

# LECTURE 12

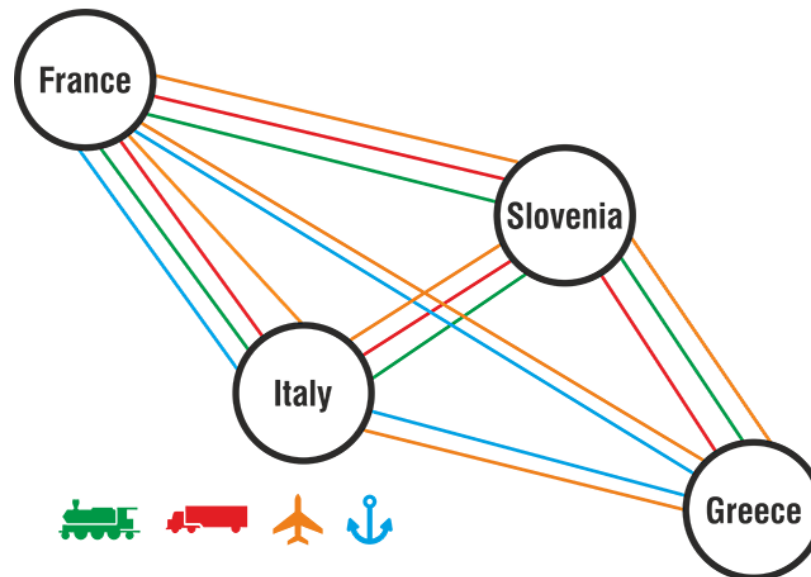
## FLOYD-WARSHALL ALGORITHM

Phạm Nguyễn Sơn Tùng

Email: [sontungtn@gmail.com](mailto:sontungtn@gmail.com)

# Định nghĩa Floyd-Warshall

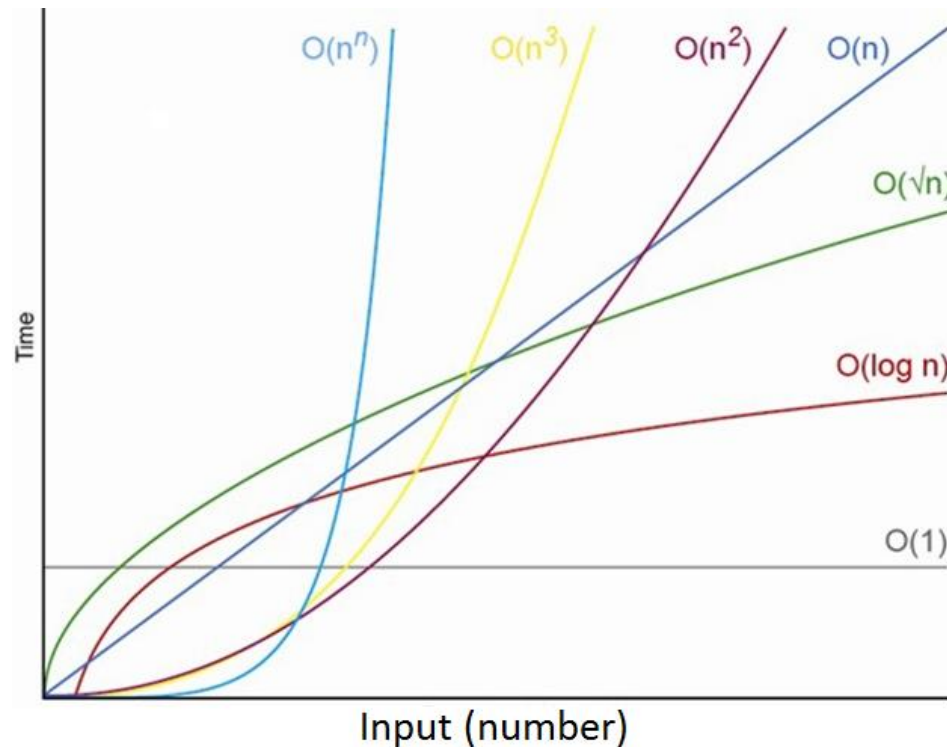
Thuật toán **Floy Warshall** là thuật toán tìm đường đi ngắn nhất của **tất cả các cặp đỉnh** trong đồ thị **có hướng** có trọng số (trọng số có thể dương **hoặc âm**).



