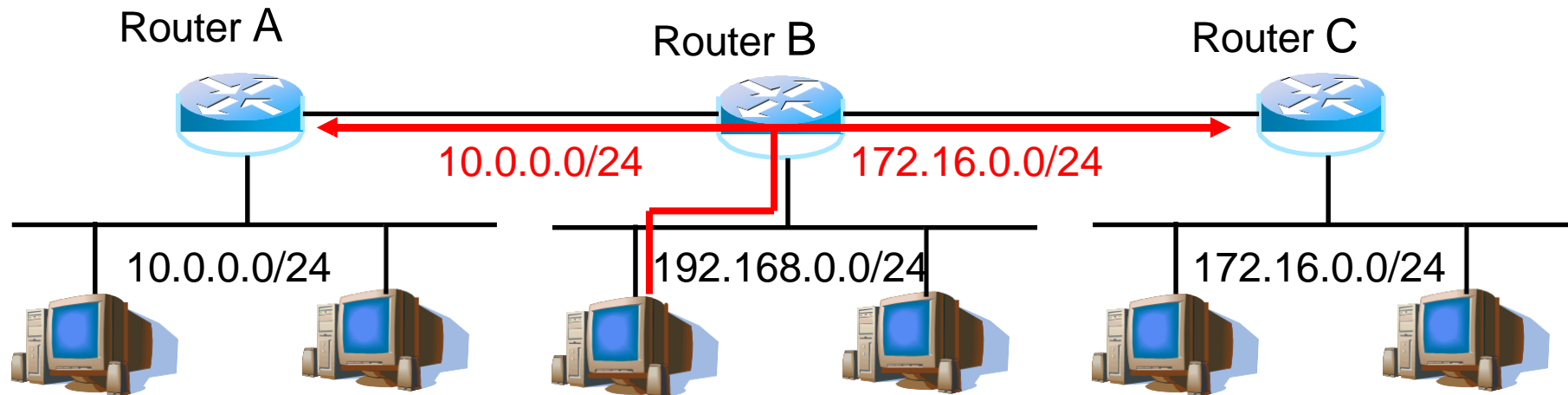
Bảng chọn đường

- Chỉ ra danh sách các đường đi có thể, được lưu trong bộ nhớ của router
- Các thành phần chính của bảng chọn đường
 - địa chỉ đích/mặt nạ mạng
 - Router kế tiếp

Bảng chọn đường và cơ chế chuyển tiếp (1)



Network	Next-hop
10.0.0.0/24	A
172.16.0.0/24	C



Lưu ý quy tắc: **No routes, no reachability!**

Quy tắc “Longest matching”(1)



- Giả sử một địa chỉ mạng đích lại có nhiều hơn một mục trong bảng chọn đường
- địa chỉ đích : 11.1.2.5
- Router kế tiếp nào sẽ được sử dụng?

Network	Next hop
11.0.0.0/8	A
11.1.0.0/16	B
11.1.2.0/24	C

Quy tắc “Longest matching”(2)



địa chỉ đích:

11.1.2.5 = 00001011.00000001.00000010.00000101

đường đi 1:

11.1.2.0/24 = 00001011.00000001.00000010.00000000

đường đi 2:

11.1.0.0/16 = 00001011.00000001.00000000.00000000

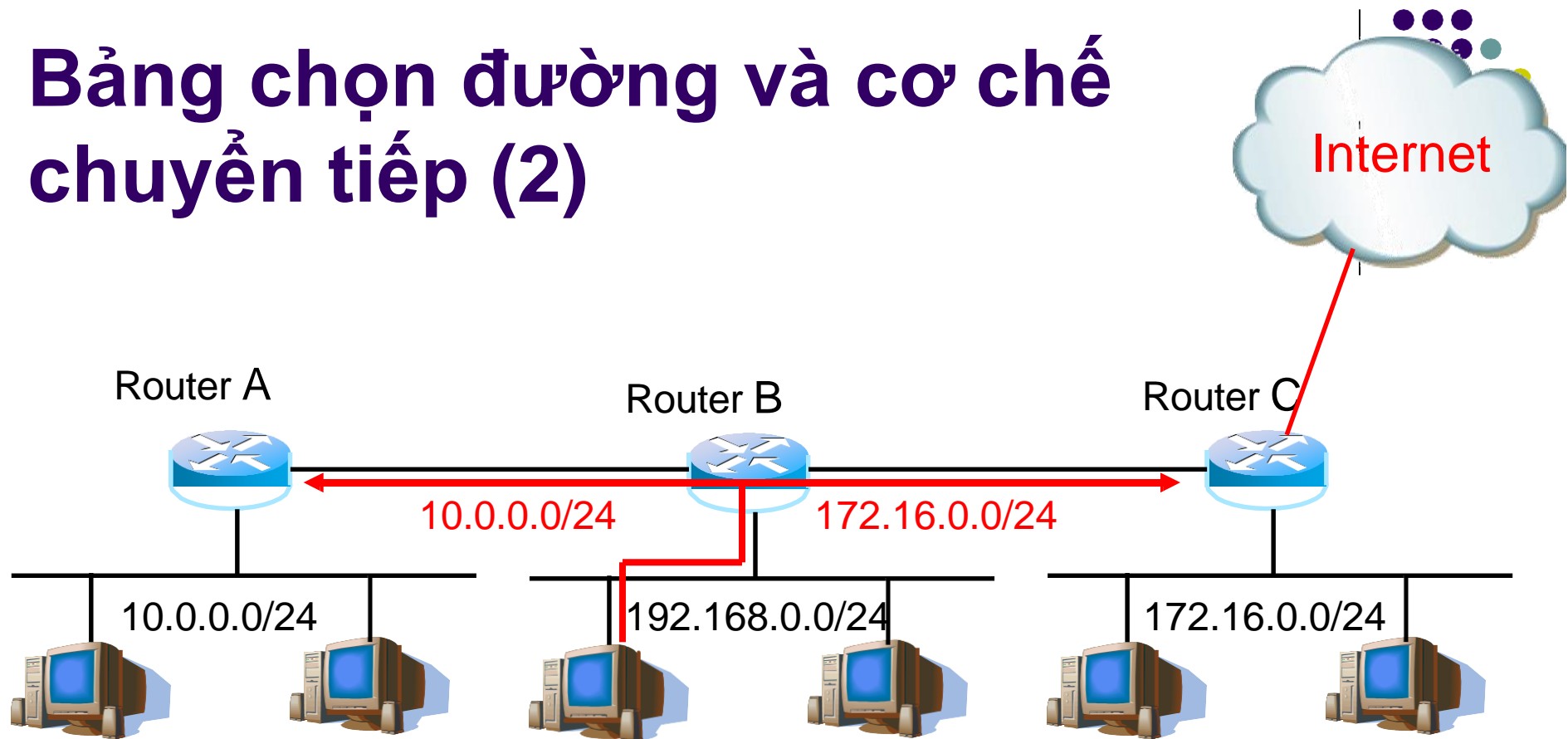
đường đi 3:

11.0.0.0/8 = 00001011.00000000.00000000.00000000

“Longest matching” là gì?

Tại sao phải cần quy tắc này?

Bảng chọn đường và cơ chế chuyển tiếp (2)



Network	Next-hop
10.0.0.0/24	A
172.16.0.0/24	C
192.168.0.0/24	Direct

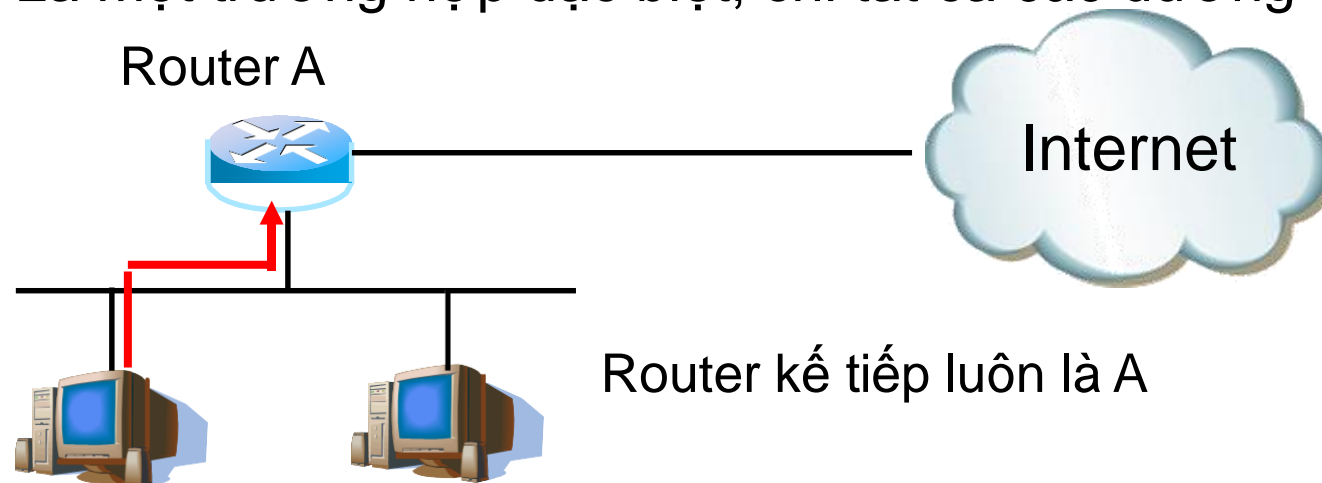
Q. Mô tả bảng chọn đường trên C

Nếu C nối vào Internet?



Đường đi mặc định

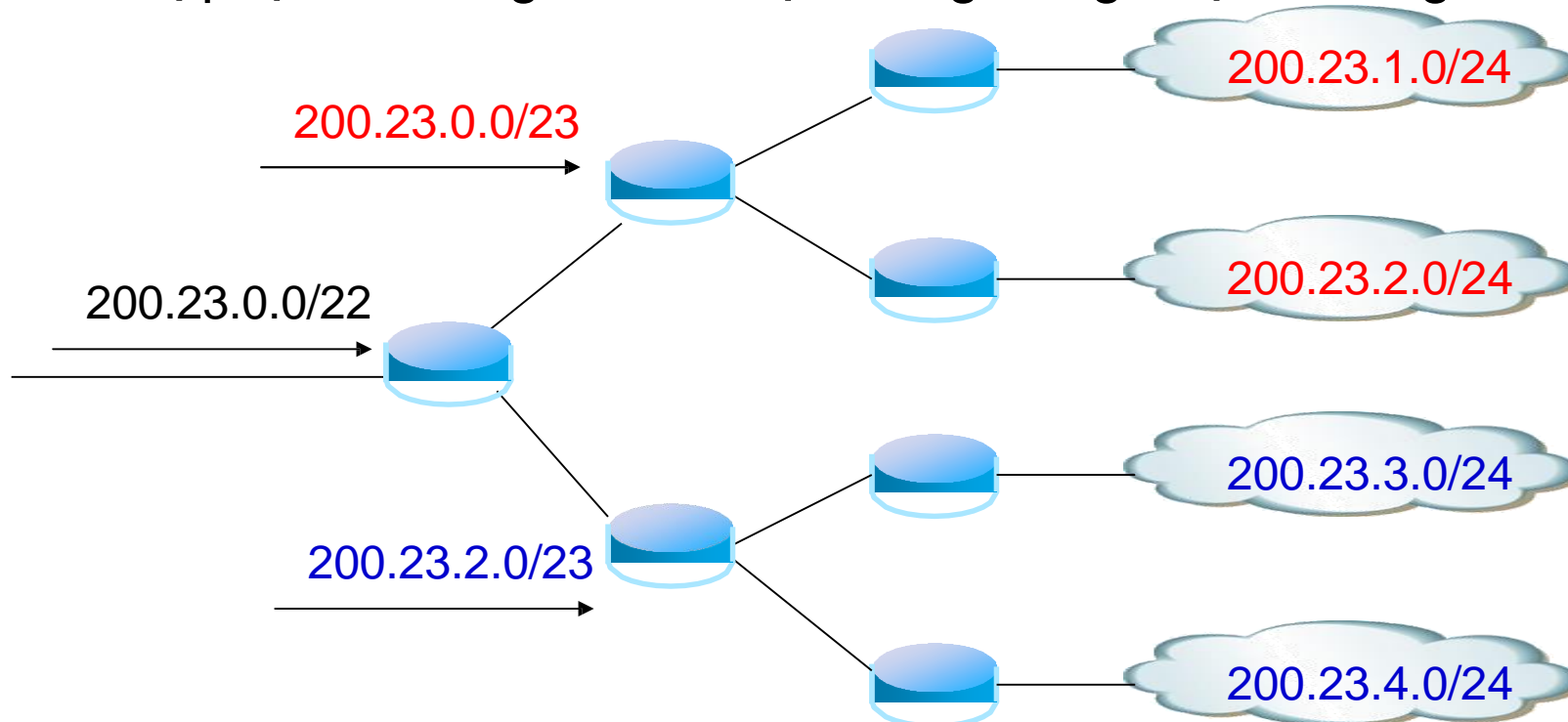
- Nếu đường đi không tìm thấy trong bảng chọn đường
 - Đường đi mặc định trở đến một router kết tiếp
 - Trong nhiều trường hợp, đây là đường đi duy nhất
- 0.0.0.0/0
 - Là một trường hợp đặc biệt, chỉ tất cả các đường đi



Kết hợp đường đi (Routing aggregation)



- Có bao nhiêu mạng con trên mạng Internet?
- Sẽ có rất nhiều mục trong bảng chọn đường?
- Các mạng con kế tiếp với cùng địa chỉ đích có thể được tổng hợp lại để làm giảm số mục trong bảng chọn đường.





