

Chương 4. ĐƯỜNG ĐI NGẮN NHẤT

Thuật toán Bellman-Ford

ThS. Nguyễn Chí Hiếu

2019

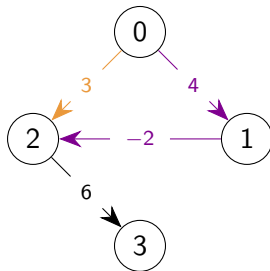
NỘI DUNG

- 1 Đồ thị có trọng số âm
- 2 Thuật toán Bellman-Ford
- 3 Ứng dụng của thuật toán Bellman-Ford

Đồ thị có trọng số âm

Ví dụ 1

Cho đồ thị G gồm 4 đỉnh, tìm đường đi ngắn nhất từ $0 \rightsquigarrow 3$:

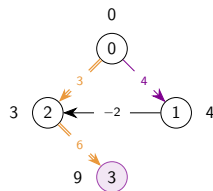
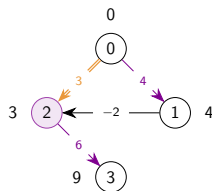
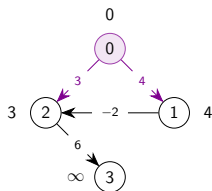
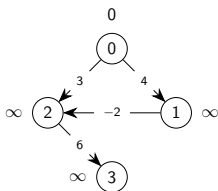


Hình 1: Đồ thị G có trọng số âm.

Đồ thị có trọng số âm

Câu hỏi

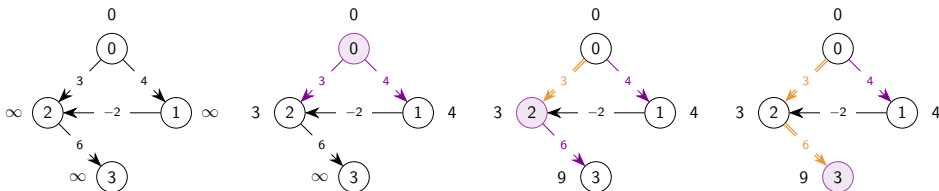
- Áp dụng thuật toán Dijkstra được không?



Đồ thị có trọng số âm

Câu hỏi

- Áp dụng thuật toán Dijkstra được không?



- Đường đi ngắn nhất: $0 \rightarrow 2 \rightarrow 3$ với khoảng cách là 9 (*không chính xác*).

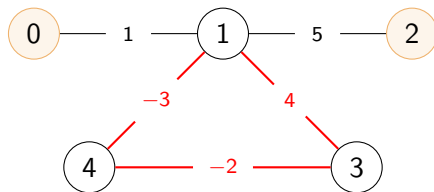
Chú ý

Thuật toán Dijkstra không áp dụng đối với đồ thị có trọng số âm.

Đồ thị có trọng số âm

Ví dụ 2

Cho đồ thị G_2 gồm 5 đỉnh như hình:

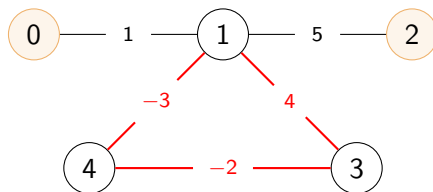


Hình 2: Đồ thị G_2 có chu trình âm.

Chu trình âm (*negative cycle*)

Là chu trình trong đồ thị có **tổng trọng số** các cạnh là một **số âm**.

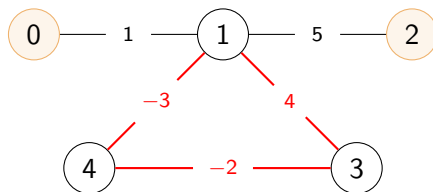
Chu trình âm



Hình 3: Đồ thị G_2 có chu trình âm.

- Đường đi ngắn nhất từ $0 \rightsquigarrow 2$?

Chu trình âm

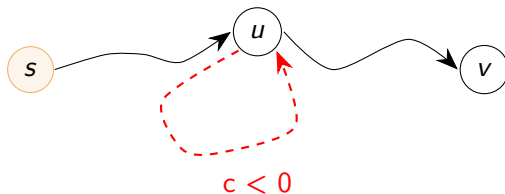


Hình 3: Đồ thị G_2 có chu trình âm.