# Độ phức tạp của Floyd-Warshall

Độ phức tạp:  $O(V^3)$

- Với **V (Vertex)** là số đỉnh.



# Ý tưởng của thuật toán

Thuật toán dùng phương pháp Quy Hoạch Động (Dynamic Programming) lưu tất cả các kết quả có được ban đầu vào ma trận.

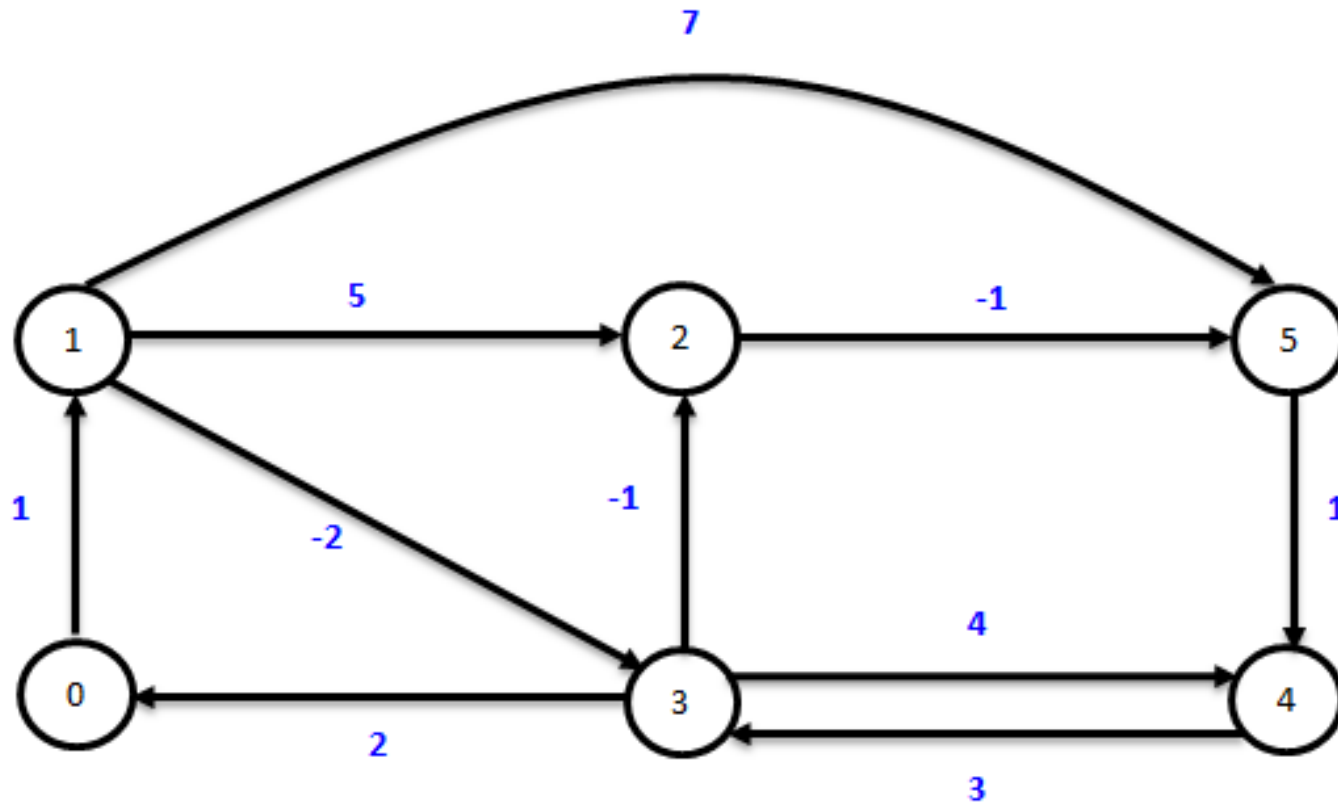
Nếu tồn tại một cách để đi từ nút a đến nút b và cách để đi từ nút b đến nút c, thì sẽ có cách đi từ nút a đến nút c.