Kết hợp đường đi (2)

- Ví dụ về Viettel
 - Không gian địa chỉ IP: khá lớn
 - 203.113.128.0-203.113.191.255
 - để kết nối đến một mạng con của Viettel (khách hàng): Chỉ cần chỉ ra đường đi đến mạng Viettel
- Đường đi mặc định chính là một dạng của việc kết hợp đường
 - 0.0.0.0/0

Ví dụ về bảng chọn đường – máy trạm



```
C:\Documents and Settings\hongson>netstat -rn
```

```
Route Table
```

```
=====
```

```
Interface List
```

```
0x1 .....MS TCP Loopback interface
```

```
0x2 ...08 00 1f b2 a1 a3 ..... Realtek RTL8139 Family PCI Fast Ethernet NIC -
```

```
=====
```

```
Active Routes:
```

Network	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	192.168.1.1	192.168.1.34	20
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
192.168.1.0	255.255.255.0	192.168.1.34	192.168.1.34	20
192.168.1.34	255.255.255.255	127.0.0.1	127.0.0.1	20
192.168.1.255	255.255.255.255	192.168.1.34	192.168.1.34	20
224.0.0.0	240.0.0.0	192.168.1.34	192.168.1.34	20
255.255.255.255	255.255.255.255	192.168.1.34	192.168.1.34	1

```
Default Gateway: 192.168.1.1
```

```
=====
```

Ví dụ về bảng chọn đường – Router (trích)

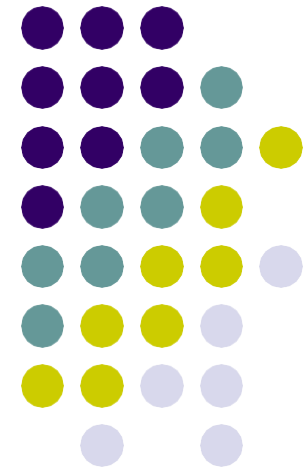


```
#show ip route
```

Prefix	Next Hop
203.238.37.0/24	via 203.178.136.14
203.238.37.96/27	via 203.178.136.26
203.238.37.128/27	via 203.178.136.26
203.170.97.0/24	via 203.178.136.14
192.68.132.0/24	via 203.178.136.29
203.254.52.0/24	via 203.178.136.14
202.171.96.0/24	via 203.178.136.14

Chọn đường tĩnh và chọn đường động

Chọn đường tĩnh
Chọn đường động
Ưu điểm – nhược điểm



Vấn đề cập nhật bảng chọn đường

- Sự thay đổi cấu trúc mạng: thêm mạng mới, một nút mạng bị mất điện
- Sự cần thiết phải cập nhật bảng chọn đường
 - Cho tất cả các nút mạng (về lý thuyết)
 - Thực tế, chỉ một số nút mạng phải cập nhật

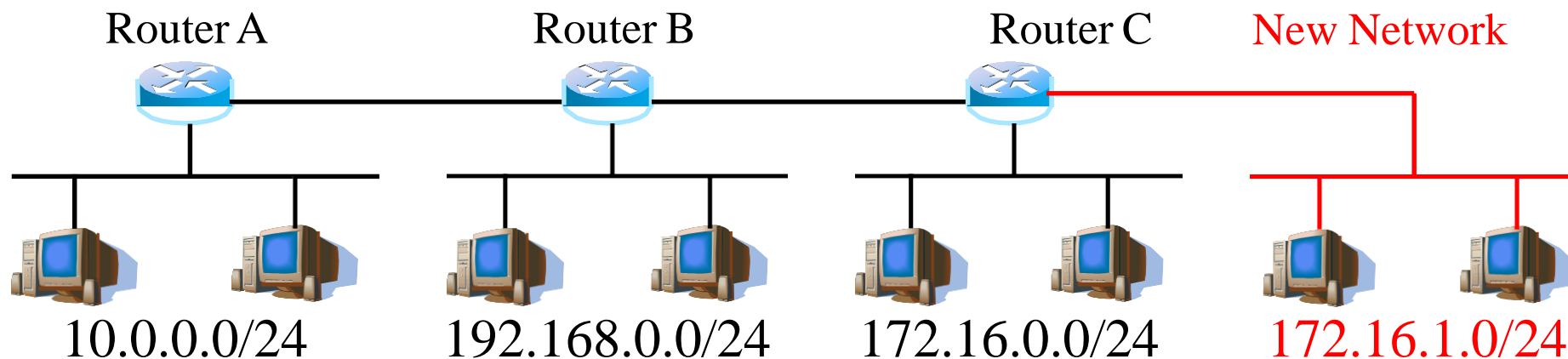
Network	Next-hop
192.168.0.0/24	B
172.16.0.0/24	B

172.16.1.0/24 B

Network	Next-hop
10.0.0.0/24	A
172.16.0.0/24	C

172.16.1.0/24 C

Network	Next-hop
10.0.0.0/24	B
192.168.0.0/24	B





Làm thế nào để cập nhật?

- Chọn đường tĩnh
 - Các mục trong bảng chọn đường được sửa đổi thủ công bởi người quản trị
- Chọn đường động
 - Tự động cập nhật bảng chọn đường
 - Bảng các giao thức chọn đường

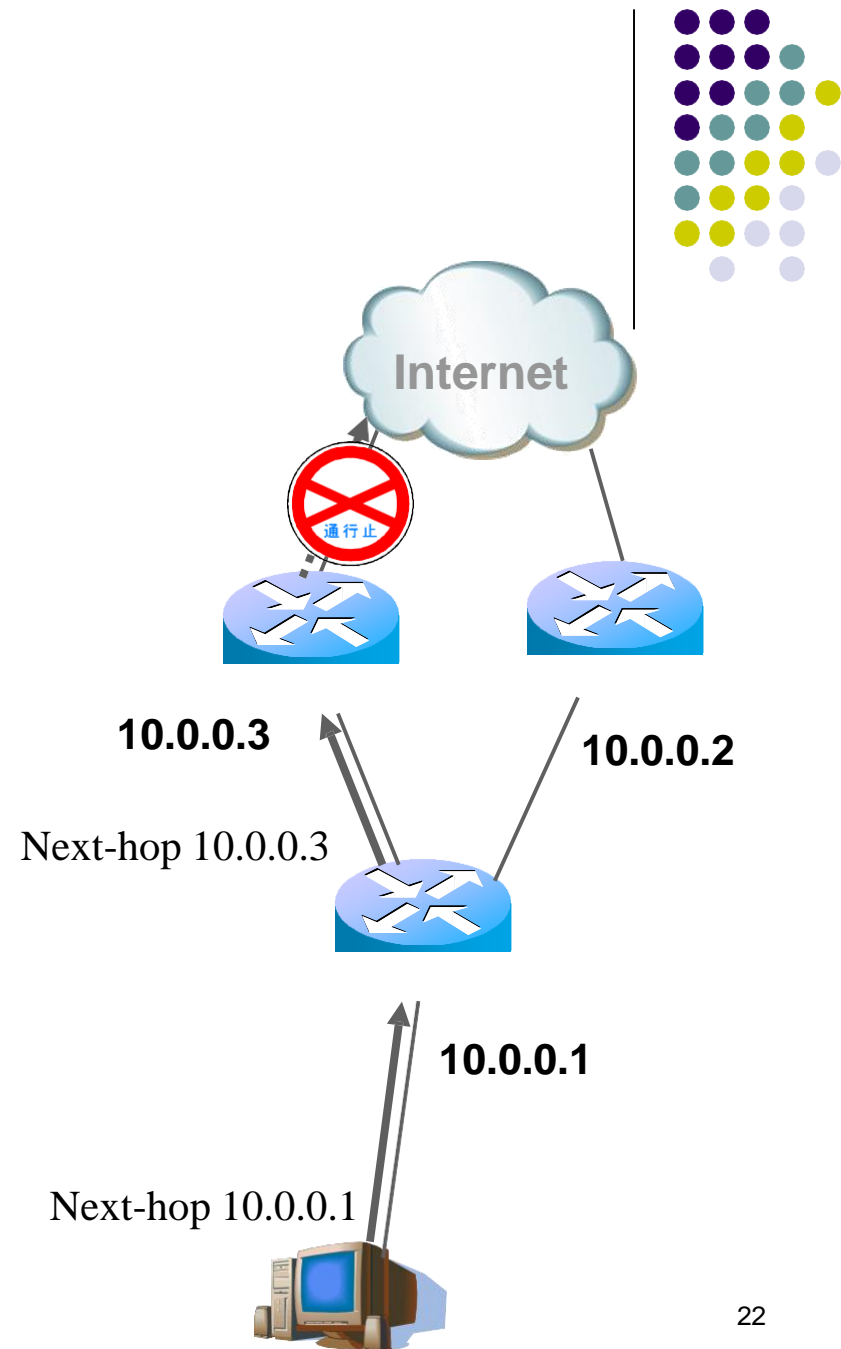
Chọn đường tĩnh

- Khi có sự cố:
 - Không thể nối vào Internet kể cả khi có tồn tại đường đi dự phòng
 - Người quản trị mạng cần thay đổi

Bảng chọn đường của 10.0.0.1 (1 phần)

Prefix	Next-hop
0.0.0.0/0	10.0.0.3

Kết nối bị lỗi



Chọn đường động

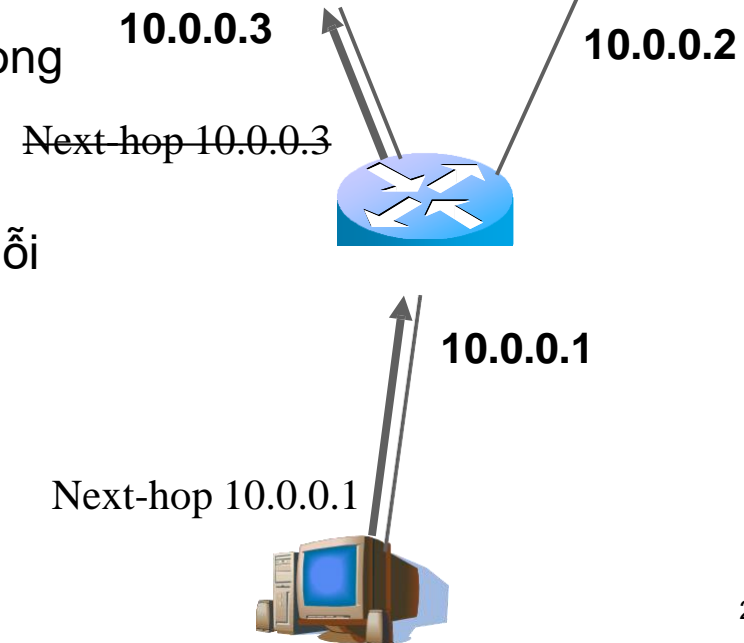
- Khi có sự cố:
 - đường đi thay thế được cập nhật một cách tự động

Bảng chọn đường của 10.0.0.1 (1 phần)

Prefix	Next-hop
0.0.0.0/0	10.0.0.2
0.0.0.0/0	10.0.0.3

Kết nối dự phòng

Kết nối bị lỗi



đặc điểm của chọn đường tĩnh



- Ưu
 - Ổn định
 - An toàn
 - Không bị ảnh hưởng bởi các yếu tố tác động
- Nhược
 - Cứng nhắc
 - Không thể sử dụng tự động kết nối dự phòng
 - Khó quản lý

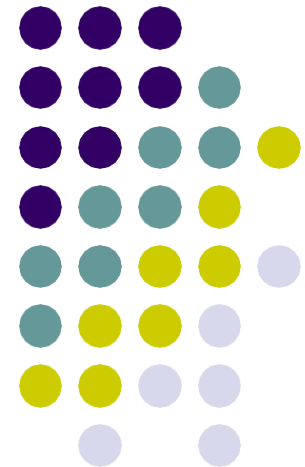


Chọn đường động

- Ưu
 - Dễ quản lý
 - Tự động sử dụng kết nối dự phòng
- Nhược
 - Tính an toàn
 - Các giao thức chọn đường phức tạp và khó hiểu
 - Khó quản lý

Các giải thuật và giao thức chọn đường

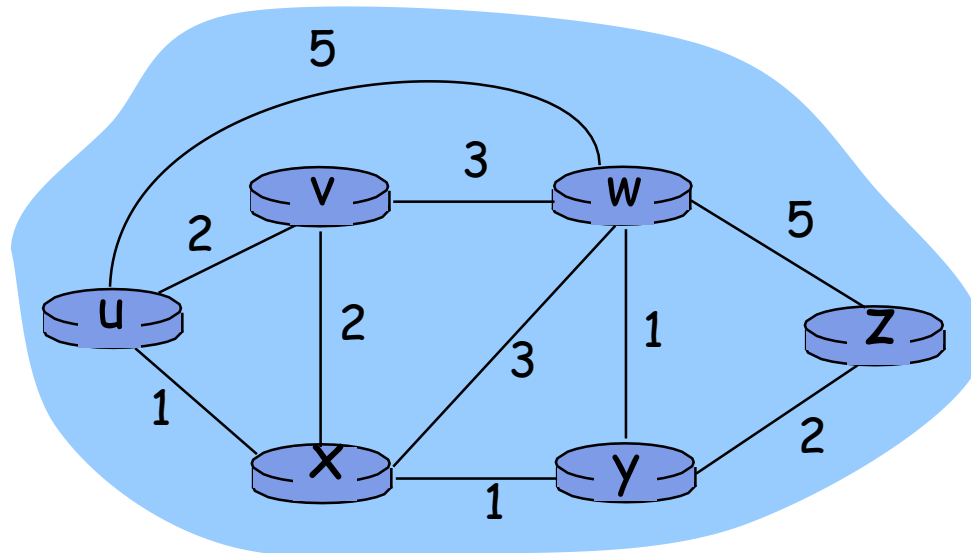
Giải thuật Dijkstra và Bellman-Ford
Giao thức dạng link-state và dạng
distance-vector



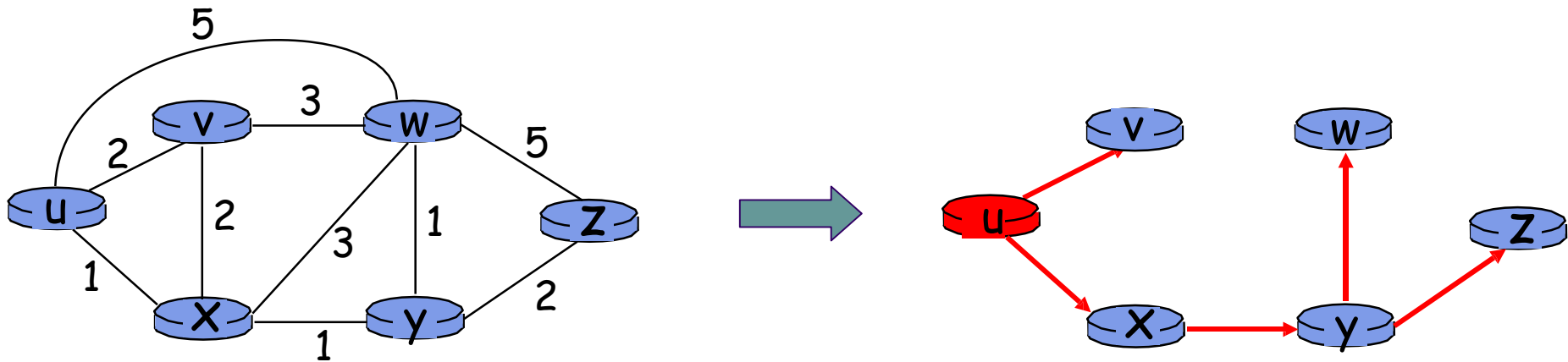


Biểu diễn mạng bởi đồ thị

- Đồ thị với các nút (bộ định tuyến) và các cạnh (liên kết)
- Chi phí cho việc sử dụng mỗi liên kết $c(x,y)$
 - Băng thông, độ trễ, chi phí, mức độ tắc nghẽn...
- Giải thuật chọn đường: Xác định đường đi ngắn nhất giữa hai nút bất kỳ