• Đường đi ngắn nhất từ $0 \rightsquigarrow 2$?

- ☐ $0 \rightarrow 1 \rightarrow 2 : 6$
- ☐ $0 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 1 \rightarrow 2 : 5$
- ☐ $0 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 1 \rightarrow 2 : 4$
- ☐ ...

Chu trình âm

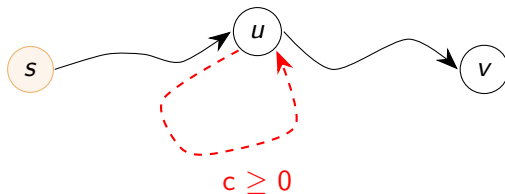


Hình 4: Đường đi từ $s \rightsquigarrow v$.

Nhận xét

- Nếu đường đi từ $s \rightsquigarrow v$ có *chứa chu trình âm*, thì *không tồn tại* đường đi ngắn nhất từ $s \rightsquigarrow v$.

Chu trình âm

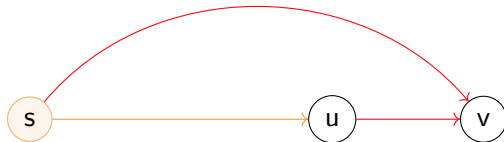


Hình 5: Đường đi từ $s \rightsquigarrow v$.

Nhận xét

- Nếu đường đi từ $s \rightsquigarrow v$ *không chứa chu trình âm*, thì *tồn tại* đường đi ngắn nhất từ $s \rightsquigarrow v$.

Thuật toán Bellman-Ford



Hình 6: Đường đi ngắn nhất từ $s \rightsquigarrow v$.

Thuật toán Bellman-Ford

Nguyên tắc *giảm cạnh không cần thiết* (edge relaxation)

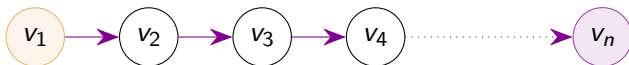
Giả sử, cạnh $(u, v) \in E$ là *cạnh cuối* của đường đi ngắn nhất từ $s \rightsquigarrow u \rightarrow v$.

- $d[u]$ là khoảng cách đường đi ngắn nhất từ $s \rightsquigarrow u$.
- $d[v]$ là khoảng cách đường đi ngắn nhất từ $s \rightsquigarrow v$.
- Nếu cạnh (u, v) giúp đường đi từ $s \rightsquigarrow u \rightarrow v$ ngắn hơn trước, thì cập nhật lại khoảng cách và đường đi tại v .

$$\begin{cases} d[v] &= \min \{d[v], d[u] + w(u, v)\} \\ p[v] &= u \end{cases} \quad (1)$$

với $p[v] = u$ có nghĩa là u chính là đỉnh kề trước của v .

Thuật toán Bellman-Ford



Hình 7: Đường đi ngắn nhất từ $v_1 \rightsquigarrow v_n$ chứa tối đa $|n - 1|$ cạnh.

Ý tưởng

- Dựa vào nguyên tắc *giảm cạnh không cần thiết*, một khoảng cách xấp xỉ dần được thay thế bằng *khoảng cách chính xác* của đường đi ngắn nhất.
- Vì đường đi ngắn nhất giữa hai đỉnh trong đồ thị $G = (V, E)$ có *tối đa $|V| - 1$ cạnh*, nên chỉ cần *lặp tối đa $k = |V| - 1$ lần* để tìm được đường đi ngắn nhất giữa hai đỉnh. Mỗi lần lặp, thực hiện nguyên tắc *giảm cạnh không cần thiết* đối với tất cả các cạnh.
- Chỉ áp dụng đối với đồ thị không chứa chu trình âm.