Dò tìm đường đi bằng cách thử qua các đỉnh trung gian, nếu phát hiện đường đi mới ngắn hơn thì cập nhật đường đi mới và chỉnh sửa các thông tin cần thiết.

Trong quá trình cập nhật chi phí, tiến hành **lưu đỉnh cha** của đỉnh kè. Kết quả lưu lại cuối cùng là đường đi với chi phí nhỏ nhất đi từ đỉnh xuất phát đến tất cả các đỉnh trong đồ thị.

# Bài toán minh họa

Cho đồ thị **có hướng** như hình vẽ. Tìm đường đi **ngắn nhất** từ **đỉnh 0** đến **tất cả** các đỉnh khác.



# BƯỚC 0: CHUẨN BỊ DỮ LIỆU

# Bước 0: Chuẩn bị dữ liệu (1)

Từ **ma trận kề** cho trước, chuyển thông tin vào **các cấu trúc dữ liệu** cần thiết.

6					
0	1	$\infty$	$\infty$	$\infty$	$\infty$
$\infty$	0	5	-2	$\infty$	7
$\infty$	$\infty$	0	$\infty$	$\infty$	-1
2	$\infty$	-1	0	4	$\infty$
$\infty$	$\infty$	$\infty$	3	0	$\infty$
$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 0: Chuẩn bị dữ liệu (2)

Chuyển ma trận kề vào **graph**.

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0



# Bước 0: Chuẩn bị dữ liệu (3)

Ma trận chứa chi phí đường đi **dist** chuyển từ ma trận **graph**.

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

## Bước 0: Chuẩn bị dữ liệu (4)

Ma trận lưu vết đường đi **path**, đỉnh nào có liên kết với nhau thì giá trị cha đỉnh đó là chính nó, ngược lại là -1.

	0	1	2	3	4	5
0	-1	0	-1	-1	-1	-1
1	-1	-1	1	1	-1	1
2	-1	-1	-1	-1	-1	2
3	3	-1	3	-1	3	-1
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1

# **BƯỚC 1**

## **CHẠY VÒNG LẶP DUYỆT QUA MA**

### **TRẬN KÈ LẦN 1**

# Bước 1: Chạy thuật toán (**k=0**)

$$\underset{0}{\text{dist}[i][j]} > \underset{0}{\text{dist}[i][k]} + \underset{0}{\text{dist}[k][j]} \quad \times$$

**j=0**

**i=0**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j] \quad \times$$

1

0

1

j=1

i=0

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\underset{\infty}{\text{dist}[i][j]} > \underset{0}{\text{dist}[i][k]} + \underset{\infty}{\text{dist}[k][j]} \quad \times$$

**j=2**

**i=0**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\underset{\infty}{\text{dist}[i][j]} > \underset{0}{\text{dist}[i][k]} + \underset{\infty}{\text{dist}[k][j]} \quad \text{X}$$

**j=3**

**i=0**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\underset{\infty}{\text{dist}[i][j]} > \underset{0}{\text{dist}[i][k]} + \underset{\infty}{\text{dist}[k][j]} \quad \times$$

**j=4**

**i=0**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0



# Bước 1: Chạy thuật toán (**k=0**)

$$\underset{\infty}{\text{dist}[i][j]} > \underset{0}{\text{dist}[i][k]} + \underset{\infty}{\text{dist}[k][j]} \quad \times$$