Cây đường đi ngắn nhất - SPT



- SPT – Shortest Path Tree
- Các cạnh xuất phát từ nút gốc và tới các lá
- Đường đi duy nhất từ nút gốc tới nút v, là đường đi ngắn nhất giữa nút gốc và nút v
- Mỗi nút sẽ có một SPT của riêng nút đó



Tập trung hay phân tán

- Tập trung
 - Thu thập thông tin vào một nút mạng
 - Sử dụng các **giải thuật tìm đường đi trên đồ thị**
 - Phân bổ băng chọn đường từ nút trung tâm tới các nút
- Phân tán
 - Mỗi nút tự xây dựng bảng chọn đường riêng
 - **Giao thức chọn đường**: Link-state hoặc distance-vector
 - Được sử dụng phổ biến trong thực tế



Tập trung hay phân tán

- Thông tin chọn đường là cần thiết để xây dựng bảng chọn đường
- Tập trung hay phân tán?
 - Tập trung:
 - Mỗi router có thông tin đầy đủ về trạng thái của mạng
 - Giải thuật dạng “link state”
 - Phân tán:
 - Các nút chỉ biết được trạng thái của liên kết vật lý tới nút kế bên
 - Liên tục lặp lại việc tính toán và trao đổi thông tin với nút kế bên
 - Giải thuật dạng “distance vector”
 - “Bạn của bạn cũng là bạn”



Giải thuật dạng link-state

Giải thuật Dijkstra's

- Mỗi nút đều có sơ đồ và chi phí mỗi link
 - Quảng bá “Link-state”
- Tìm đường đi chi phí nhỏ nhất từ một nút (‘nguồn’) tới tất cả các nút khác
 - dùng để xây dựng bảng chọn đường



Ký hiệu

- $G = (V, E)$: đồ thị với tập đỉnh V và tập cạnh E
- $c(x, y)$: chi phí của liên kết x tới y ; $= \infty$ nếu không phải 2 nút kề nhau
- $d(v)$: chi phí hiện thời của đường đi từ nút nguồn tới nút đích. v
- $p(v)$: nút ngay trước nút v trên đường đi từ nguồn tới đích
- T : Tập các nút mà đường đi ngắn nhất đã được xác định



Các thủ tục

- **Init():**

Với mỗi nút v , $d[v] = \infty$, $p[v] = \text{NIL}$

$d[s] = 0$

- **Improve(u, v)**, trong đó (u, v) là một cạnh nào đó của G

if $d[v] > d[u] + c(u, v)$ then

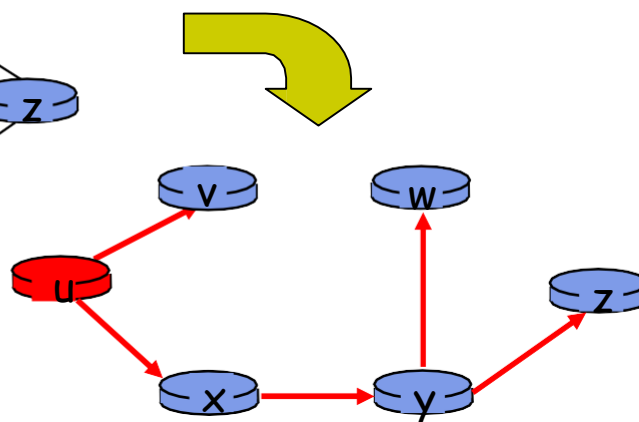
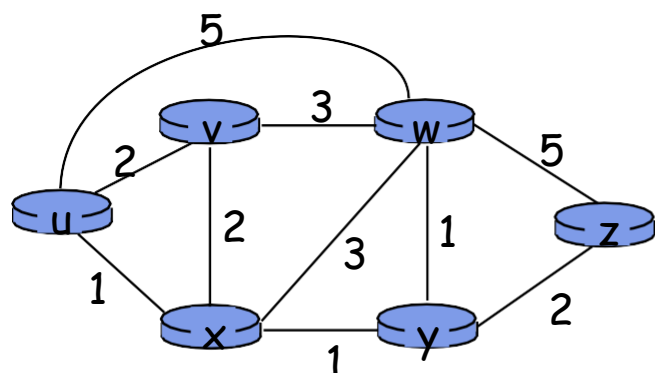
$d[v] = d[u] + c(u, v)$

$p[v] = u$

Dijkstra's algorithm: Ví dụ



Step	T	$d(v), p(v)$	$d(w), p(w)$	$d(x), p(x)$	$d(y), p(y)$	$d(z), p(z)$
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

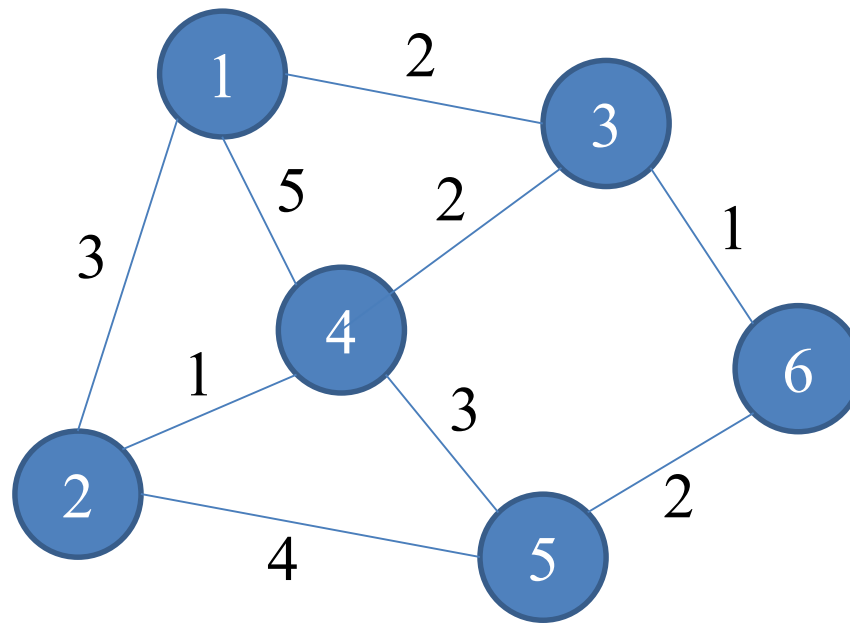


SPT của u:

Bảng chọn đường của u:

destination	link
v	(u,v)
x.	(u,x)
y.	(u,x)
w	(u,x)
z	(u,x ³) ⁵

Ví dụ



#	N	D2	D3	D4	D5	D6	P2	P3	P4	P5	P6
	{1}	3	2	5	∞	∞	1	1	1	1	1
1	{1, 3}	3	2	4	∞	3	1	1	3	1	3
2	{1, 3, 2}	3		4	7	3	1		3	2	3
3	{1, 3, 2, 6}			4	5	3			3	6	3
4	{1, 3, 2, 6, 4}			4	5				3	6	
5	{1, 3, 2, 6, 4, 5}				5					6	



Giải thuật dạng distance-vector

Được dùng trong ARPNet và RIP

Ban đầu:

Mỗi node biết thông tin về khoảng cách (distance, cost) tới các node hàng xóm

Kết thúc giải thuật:

Khoảng cách tới tất cả các node được biết bao gồm thông tin về node kế tiếp

Đề thực thi giải thuật distance-vector

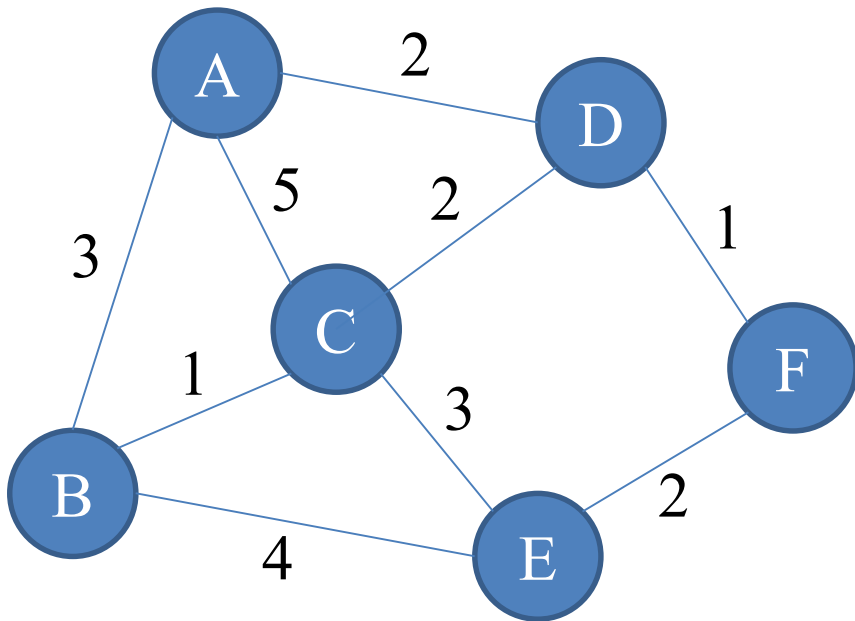


1. Thông tin gì được trao đổi?
2. Khi nhận được thông tin thì các node làm gì?
3. Khi nào thì thông tin được gửi đi?

Thông tin bảng định tuyến



- Mỗi node có 1 bảng định tuyến (distance-vector)
 - Thông tin về đích đến
 - Ước lượng chi phí đến đích
 - Node kế tiếp để đi đến đích



Bảng định tuyến lúc khởi tạo

Destination	Cost	Next-hop
A	0	A
B	3	B
C	5	C
D	2	D

Thông tin trao đổi giữa routers



- Một phần thông tin trong bảng định tuyến
 1. Thông tin về đích đến
 2. Ước lượng chi phí đến đích
 3. ~~Node kế tiếp để đi đến đích~~

Router tính đường đi dựa trên thông tin nhận được thế nào?



Phương trình Bellman-Ford

$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

$d_x(y)$:= chi phí của đường đi ngắn nhất từ x tới y

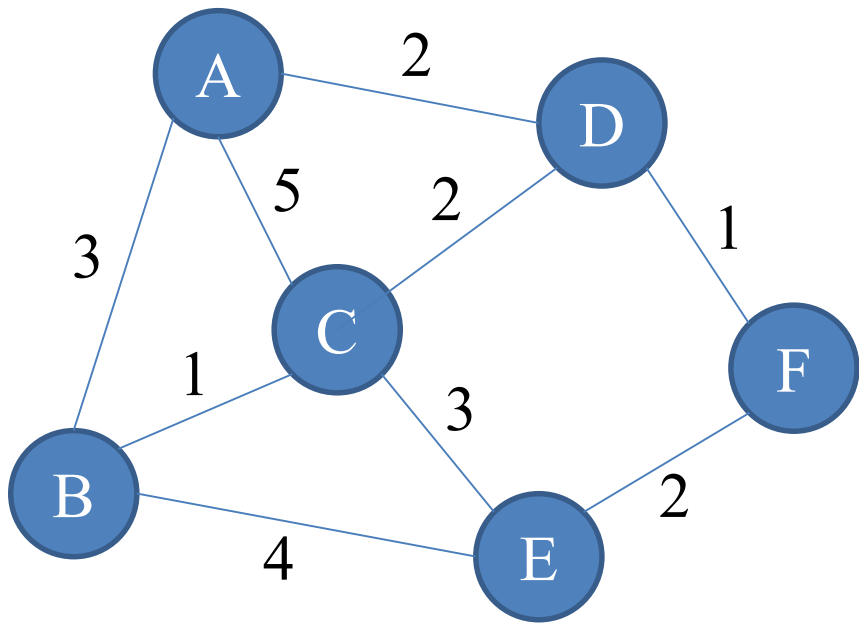
Áp dụng phương trình Bellman-Ford cho tất cả các v là hàng xóm của $x \rightarrow d_x(y)$

Các bước thực thi trên router



- Khi nhận thông tin từ router hàng xóm v
 - Cập nhật thông tin về chi phí tới đích theo phương trình Bellman-Ford
 - Thay đổi thông tin node kế tiếp tương ứng
- Lặp lại tính toán với mỗi đích đến trong bảng định tuyến
- Sau chuỗi thông tin định tuyến trao đổi, tính chi phí tối ưu nhất.