Thuật toán Bellman-Ford

Thuật toán 1: BellmanFord(G, s)

- Đầu vào: đồ thị G , đỉnh s .
- Đầu ra: đường đi ngắn nhất từ đỉnh s đến các đỉnh khác.

```

1  for each  $u \in V$ 
2       $d[u] \leftarrow \infty$            // distance
3       $p[u] \leftarrow \text{NO\_PATH}$  // previous/parent
4   $d[s] \leftarrow 0$ 
5
6  for  $k \leftarrow 1$  to  $|V| - 1$ 
7      for each edge( $u, v$ )  $\in E$  // Relax( $u, v$ )
8          if  $d[v] > d[u] + w(u, v)$ 
9               $d[v] \leftarrow d[u] + w(u, v)$ 
10              $p[v] \leftarrow u$ 

```

Thuật toán Bellman-Ford

Giải thích

- Dòng 1 \rightarrow 3: khởi tạo khoảng cách và đường đi ngắn nhất đến tất cả đỉnh trong đồ thị.
- Dòng 4: bắt đầu tại đỉnh s , cập nhật khoảng cách của đường đi đến đỉnh này là 0.
- Dòng 6: lặp $k = |V| - 1$ lần, tìm k cạnh trong đường đi ngắn nhất từ đỉnh 0 đến các đỉnh khác *thực hiện nguyên tắc giảm cạnh không cần thiết đối với tất cả các cạnh*.
- *Sau khi thực hiện vòng lặp thứ $k = |V| - 1$, giá trị của $d[v]$ chính là khoảng cách đường đi ngắn nhất từ $s \rightsquigarrow v$.*

Thuật toán Bellman-Ford

Độ phức tạp của thuật toán

```

6   for k ← 1 to |V| - 1
7       for each edge(u, v) ∈ E // Relax(u, v)
8           if d[v] > d[u] + w(u, v)
9               d[v] ← d[u] + w(u, v)
10              p[v] ← u

```

- Dòng 6: lặp $k = |V| - 1$ lần $\Rightarrow \mathcal{O}(|V| - 1)$.
- Dòng 7: lặp $|E|$ lần $\Rightarrow \mathcal{O}(|E|)$.

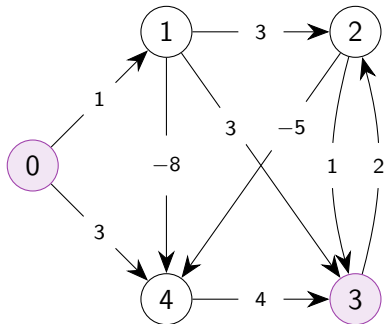
Do đó, độ phức tạp thời gian của thuật toán

$$T \approx \mathcal{O}(|V||E|).$$

Thuật toán Bellman-Ford

Ví dụ 3

Cho đồ thị G gồm 5 đỉnh. Tìm đường đi ngắn nhất từ $0 \rightsquigarrow 3$.



Hình 8: Đồ thị G .

Thuật toán Bellman-Ford

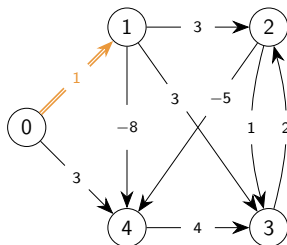
- Bước 1: Khởi tạo khoảng cách và đường đi ngắn nhất từ đỉnh 0 đến tất cả các đỉnh trong đồ thị.

Khởi tạo					
	0	1	2	3	4
d	0	∞	∞	∞	∞
p	-1	-1	-1	-1	-1

- Bước 2: Vì đồ thị có 5 đỉnh, nên lặp $k = 4$ lần để tìm k cạnh trong đường đi ngắn nhất từ đỉnh 0 đến các đỉnh khác.

▶ ...
 ▶ ...
 ▶ ...

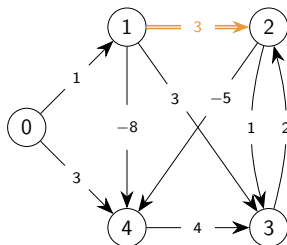
Thuật toán Bellman-Ford



- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$ $(0, 4)$, $(1, 4)$, $(2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	∞	∞	∞
p	-1	0	-1	-1	-1

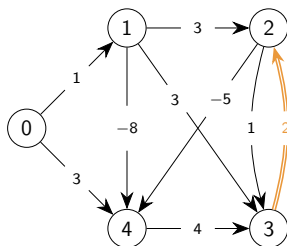
Thuật toán Bellman-Ford



- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	4	∞	∞
p	-1	0	1	-1	-1

Thuật toán Bellman-Ford



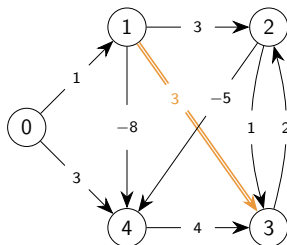
- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh

	0	1	2	3	4
d	0	1	4	∞	∞
p	-1	0	1	-1	-1

- Xét cạnh $(3, 2)$, tìm $d[2] = \min \{d[2], d[3] + w[3, 2]\}$

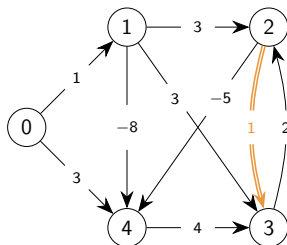
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	4	4	∞
p	-1	0	1	1	-1

Thuật toán Bellman-Ford

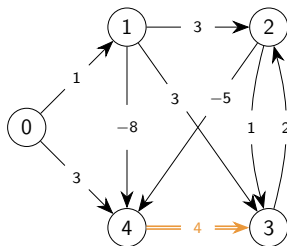


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh

	0	1	2	3	4
d	0	1	4	4	∞
p	-1	0	1	1	-1

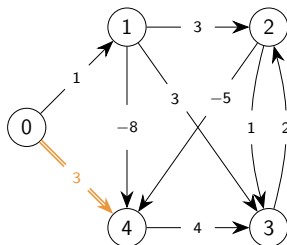
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	4	4	∞
p	-1	0	1	1	-1

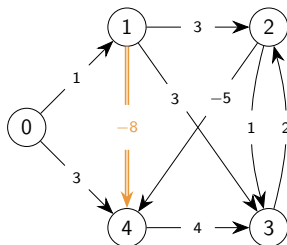
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	4	4	3
p	-1	0	1	1	0

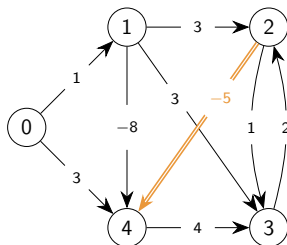
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

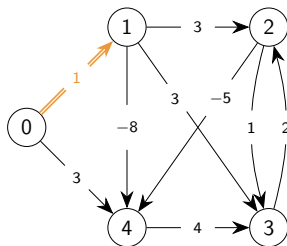
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

1 cạnh					
	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

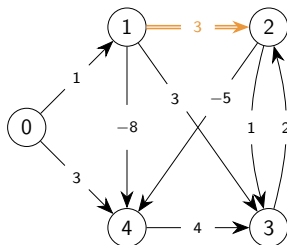
Thuật toán Bellman-Ford



- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

2 cạnh					
	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

Thuật toán Bellman-Ford

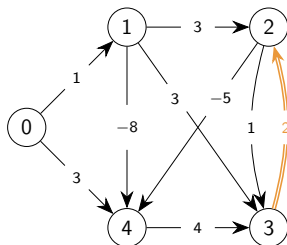


- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

2 cạnh

	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

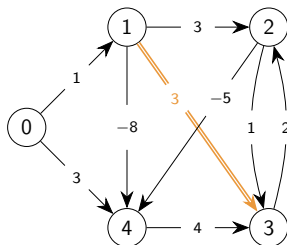
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh					
	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

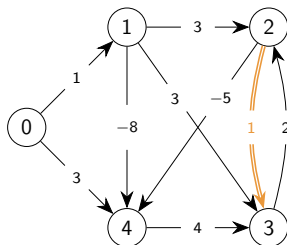
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh					
	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

Thuật toán Bellman-Ford

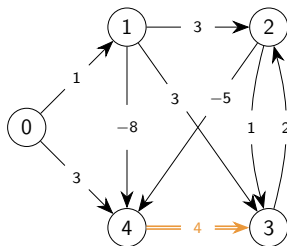


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh

	0	1	2	3	4
d	0	1	4	4	-7
p	-1	0	1	1	1

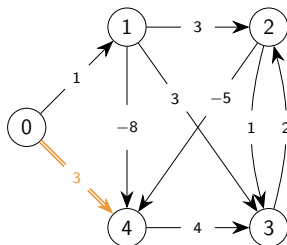
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh					
	0	1	2	3	4
d	0	1	4	-3	-7
p	-1	0	1	4	1