**j=5**

**i=0**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j] \quad \times$$

$\infty$

0

$\infty$

**j=0**

**i=1**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j] \quad \times$$

 $\infty$ 

0

 $\infty$ **j=0****i=2**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

**1**  $\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j]$  ✓

$\infty$                       2                      1

**j=1**

**i=3**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	$\infty \rightarrow 3$	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

**1**  $\text{path}[i][j] = \text{path}[k][j]$

Cập nhật giá trị đỉnh cha trong mảng **path[i][j] = 0**.

**j=1**

**i=3**

	0	1	2	3	4	5
0	-1	0	-1	-1	-1	-1
1	-1	-1	1	1	-1	1
2	-1	-1	-1	-1	-1	2
3	3	-1->0	3	-1	3	-1
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1

# Bước 1: Chạy thuật toán (**k=0**)

$$\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j] \quad \times$$

**j=0**

**i=4**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# Bước 1: Chạy thuật toán (**k=0**)

$$\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j] \quad \times$$

**j=0**

**i=5**

	0	1	2	3	4	5
0	0	1	$\infty$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

# **BƯỚC 2**

## **CHẠY VÒNG LẶP DUYỆT QUA MA**

### **TRẬN KÈ LẦN 2**



## Bước 2: Chạy thuật toán (**k=1**)

**1**  $\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j]$  ✓

$\infty$                       1                      5

j=2

i=0

	0	1	2	3	4	5
0	0	1	$\infty \rightarrow 6$	$\infty$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

## Bước 2: Chạy thuật toán (**k=1**)

**1**  $\text{path}[i][j] = \text{path}[k][j]$

Cập nhật giá trị đỉnh cha trong mảng  $\text{path}[i][j] = 1$ .

**j=2**

**i=0**

	0	1	2	3	4	5
0	-1	0	-1->1	-1	-1	-1
1	-1	-1	1	1	-1	1
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	-1
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1

## Bước 2: Chạy thuật toán (**k=1**)

**2**  $\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j]$  ✓

$\infty$                       1                      -2

**j=3**

**i=0**

	0	1	2	3	4	5
0	0	1	6	$\infty \rightarrow -1$	$\infty$	$\infty$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

## Bước 2: Chạy thuật toán (**k=1**)

**2**  $\text{path}[i][j] = \text{path}[k][j]$

Cập nhật giá trị đỉnh cha trong mảng  $\text{path}[i][j] = 1$ .

**j=3**

**i=0**

	0	1	2	3	4	5
0	-1	0	1	-1->1	-1	-1
1	-1	-1	1	1	-1	1
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	-1
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1

## Bước 2: Chạy thuật toán (**k=1**)

**3**  $\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j]$  ✓

$\infty$                       1                      7

**j=5**

**i=0**

	0	1	2	3	4	5
0	0	1	6	-1	$\infty$	$\infty \rightarrow 8$
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	$\infty$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

## Bước 2: Chạy thuật toán (k=1)

3

$\text{path}[i][j] = \text{path}[k][j]$

Cập nhật giá trị đỉnh cha trong mảng  $\text{path}[i][j] = 1$ .

j=5

i=0

	0	1	2	3	4	5
0	-1	0	1	1	-1	-1->1
1	-1	-1	1	1	-1	1
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	-1
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1

## Bước 2: Chạy thuật toán (**k=1**)

**4**  $\text{dist}[i][j] > \text{dist}[i][k] + \text{dist}[k][j]$  ✓

$\infty$                       3                      7

**j=5**

**i=3**

	0	1	2	3	4	5
0	0	1	6	-1	$\infty$	8
1	$\infty$	0	5	-2	$\infty$	7
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	$\infty \rightarrow 10$
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

## Bước 2: Chạy thuật toán (**k=1**)

**4**  $\text{path}[i][j] = \text{path}[k][j]$

Cập nhật giá trị đỉnh cha trong mảng  $\text{path}[i][j] = 1$ .