Minh họa – Node A + thông tin từ node B



Dest	Cost	NH
A	0	A
B	3	B
C	5	C
D	2	D

Bảng định tuyến
node A

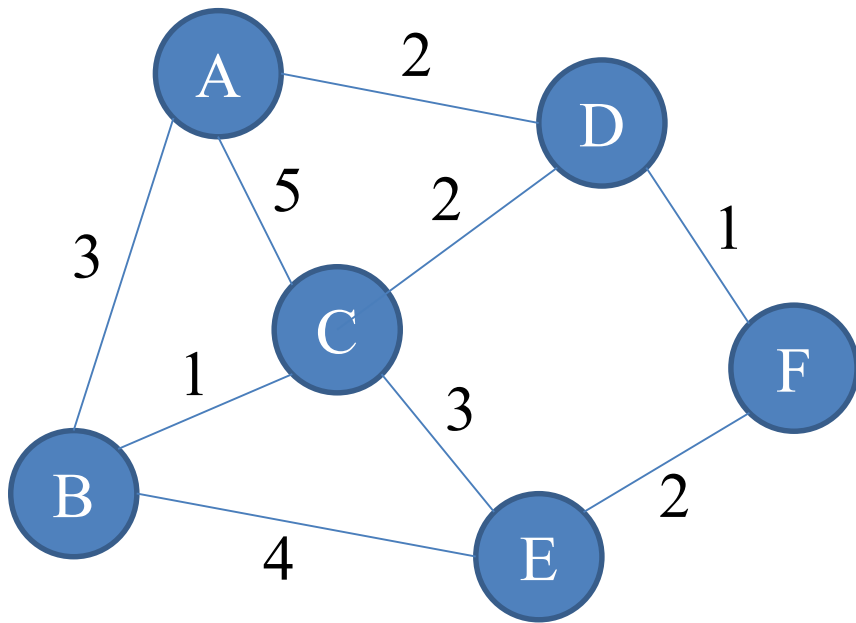
Dest	Cost
B	0
A	3
C	1
E	4

Thông tin từ
node **B**



Dest	Cost	NH
A	0	A
B	3	B
C	5 4	B
D	2	D
E	7	B

Minh họa – Node A + thông tin từ node C



Dest	Cost	NH
A	0	A
B	3	B
C	5	C
D	2	D

Dest	Cost
C	0
A	5
B	1
E	3
D	2

Thông tin
từ
node **C**

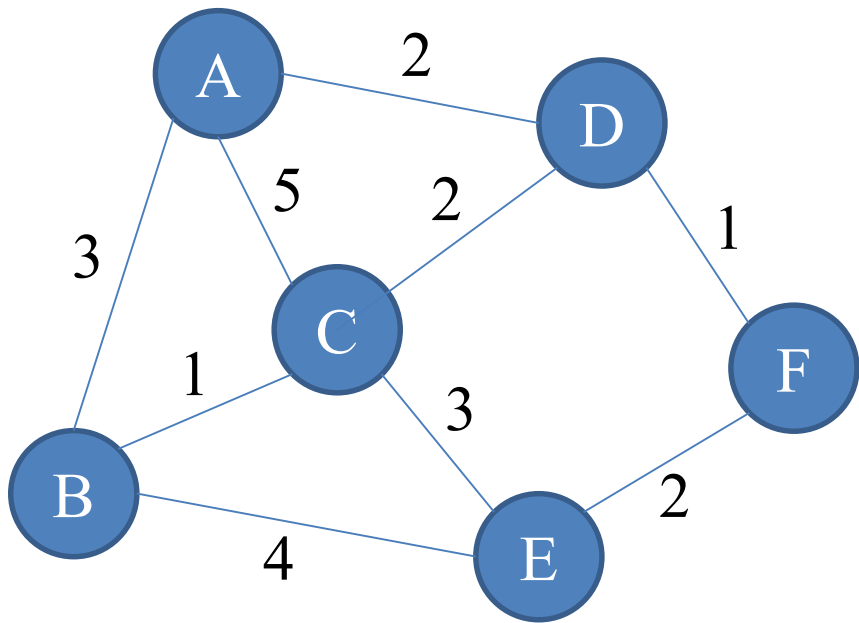
Bảng định tuyến
node A



Dest	Cost	NH
A	0	A
B	3	B
C	4	C
D	2	D
E	7	B

3

Minh họa – Node A + thông tin từ node D



Dest	Cost	NH
A	0	A
B	3	B
C	5	C
D	2	D

Dest	Cost
D	0
A	2
C	2
F	1

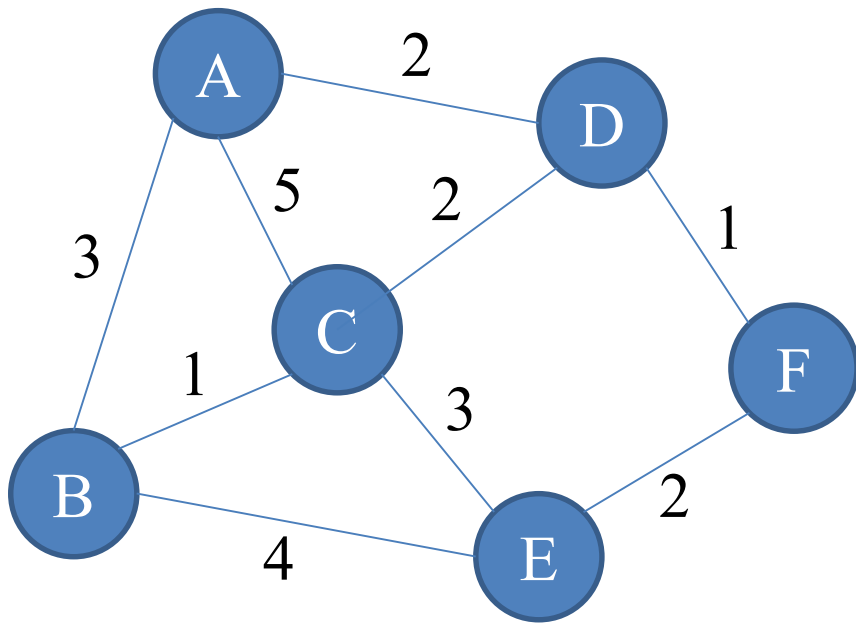
Thông tin
từ
node **D**

Bảng định tuyến
node A



Dest	Cost	NH
A	0	A
B	3	B
C	4	C
D	2	D
E	7	B
F	3	D

Minh họa – Node B + thông tin từ node A



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E

Bảng định tuyến
node B

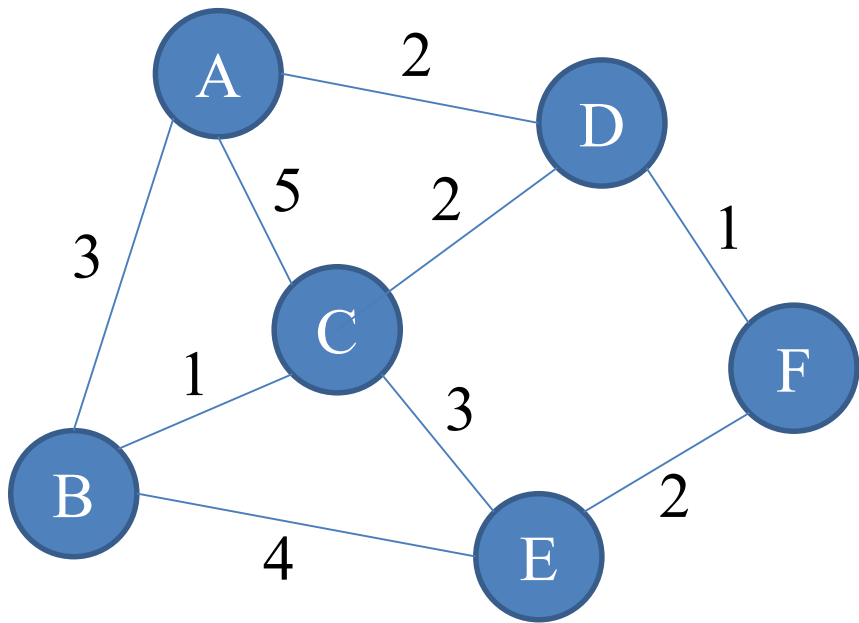


Dest	Cost
A	0
B	3
C	4
D	2
E	7
F	3

Thông tin
từ
node A

Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	5	A
F	6	A

Minh họa – Node B + thông tin từ node C



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	5	A
F	6	A

Bảng định tuyến
node B



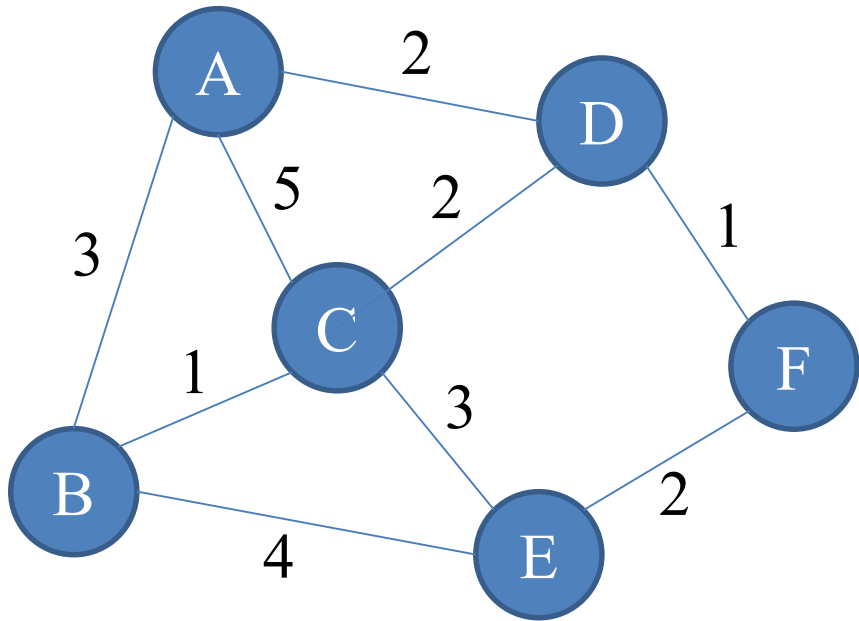
Dest	Cost
C	0
A	5
B	1
E	3
D	2

Thông tin
từ
node **C**

Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	5 3	C
F	6	A

3

Minh họa – Node B + thông tin từ node E



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	3	A
F	6	A

Bảng định tuyến
node B

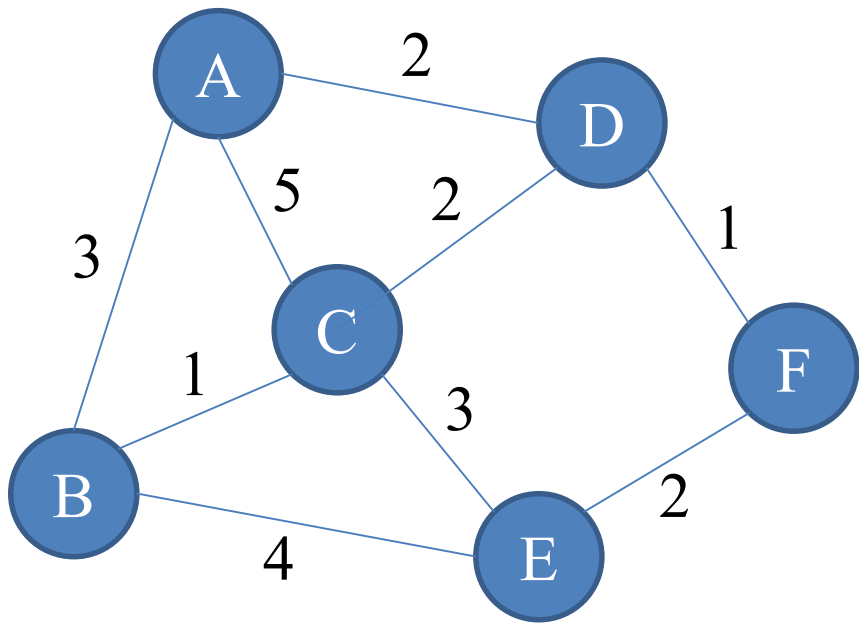


Dest	Cost
E	0
B	4
C	3
F	2

Thông tin
từ
node **E**

Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	3	A
F	6	A

Minh họa – Node C + thông tin từ node A



Dest	Cost	NH
C	0	C
A	5	A
B	1	B
D	2	D
E	3	E

Bảng định tuyến
node D

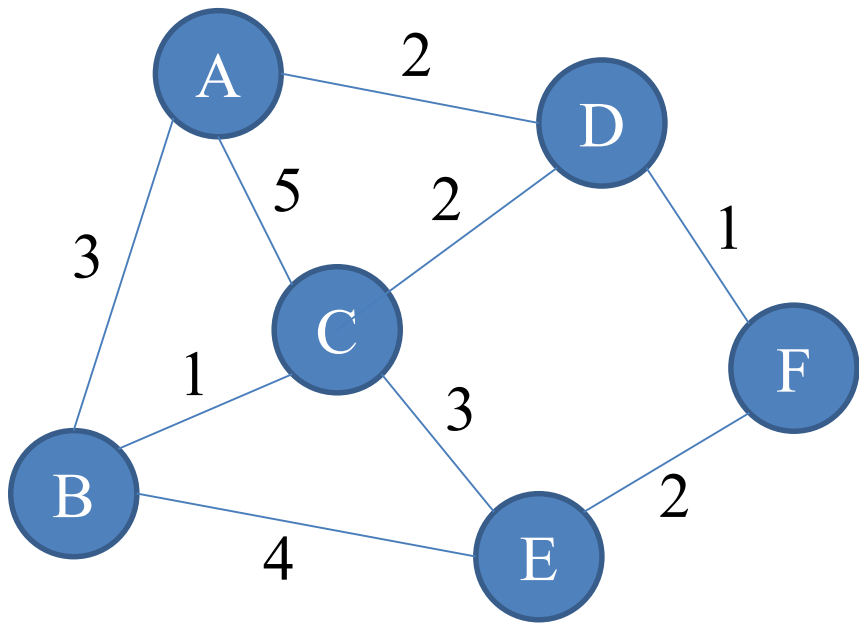
Dest	Cost
A	0
B	3
C	4
D	2
E	7
F	3

Thông tin
từ
node A



Dest	Cost	NH
C	0	C
A	5 4	A B
B	1	B
D	2	D
E	3	E
F	8	A

Minh họa – Node C + thông tin từ node B

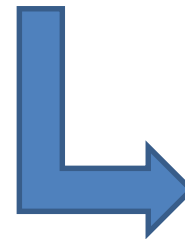


Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	8	A