Thuật toán Bellman-Ford

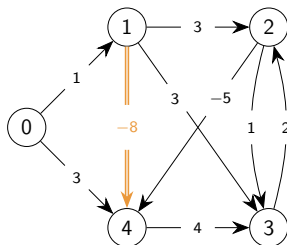


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh

	0	1	2	3	4
d	0	1	4	-3	-7
p	-1	0	1	4	1

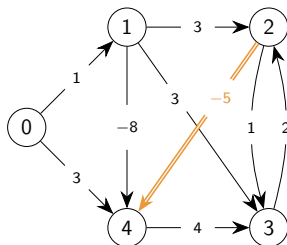
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh					
	0	1	2	3	4
d	0	1	4	-3	-7
p	-1	0	1	4	1

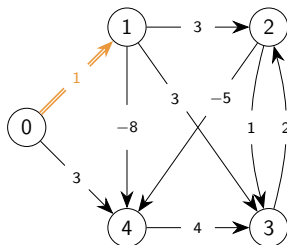
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

2 cạnh					
	0	1	2	3	4
d	0	1	4	-3	-7
p	-1	0	1	4	1

Thuật toán Bellman-Ford

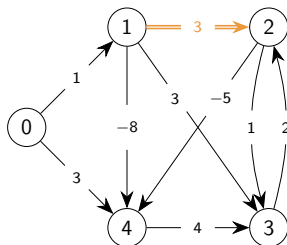


- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

3 cạnh

	0	1	2	3	4
d	0	1	4	-3	-7
p	-1	0	1	4	1

Thuật toán Bellman-Ford

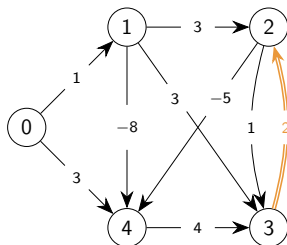


- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

3 cạnh

	0	1	2	3	4
d	0	1	4	-3	-7
p	-1	0	1	4	1

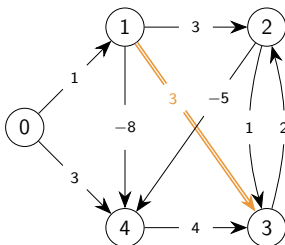
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh					
	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

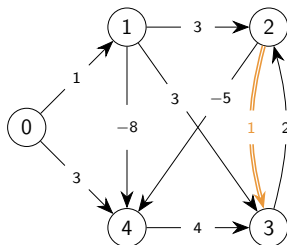


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

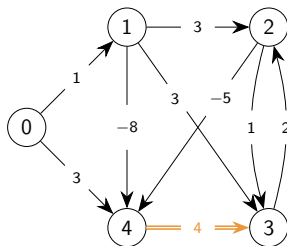


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

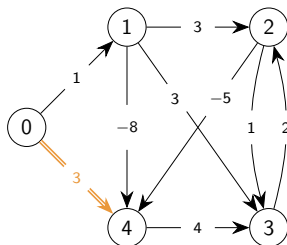


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

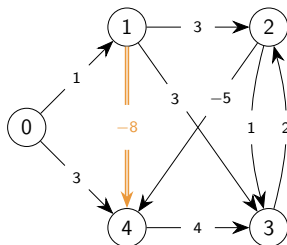


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

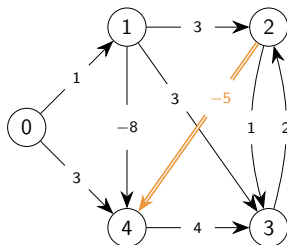
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh					
	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

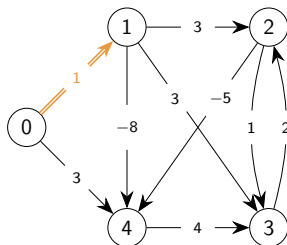
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