**j=5**

**i=3**

	0	1	2	3	4	5
0	-1	0	1	1	-1	1
1	-1	-1	1	1	-1	1
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	-1->1
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1



# **BƯỚC 3**

## **CHẠY VÒNG LẶP DUYỆT QUA MA**

### **TRẬN KÈ LẦN 3**

## Bước 3: Chạy thuật toán (**k=2**)

Có 3 giá trị trong ma trận **dist** cần cập nhật ở khi **k = 2**.

j

	0	1	2	3	4	5
0	0	1	6	-1	$\infty$	(8) 5
1	$\infty$	0	5	-2	$\infty$	(7) 4
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	(10) -2
4	$\infty$	$\infty$	$\infty$	3	0	$\infty$
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

i

## Bước 3: Chạy thuật toán (**k=2**)

Tương tự cũng sẽ có 3 giá trị trong ma trận **path** cần cập nhật khi **k=2**.

j

	0	1	2	3	4	5
0	-1	0	1	1	-1	(1) 2
1	-1	-1	1	1	-1	(1) 2
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	(1) 2
4	-1	-1	-1	4	-1	-1
5	-1	-1	-1	-1	5	-1

i

# **BƯỚC 4**

## **CHẠY VÒNG LẶP DUYỆT QUA CÁC DANH SÁCH KÈ LẦN 4**

# Bước 4: Chạy thuật toán (**k=3**)

Có 11 giá trị trong ma trận **dist** cần cập nhật ở khi **k = 3**.

j

	0	1	2	3	4	5
0	0	1	(6) -2	-1	( $\infty$ ) 3	(5) -3
1	( $\infty$ ) 0	0	(5) -3	-2	( $\infty$ ) 2	(4) -4
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	-2
4	( $\infty$ ) 5	( $\infty$ ) 6	( $\infty$ ) 2	3	0	( $\infty$ ) 1
5	$\infty$	$\infty$	$\infty$	$\infty$	1	0

i

## Bước 4: Chạy thuật toán ( $k=3$ )

Tương tự cũng sẽ có 11 giá trị trong ma trận **path** cần cập nhật khi  $k=3$ .

j

	0	1	2	3	4	5
0	-1	0	(1) 3	1	(-1) 3	(2) 2
1	(-1) 3	-1	(1) 3	1	(-1) 3	(2) 2
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	2
4	(-1) 3	(-1) 0	(-1) 3	4	-1	(-1) 2
5	-1	-1	-1	-1	5	-1

i

# **BƯỚC 5**

## **CHẠY VÒNG LẶP DUYỆT QUA CÁC DANH SÁCH KÈ LẦN 5**

# Bước 5: Chạy thuật toán (**k=4**)

Có 4 giá trị trong ma trận **dist** cần cập nhật ở khi **k = 4**.

j

	0	1	2	3	4	5
0	0	1	-2	-1	3	-3
1	0	0	-3	-2	2	-4
2	$\infty$	$\infty$	0	$\infty$	$\infty$	-1
3	2	3	-1	0	4	-2
4	5	6	2	3	0	1
5	( $\infty$ ) 6	( $\infty$ ) 7	( $\infty$ ) 3	( $\infty$ ) 4	1	0

i



# Bước 5: Chạy thuật toán (**k=4**)

Tương tự cũng sẽ có 4 giá trị trong ma trận **path** cần cập nhật khi **k=4**.

j

	0	1	2	3	4	5
0	-1	0	3	1	3	2
1	3	-1	3	1	3	2
2	-1	-1	-1	-1	-1	2
3	3	0	3	-1	3	2
4	3	0	3	4	-1	2
5	(-1) 3	(-1) 0	(-1) 3	(-1) 4	5	-1

i

# **BƯỚC 6**

## **CHẠY VÒNG LẶP DUYỆT QUA CÁC DANH SÁCH KÈ LẦN 6**

# Bước 6: Chạy thuật toán (**k=5**)

Có 7 giá trị trong ma trận **dist** cần cập nhật ở khi **k = 5**.