Dest	Cost
B	0
A	3
C	1
E	4
D	3
F	6

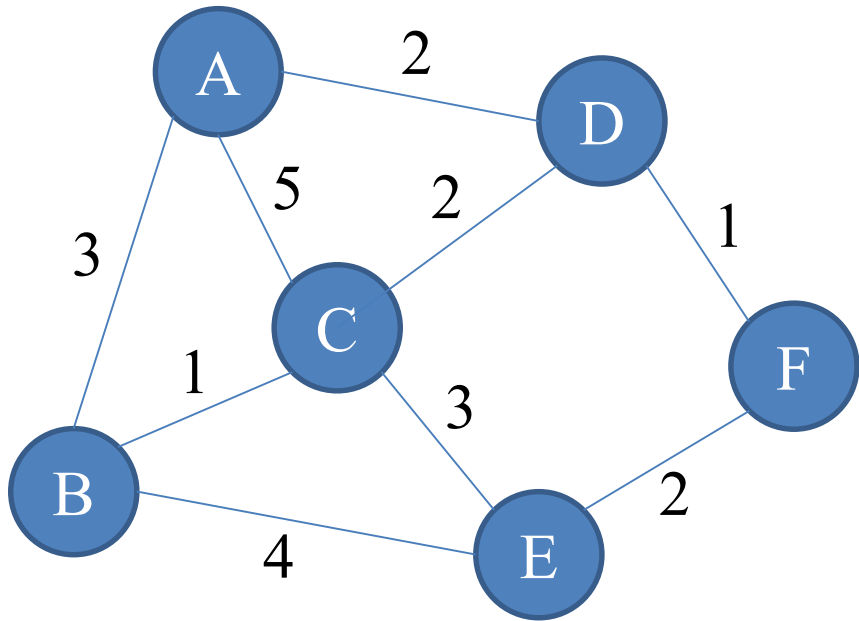
Thông tin
từ
node **B**

Bảng định tuyến
node C



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	8 7	B

Minh họa – Node C + thông tin từ node E



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	7	B

Bảng định tuyến
node C

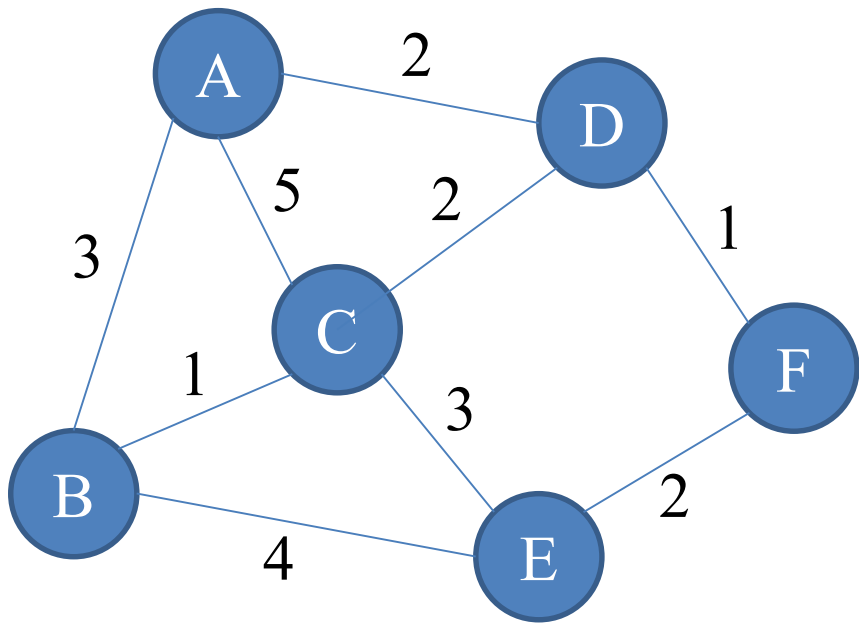


Dest	Cost
E	0
B	4
C	3
F	2

Thông tin
từ
node **E**

Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	7 5	E

Minh họa – Node C + thông tin từ node D



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	5	E

Bảng định tuyến
node C

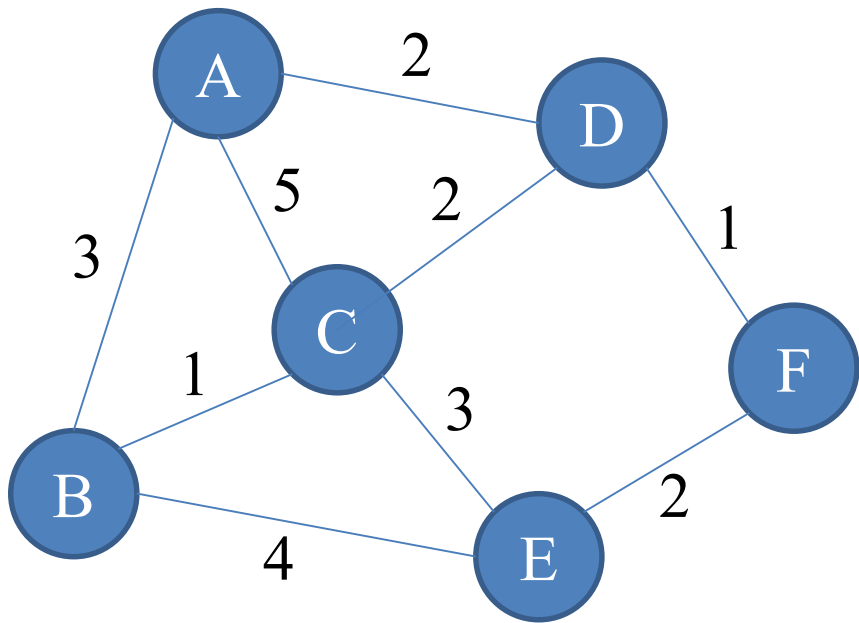


Dest	Cost
E	0
A	2
C	2
F	1

Thông tin
từ
node **D**

Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	5 3	D

Minh họa – Node D + thông tin từ node A



Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F

Bảng định tuyến
node D

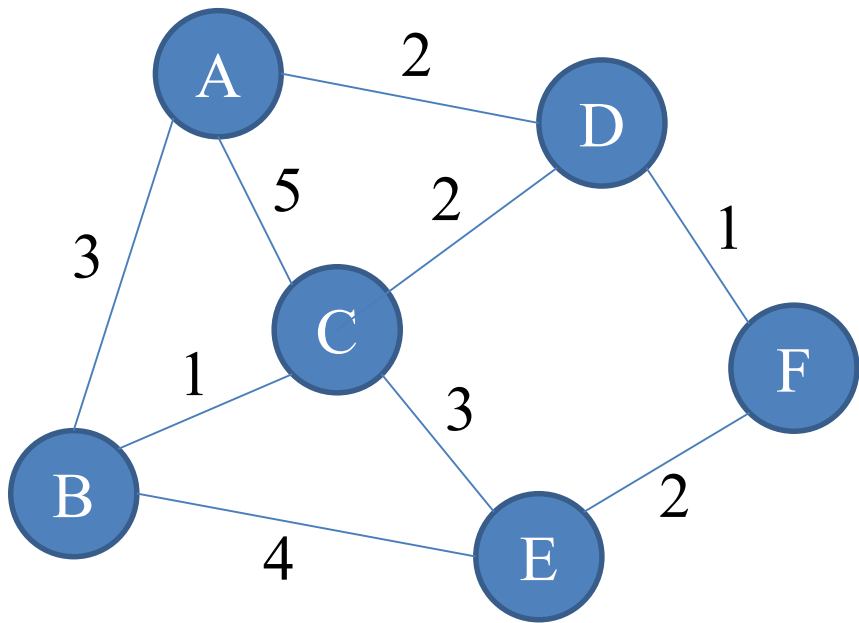


Dest	Cost
A	0
B	3
C	4
D	2
E	7
F	3

Thông tin
từ
node A

Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	5	A
E	9	A

Minh họa – Node D + thông tin từ node C



Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	5	A
E	9	A

Dest	Cost
C	0
A	4
B	1
D	2
E	3
F	3

Thông tin
từ
node **C**

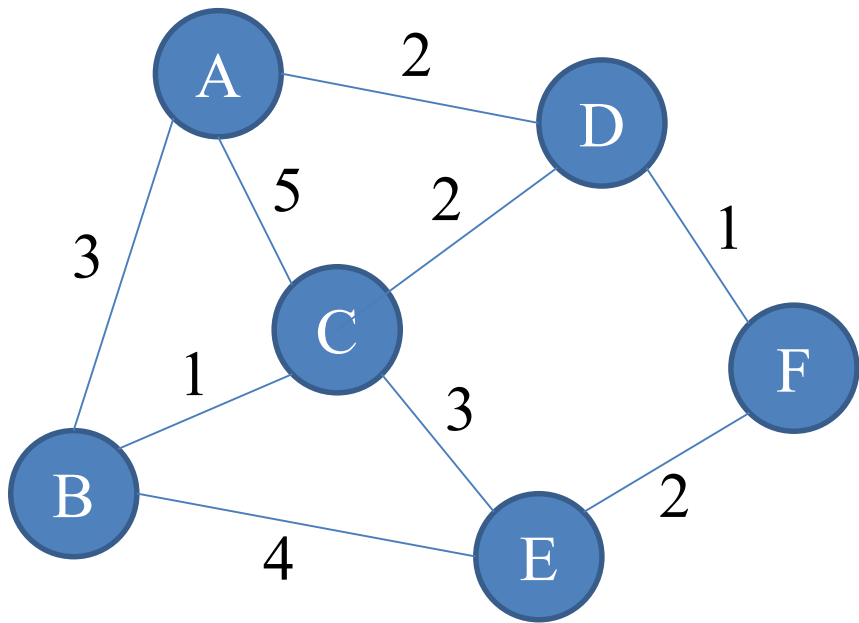
Bảng định tuyến
node D



Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	5 3	C
E	9 5	C

3

Minh họa – Node D + thông tin từ node F



Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	3	C
E	5	C

Bảng định tuyến
node D

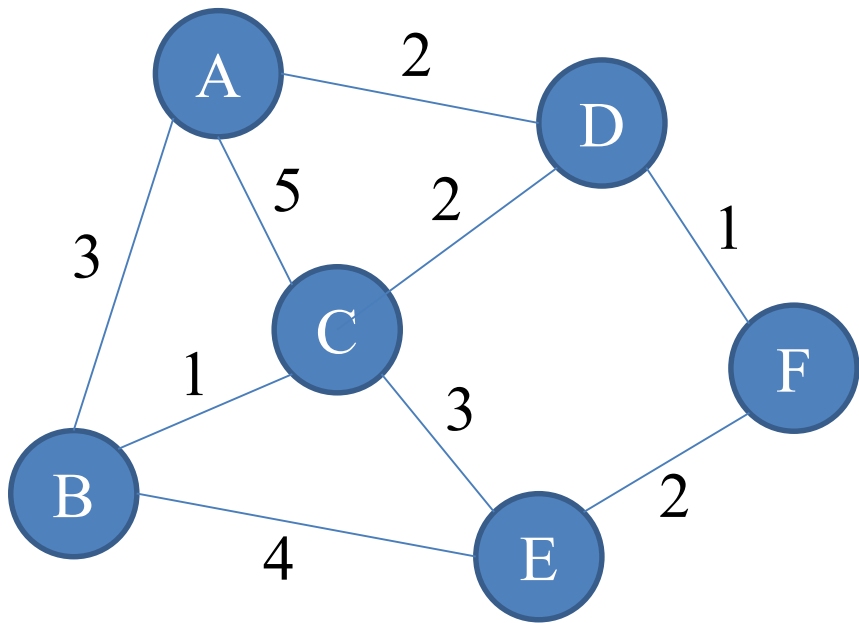


Dest	Cost
F	0
E	2
D	1

Thông tin
từ
node **F**

Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	3	C
E	5 3	F

Minh họa – Node E + thông tin từ node B

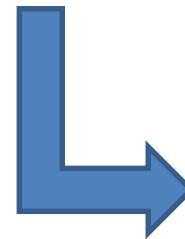


Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F

Bảng định tuyến
node E

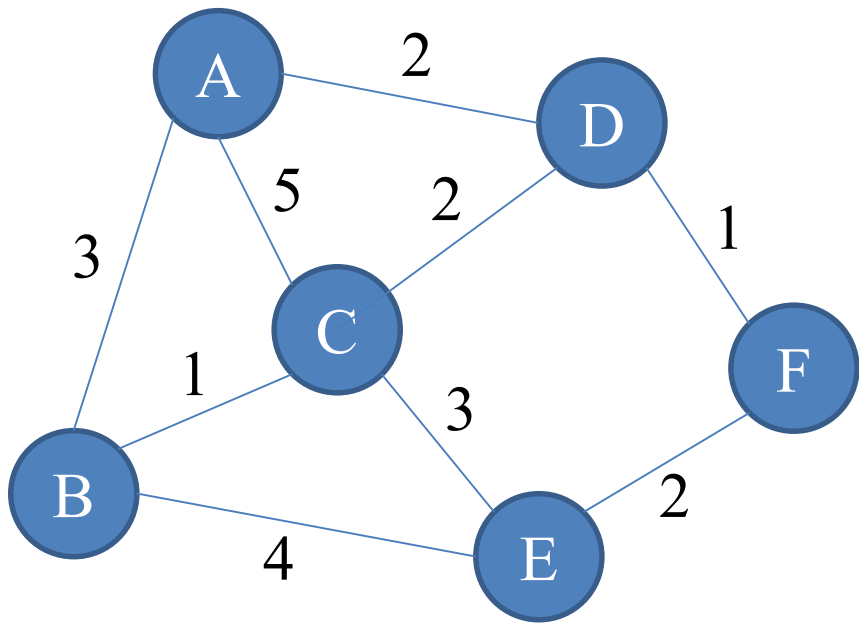
Dest	Cost
B	0
A	3
C	1
E	4
D	3
F	6

Thông tin
từ
node **B**



Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	7	B

Minh họa – Node E + thông tin từ node C



Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	7	B

Bảng định tuyến
node E

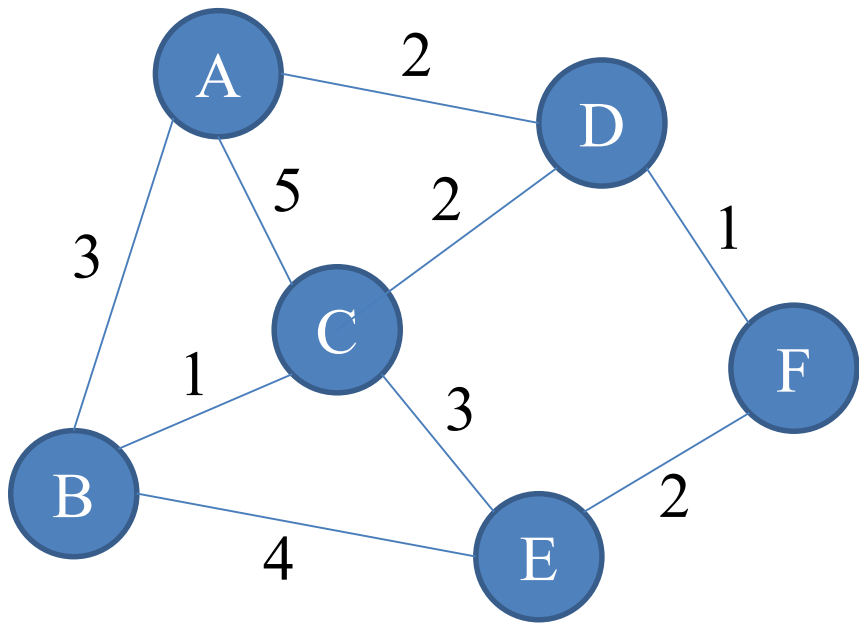


Dest	Cost
C	0
A	4
B	1
D	2
E	3
F	3

Thông tin
từ
node **C**

Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	7 5	B

Minh họa – Node E + thông tin từ node F



Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	5	C

Bảng định tuyến
node E

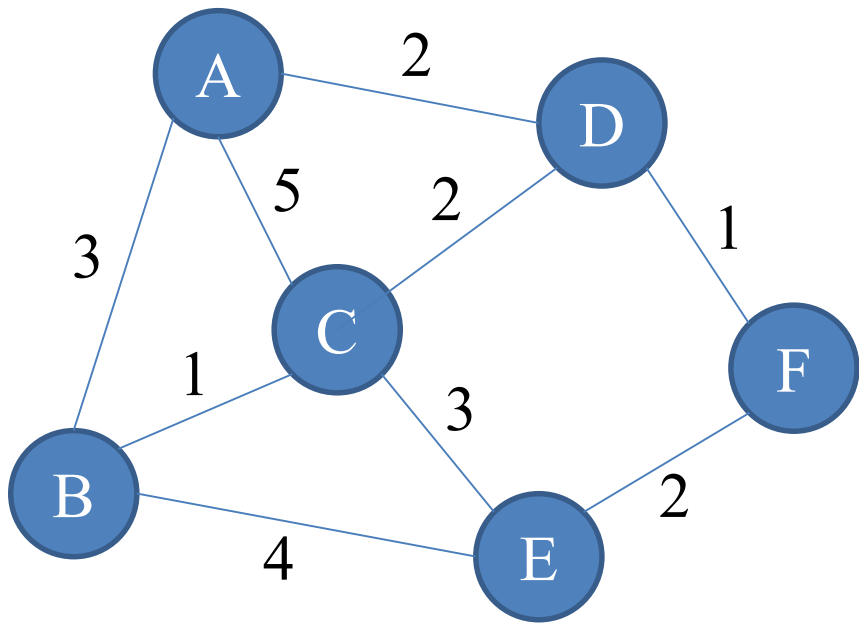


Dest	Cost
F	0
E	2
D	1

Thông tin
từ
node **F**

Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	5 3	F

Minh họa – Node F + thông tin từ node E



Dest	Cost	NH
F	0	F
E	2	E
D	1	D

Bảng định tuyến
node F

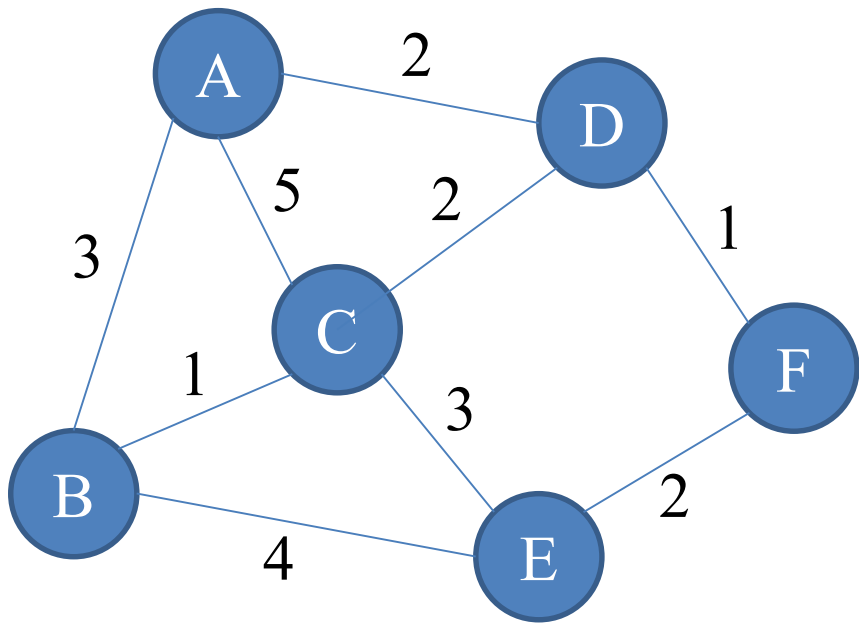


Dest	Cost
E	0
B	4
C	3
F	2
A	7
D	3

Thông tin
từ
node **E**

Dest	Cost	NH
F	0	F
E	2	E
D	1	D
A	9	E
B	6	E
C	5	E

Minh họa – Node F + thông tin từ node D



Dest	Cost	NH
F	0	F
E	2	E
D	1	D
A	9	E
B	6	E
C	5	E

Bảng định tuyến
node F

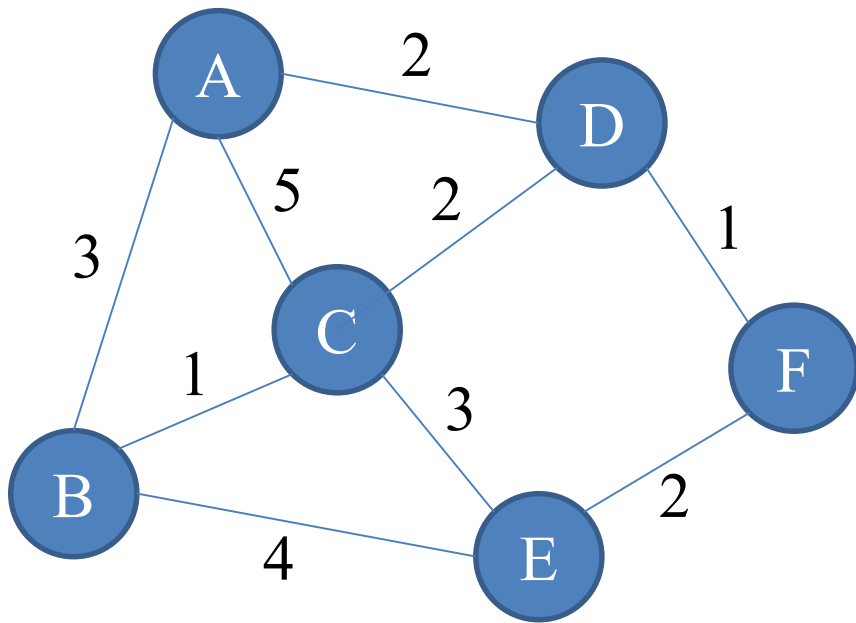


Dest	Cost
D	0
A	2
C	2
F	1
B	3
E	3

Thông tin
từ
node **D**

Dest	Cost	NH
F	0	F
E	2	E
D	1	D
A	9 3	D
B	6 4	D
C	5 3	D

Minh họa – Node A + thông tin từ node B



Dest	Cost	NH
A	0	A
B	3	B
C	4	C
D	2	D
E	7	B
F	3	D

Bảng định tuyến
node A

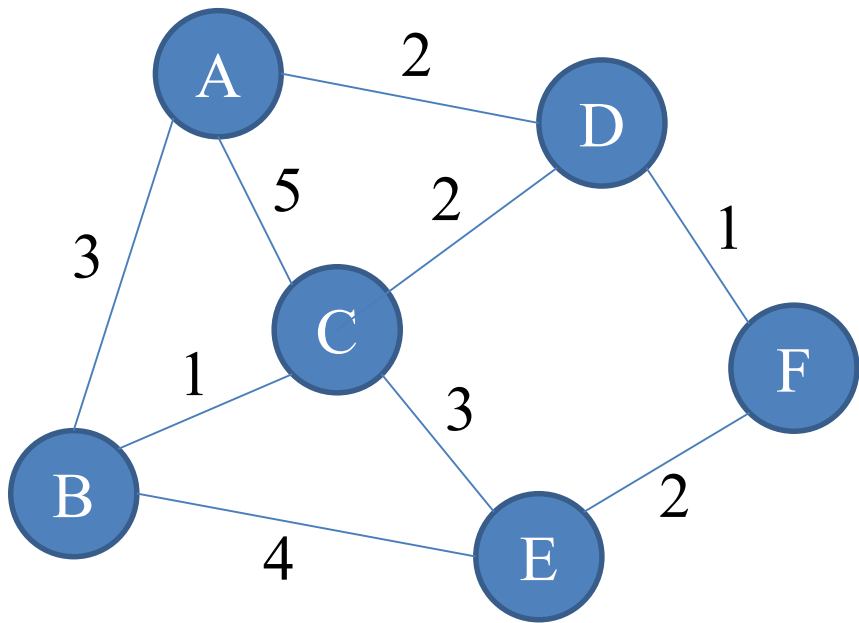
Dest	Cost
B	0
A	3
C	1
E	4
D	3
F	6

Thông tin
từ
node **B**



Giữ nguyên !

Minh họa – Node A + thông tin từ node C



Dest	Cost	NH
A	0	A
B	3	B
C	4	C
D	2	D
E	7	B
F	3	D

Bảng định tuyến
node A

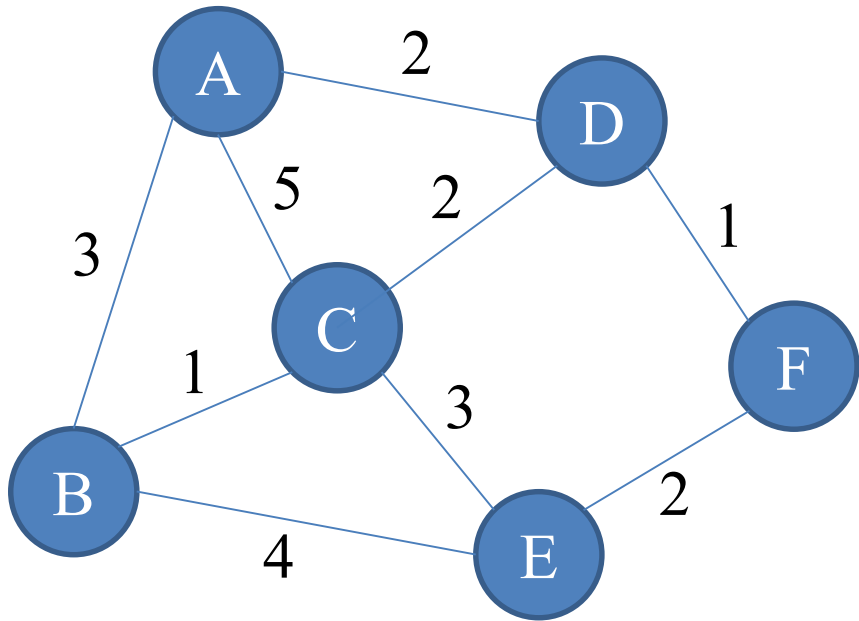
Dest	Cost
C	0
A	4
B	1
D	2
E	3
F	3

Thông tin
từ
node C



Giữ nguyên !

Minh họa – Node A + thông tin từ node D



Dest	Cost	NH
A	0	A
B	3	B
C	4	C
D	2	D
E	7	B
F	3	D

Bảng định tuyến
node A

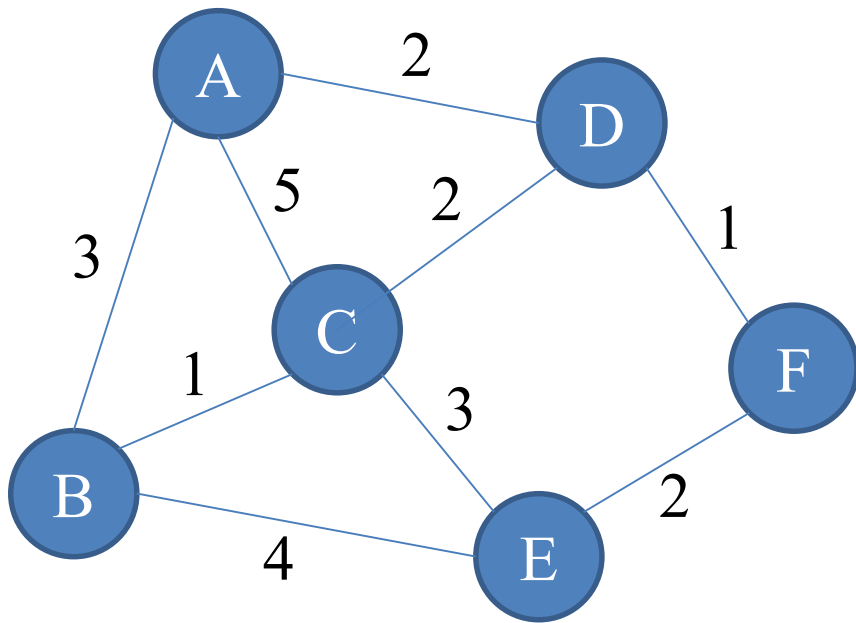


Dest	Cost
D	0
A	2
C	2
F	1
B	3
E	3

Thông tin
từ
node **D**

Dest	Cost	NH
A	0	A
B	3	B
C	4	C
D	2	D
E	7 5	B
F	3	D

Minh họa – Node B + thông tin từ node A



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	3	C
F	6	C

Bảng định tuyến
node B

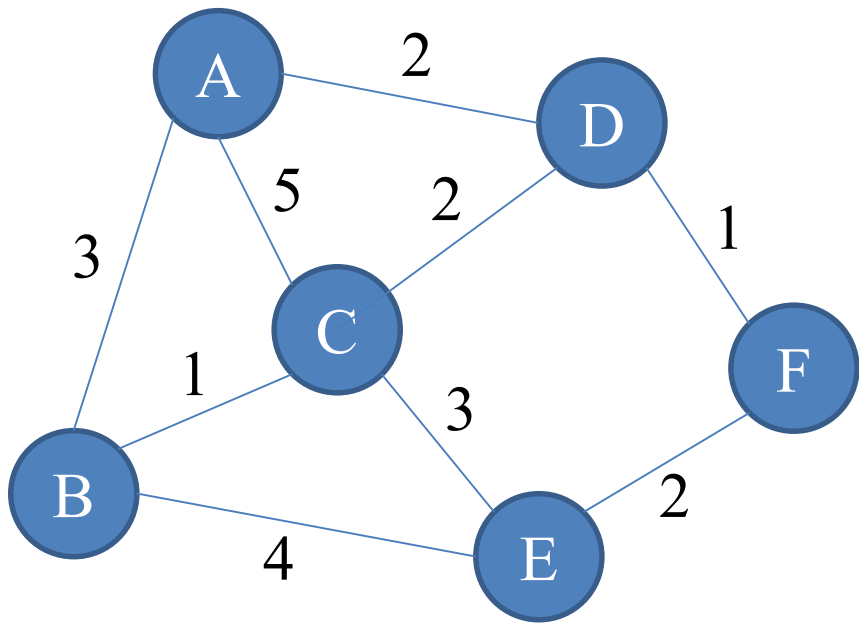
Dest	Cost
A	0
B	3
C	4
D	2
E	5
F	3

Thông tin
từ
node A



Giữ nguyên !