j

	0	1	2	3	4	5
0	0	1	-2	-1	(3) -2	-3
1	0	0	-3	-2	(2) -3	-4
2	(∞) 5	(∞) 6	0	(∞) 3	(∞) 0	-1
3	2	3	-1	0	(4) -1	-2
4	5	6	2	3	0	1
5	6	7	3	4	1	0

i

# Bước 6: Chạy thuật toán (**k=5**)

Tương tự cũng sẽ có 7 giá trị trong ma trận **path** cần cập nhật khi **k=5**.

j

	0	1	2	3	4	5
0	-1	0	3	1	(3) 5	2
1	3	-1	3	1	(3) 5	2
2	(-1) 3	(-1) 0	-1	(-1) 4	(-1) 5	2
3	3	0	3	-1	(3) 5	2
4	3	0	3	4	-1	2
5	3	0	3	4	5	-1

i

# Kết quả cuối cùng

Mảng chứa chi phí đường đi tất cả cặp đỉnh **dist**.

	0	1	2	3	4	5
0	0	1	-2	-1	-2	-3
1	0	0	-3	-2	-3	-4
2	5	6	0	3	0	-1
3	2	3	-1	0	-1	-2
4	5	6	2	3	0	1
5	6	7	3	4	1	0







# Kết quả cuối cùng

Mảng lưu vết đỉnh cha của tất cả các cặp đỉnh **path**.

	0	1	2	3	4	5
0	-1	0	3	1	5	2
1	3	-1	3	1	5	2
2	3	0	-1	4	5	2
3	3	0	3	-1	5	2
4	3	0	3	4	-1	2
5	3	0	3	4	5	-1

# In ra đường đi

Tìm đường đi ngắn nhất từ 0 đến 4.

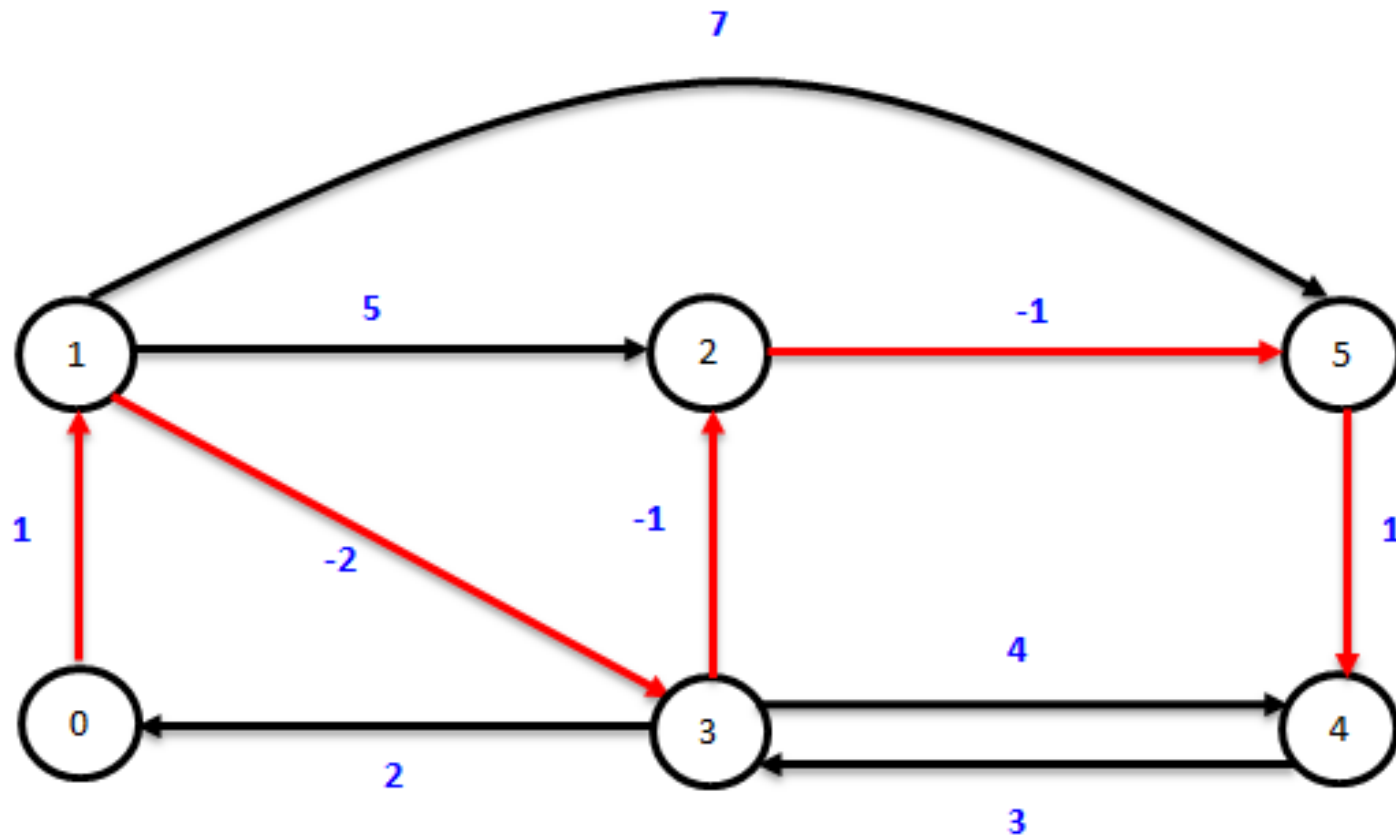
						
	0	1	2	3	4	5
path	-1	0	3	1	5	2

dist	Đỉnh	0	1	2	3	4	5
	Chi phí	0	1	-2	-1	-2	-3

**Kết quả**

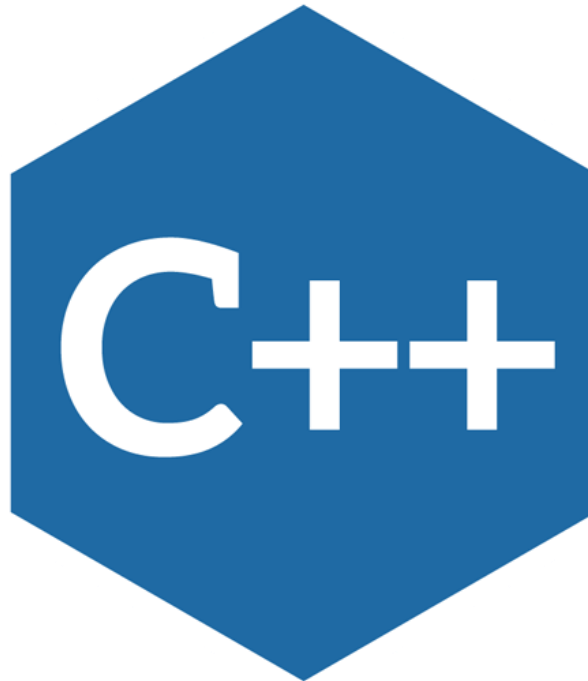
**0 → 1 → 3 → 2 → 5 → 4**  
**Chi phí: -2**

# Đường đi trên đồ thị





# MÃ NGUỒN MINH HỌA BẰNG C++



# Source Code Floyd-Warshall C++

Khai báo thư viện và các biến toàn cục:

```
#include <algorithm>
#include <iostream>
#include <vector>
using namespace std;
#define MAX 105
#define INF 1e9
vector<vector<int>> > graph;
vector<vector<int>> > dist;
vector<vector<int>> > path;
int n;
```