Minh họa – Node B + thông tin từ node C



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	3	C
F	6	A

Dest	Cost
C	0
A	4
B	1
D	2
E	3
F	3

Thông tin
từ
node **C**

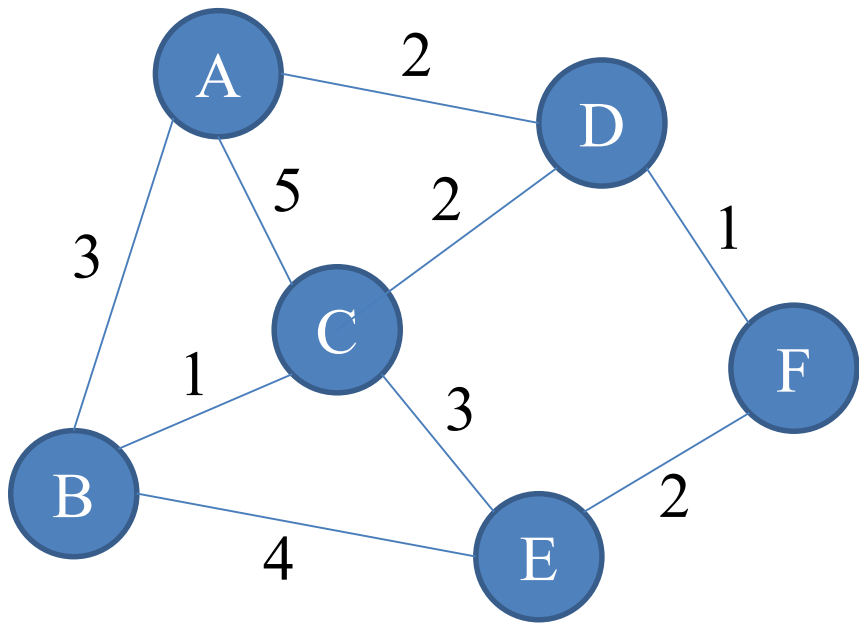
Bảng định tuyến
node B



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	3	C
F	6 4	C

4

Minh họa – Node B + thông tin từ node E



Dest	Cost	NH
B	0	B
A	3	A
C	1	C
E	4	E
D	3	A
F	4	C

Dest	Cost
E	0
B	4
C	3
F	2
A	7
D	3

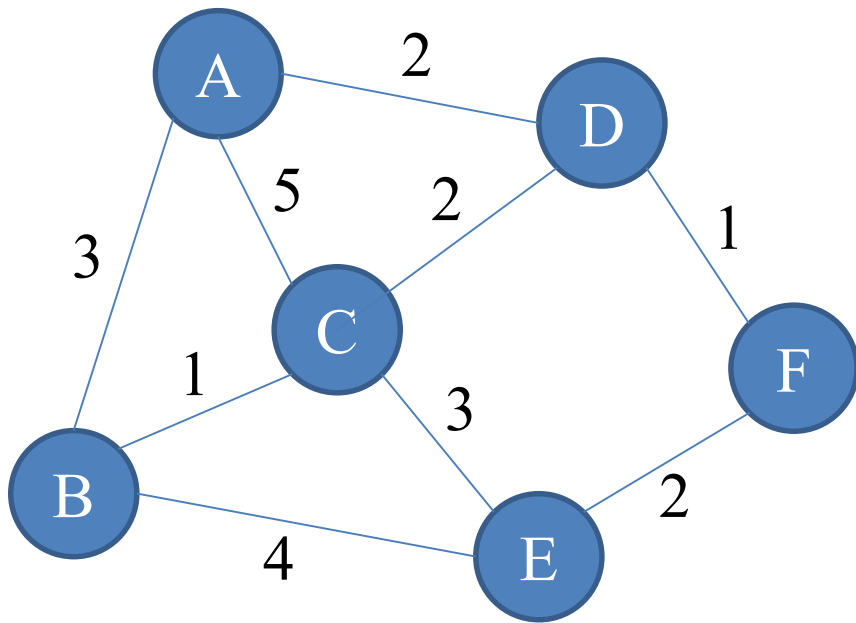
Thông tin
từ
node **E**

Bảng định tuyến
node B



Giữ nguyên !

Minh họa – Node C + thông tin từ node A



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	3	D

Bảng định tuyến
node C

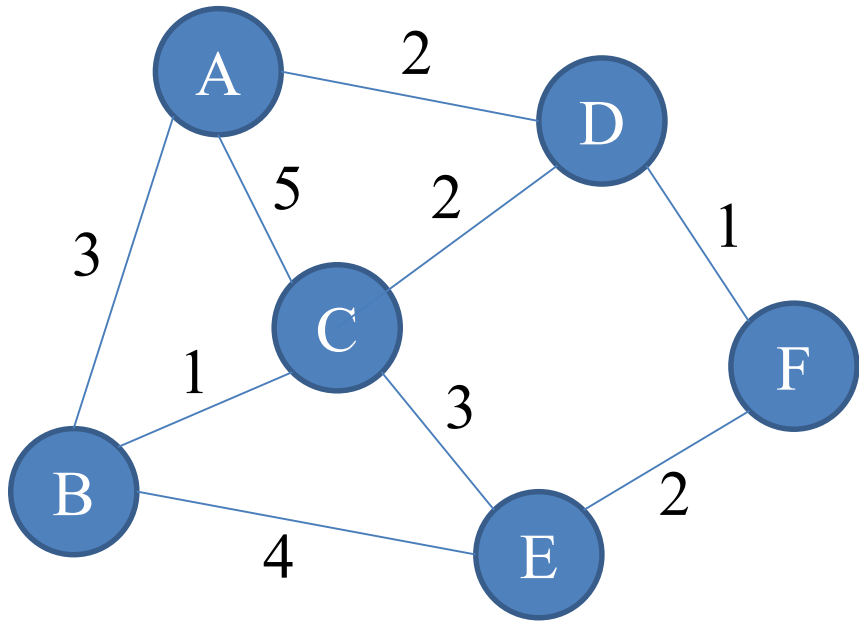
Dest	Cost
A	0
B	3
C	4
D	2
E	5
F	3

Thông tin
từ
node A



Giữ nguyên !

Minh họa – Node C + thông tin từ node B



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	3	D

Bảng định tuyến
node C

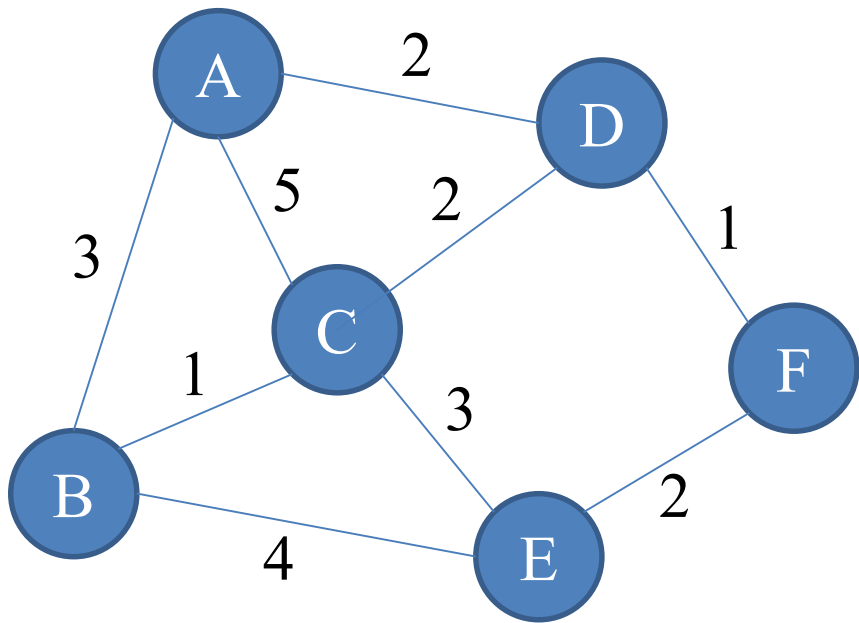
Dest	Cost
B	0
A	3
C	1
E	4
D	3
F	4

Thông tin
từ
node **B**



Giữ nguyên !

Minh họa – Node C + thông tin từ node E

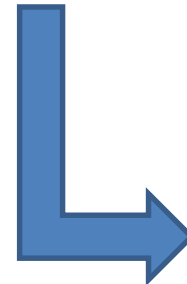


Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	3	D

Dest	Cost
E	0
B	4
C	3
F	2
A	7
D	3

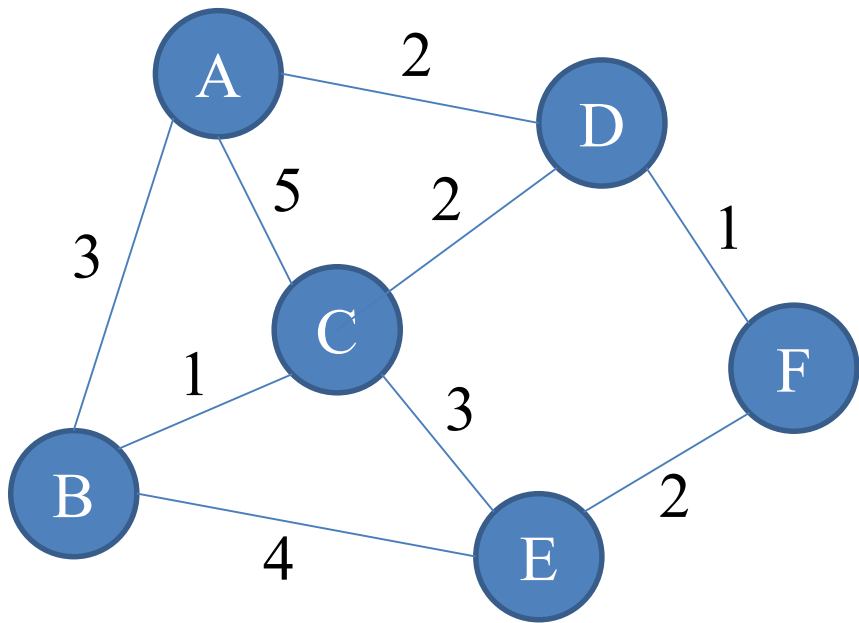
Thông tin
từ
node **E**

Bảng định tuyến
node B



Giữ nguyên !

Minh họa – Node C + thông tin từ node D



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	5	E

Dest	Cost
D	0
A	2
C	2
F	1
B	3
E	3

Thông tin
từ
node **D**

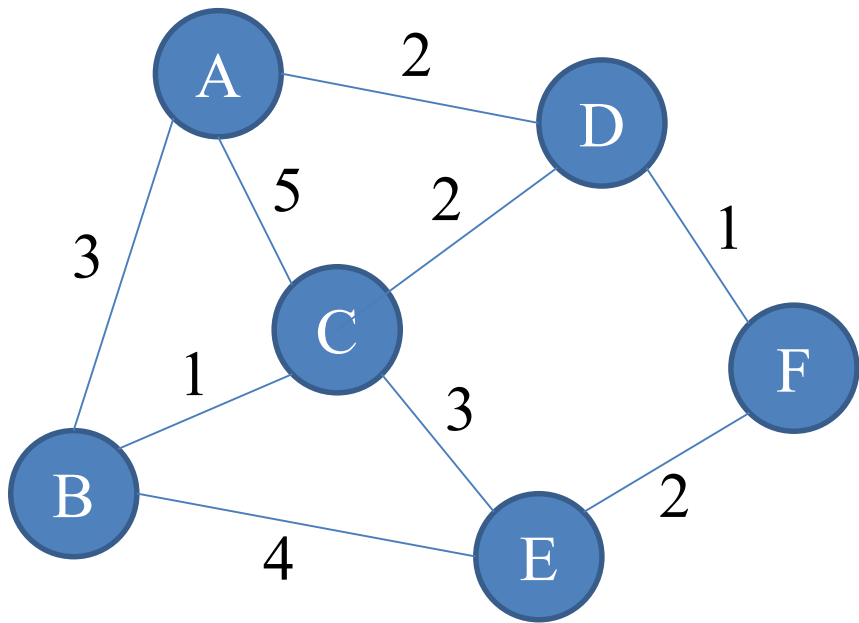
Bảng định tuyến
node B



Dest	Cost	NH
C	0	C
A	4	B
B	1	B
D	2	D
E	3	E
F	5 3	D

3

Minh họa – Node D + thông tin từ node A



Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	3	C
E	3	F

Bảng định tuyến
node D

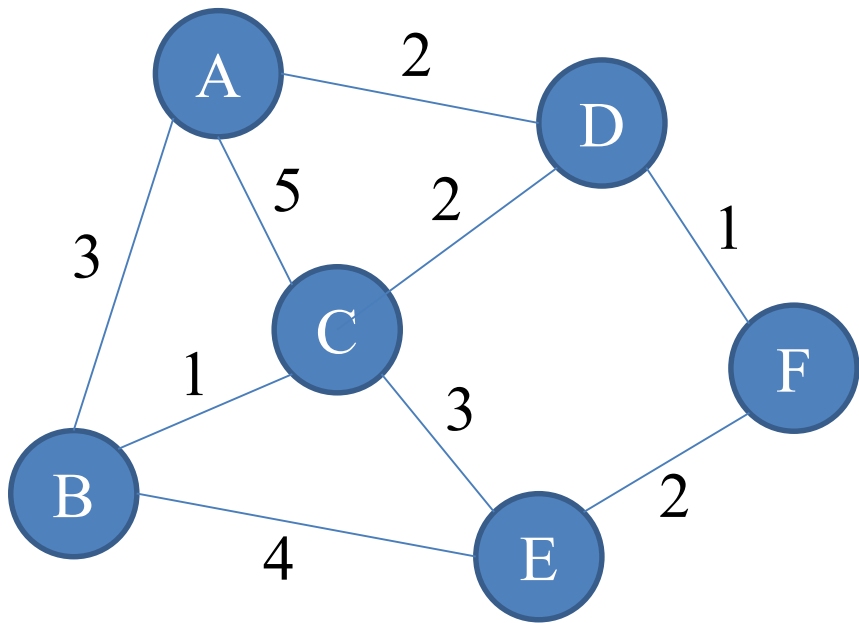
Dest	Cost
A	0
B	3
C	4
D	2
E	5
F	3

Thông tin
từ
node A



Giữ nguyên !

Minh họa – Node D + thông tin từ node C

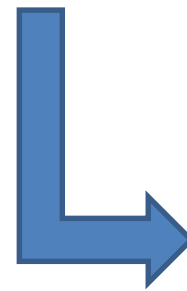


Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	3	C
E	3	F

Dest	Cost
C	0
A	4
B	1
D	2
E	3
F	3

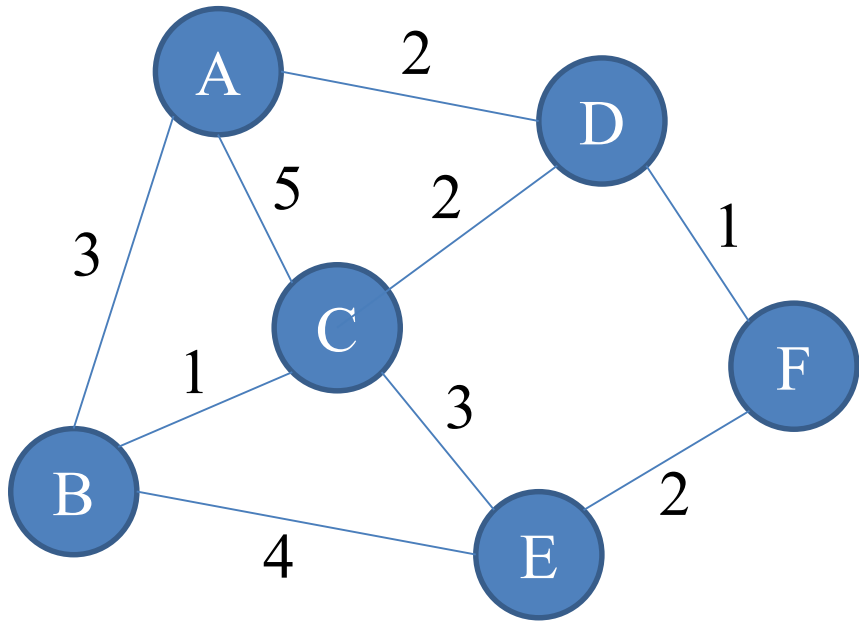
Thông tin
từ
node **C**

Bảng định tuyến
node D



Giữ nguyên !