# Source Code Floyd-Warshall C++

In đường đi:

```
void printPath(int s, int t)
{
    int b[MAX];
    int m = 0;
    while (s!=t)
    {
        b[m++] = t;
        t = path[s][t];
    }
    b[m++] = s;
    for (int i = m - 1; i >= 0; i--)
    {
        cout << b[i] << " ";
    }
}
```

# Source Code Floyd-Warshall C++

## Thuật toán chính Floyd-Warshall (part 1)

```
void FloydWarshall(vector<vector<int> > &graph, vector<vector<int> >
&dist)
{
    int i, j, k;
    for (i = 0; i < n; i++)
        for (j = 0; j < n; j++)
        {
            dist[i][j] = graph[i][j];
            if (graph[i][j] != INF && i != j)
            {
                path[i][j] = i;
            }
            else
            {
                path[i][j] = -1;
            }
        }
}
```

# Source Code Floyd-Warshall C++

## Thuật toán chính Floyd-Warshall (part 2)

```
for (k = 0; k < n; k++)
{
    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            if (dist[i][j] > dist[i][k] + dist[k][j])
            {
                dist[i][j] = dist[i][k] + dist[k][j];
                path[i][j] = path[k][j];
            }
        }
    }
}
```

# Source Code Floyd-Warshall C++

## Hàm main (part 1)

```
int main()
{
    freopen("INPUT.INP", "rt", stdin);
    int temp;
    cin >> n;
    graph = vector<vector<int> >(n, std::vector<int>(n));
    dist = vector<vector<int> >(n, std::vector<int>(n));
    path = vector<vector<int> >(n, std::vector<int>(n));
    for (int i = 0; i<n; i++)
        for (int j = 0; j < n; j++)
        {
            cin >> temp;
            if(temp==0 && i!=j)
                graph[i][j] = INF;
            else
                graph[i][j] = temp;
        }
}
```

# Source Code Floyd-Warshall C++

## Hàm main (part 2)

```
FloydWarshall(graph, dist);  
int s = 0;  
int t = 4;  
int result = dist[s][t];  
printPath(s, t);  
cout << result;  
return 0;  
}
```

# MÃ NGUỒN MINH HỌA BẰNG PYTHON





# Source Code Floyd-Warshall python

In đường đi:

```
import queue

MAX = 100
INF = int(1e9)

def printPath(s, t):
    b = []
    while s != t:
        b.append(t)
        t = path[s][t]
    b.append(s)
    for i in range(len(b)-1,-1,-1):
        print(b[i], end=' ' if i > 0 else '\n')
```

# Source Code Floyd-Warshall python

Thuật toán chính Floyd-Warshall.

```
def FloydWarshall(graph, dist):  
    for i in range(n):  
        for j in range(n):  
            dist[i][j] = graph[i][j]  
            if graph[i][j] != INF and i != j:  
                path[i][j] = i  
            else:  
                path[i][j] = -1  
    for k in range(n):  
        for i in range(n):  
            for j in range(n):  
                if dist[i][j] > dist[i][k] + dist[k][j]:  
                    dist[i][j] = dist[i][k] + dist[k][j]  
                    path[i][j] = path[k][j]
```

# Source Code Floyd-Warshall python

Hàm main.

```
if __name__ == '__main__':  
    n = int(input())  
    graph = [[None for i in range(n)] for j in range(n)]  
    dist = [[None for i in range(n)] for j in range(n)]  
    path = [[None for i in range(n)] for j in range(n)]  
    for i in range(n):  
        line = list(map(int, input().split()))  
        for j in range(n):  
            graph[i][j] = INF if line[j] == 0 and i != j else line[j]  
    FloydWarshall(graph, dist)  
    s, t = 0, 4  
    printPath(s, t)  
    print(dist[s][t])
```

# Hỏi đáp