Minh họa – Node D + thông tin từ node F



Dest	Cost	NH
D	0	D
A	2	A
C	2	C
F	1	F
B	3	C
E	3	F

Dest	Cost
F	0
E	2
D	1
A	3
B	4
C	3

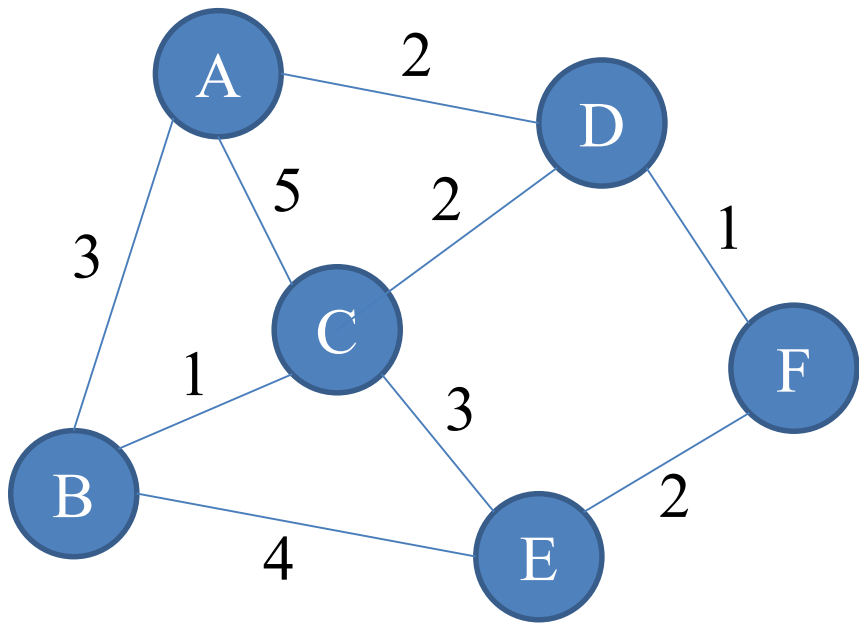
Thông tin
từ
node **F**

Bảng định tuyến
node D



Giữ nguyên !

Minh họa – Node E + thông tin từ node B



Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	3	F

Bảng định tuyến
node E

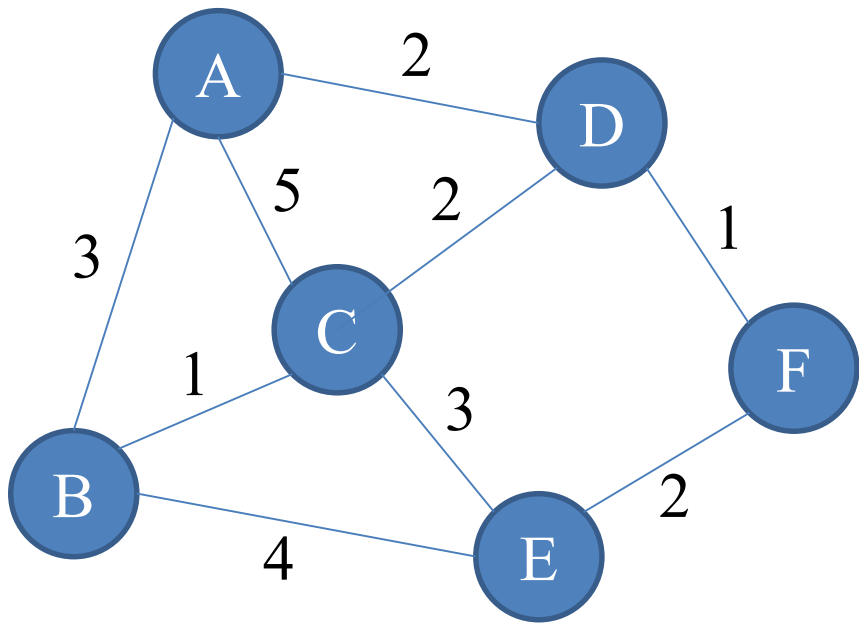
Dest	Cost
B	0
A	3
C	1
E	4
D	3
F	4

Thông tin
từ
node **B**



Giữ nguyên !

Minh họa – Node E + thông tin từ node C



Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	3	F

Bảng định tuyến
node E

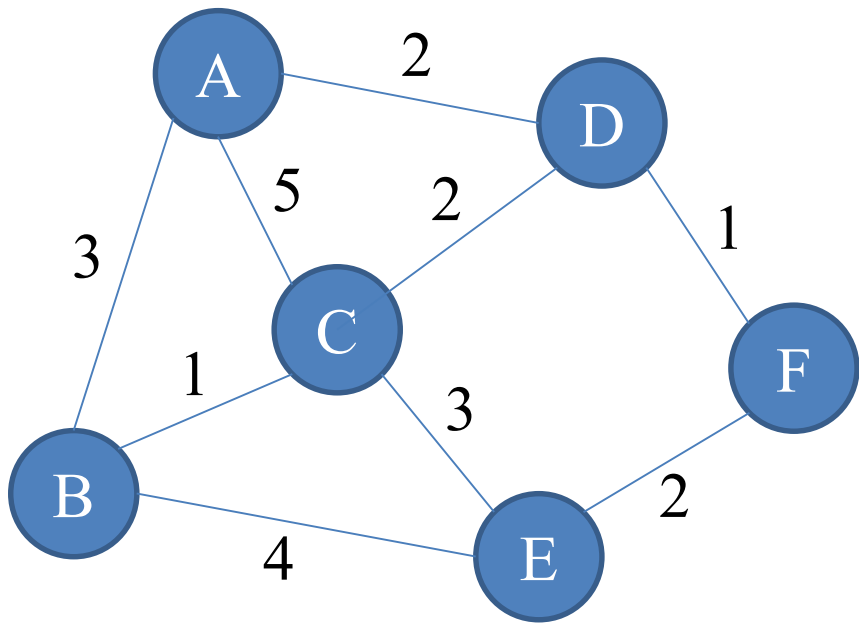
Dest	Cost
C	0
A	4
B	1
D	2
E	3
F	3

Thông tin
từ
node C



Giữ nguyên !

Minh họa – Node E + thông tin từ node F



Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7	B
D	3	F

Bảng định tuyến
node E

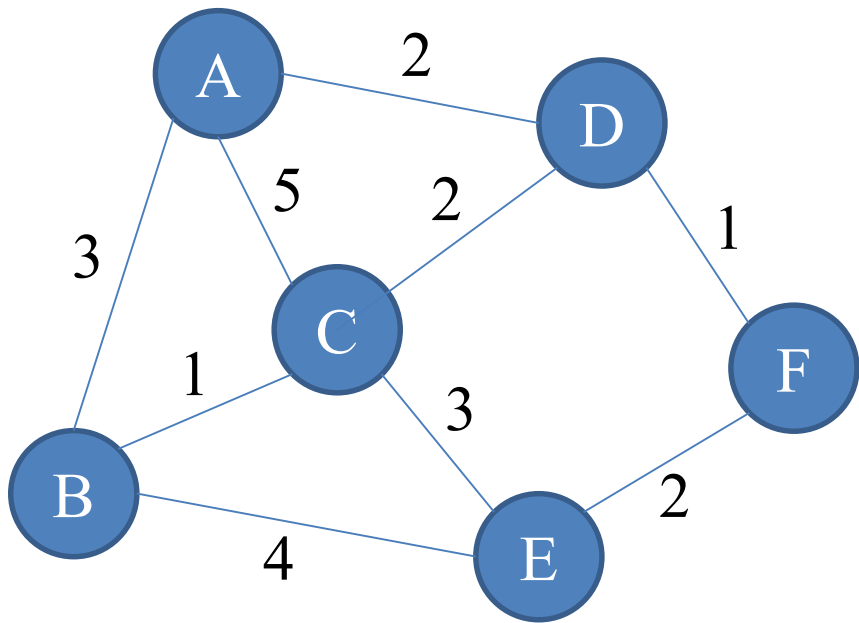


Dest	Cost
F	0
E	2
D	1
A	3
B	4
C	3

Thông tin
từ
node **F**

Dest	Cost	NH
E	0	E
B	4	B
C	3	C
F	2	F
A	7 5	F
D	3	F

Minh họa – Node F + thông tin từ node E



Dest	Cost	NH
F	0	F
E	2	E
D	1	D
A	3	D
B	4	D
C	3	D

Bảng định tuyến
node F

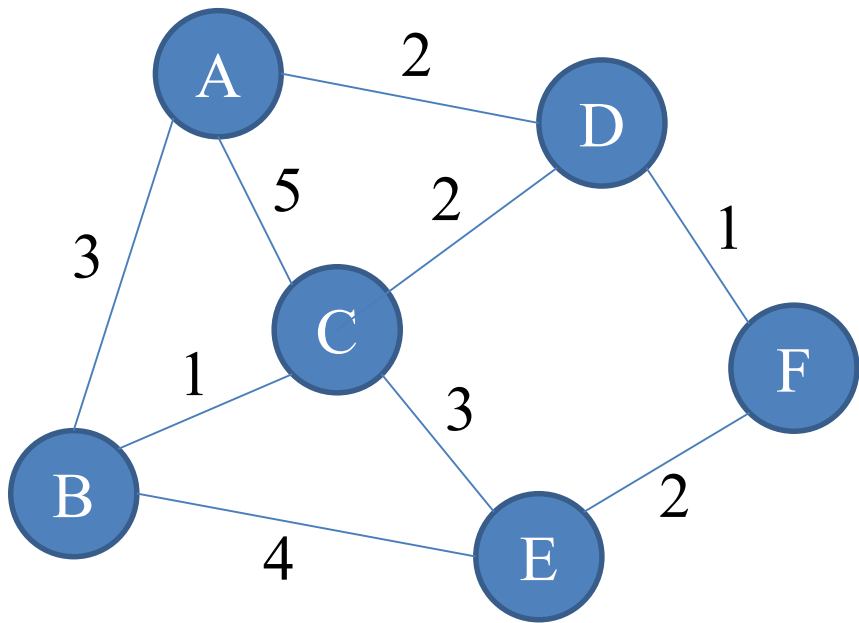
Dest	Cost
E	0
B	4
C	3
F	2
A	5
D	3

Thông tin
từ
node E



Giữ nguyên !

Minh họa – Node F + thông tin từ node D



Dest	Cost	NH
F	0	F
E	2	E
D	1	D
A	3	D
B	4	D
C	3	D

Bảng định tuyến
node F

Dest	Cost
D	0
A	2
C	2
F	1
B	3
E	3

Thông tin
từ
node **D**



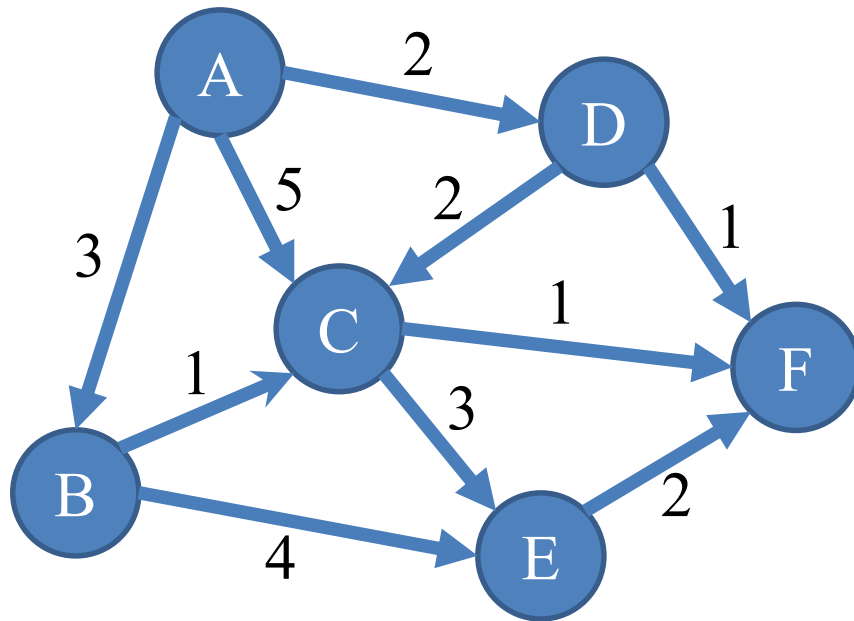
Giữ nguyên ! ₇₇

Khi nào gửi thông tin định tuyến?

- Gửi ngay khi có sự kiện
 - Link/node lỗi
 - Chi phí thay đổi
- Gửi định kỳ
 - Báo cho các node khác, mình vẫn sống
 - Cập nhật thông tin về chi phí (distance vector)
 - Chu kỳ: vài giây → vài phút

Distance **Vector**

$$d(u) > d(v) + c(u,v) \\ \rightarrow d(u) = d(v) + c(u,v) \\ \text{Bellman Ford}$$



$$A = 0$$

$$d(B) = \infty > d(A) + c(A,B) = 0 + 3$$

$$d(C) = \infty, 5, 4$$

$$d(D) = \infty, 2$$

$$d(E) = \infty, 7$$

$$d(F) = \infty, 3$$

(A,B) (A,C) (A,D) (B,C) (B, E) (C, D) (C,E) (D, F) (E, F)

So sánh các giải thuật LS và DV



Thông điệp trao đổi

- LS: n nút, E cạnh, $O(nE)$ thông điệp
- DV: Chỉ trao đổi giữa các hàng xóm
 - Thời gian hội tụ thay đổi

Tốc độ hội tụ

- LS: Thuật toán: $O(n^2)$ cần $O(nE)$ thông điệp
- DV: Thay đổi

Sự chắc chắn: Giải sử một router hoạt động sai

LS:

- nút gửi các chi phí sai
- Mỗi nút tính riêng bằng chọn đường \rightarrow có vẻ chắc chắn hơn

DV:

- DV có thể bị gửi sai
- Mỗi nút tính toán dựa trên các nút khác
 - Lỗi bị lan truyền trong mạng

Tóm tắt



- Nguyên lý của bài toán chọn đường
- Tĩnh vs. động, tập trung vs. phân tán
- Link-state vs. distance-vector

Tuần tới: Các giao thức chọn đường trên Internet



- Chọn đường phân cấp
- RIP
- OSPF
- BGP