3 cạnh					
	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

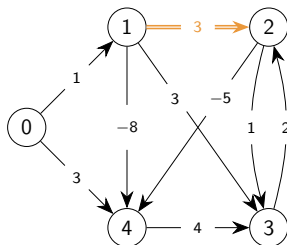


- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

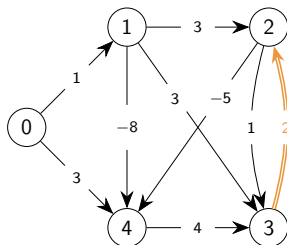


- $(0, 1)$, $(1, 2)$, $(3, 2)$, $(1, 3)$, $(2, 3)$, $(4, 3)$, $(0, 4)$, $(1, 4)$, $(2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

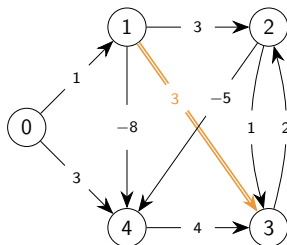
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh					
	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

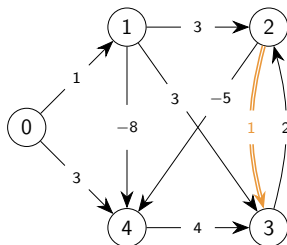


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

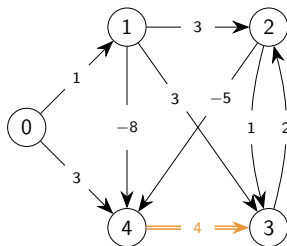


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

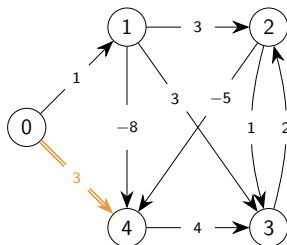


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

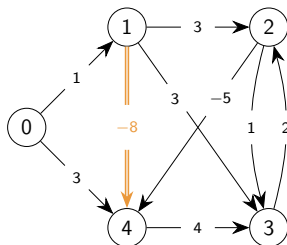


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

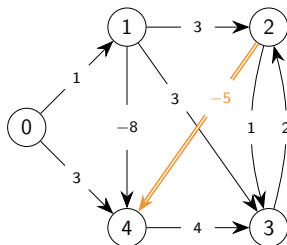


- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh

	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

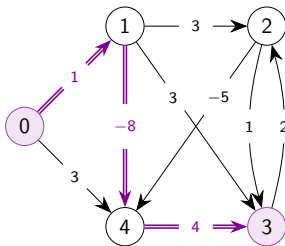
Thuật toán Bellman-Ford



- $(0, 1), (1, 2), (3, 2), (1, 3), (2, 3), (4, 3), (0, 4), (1, 4), (2, 4)$

4 cạnh					
	0	1	2	3	4
d	0	1	-1	-3	-7
p	-1	0	3	4	1

Thuật toán Bellman-Ford

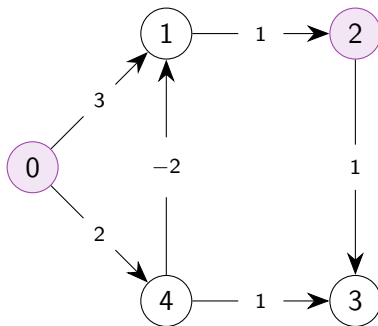


- Đường đi ngắn nhất từ 0 \rightsquigarrow 3: $0 \rightarrow 1 \rightarrow 4 \rightarrow 3$.
- Khoảng cách của đường đi ngắn nhất từ 0 \rightsquigarrow 3: -3 .

Thuật toán Bellman-Ford

Ví dụ 4

Cho đồ thị G gồm 5 đỉnh. Tìm đường đi ngắn nhất từ $0 \rightsquigarrow 2$.



Hình 9: Đồ thị G .

Thuật toán Bellman-Ford

- Áp dụng thuật toán Bellman-Ford tìm chu trình âm của đồ thị.

Thuật toán 2: BellmanFord(G, s)

```

1  for each  $u \in V$ 
2       $d[u] \leftarrow \infty$  // distance
3       $p[u] \leftarrow \text{NO\_PATH}$  // previous/parent
4   $d[s] \leftarrow 0$ 
5
6  for  $k \leftarrow 1$  to  $|V| - 1$ 
7      for each edge  $(u,v) \in E$  // Relax( $u,v$ )
8          if  $d[v] > d[u] + w(u,v)$ 
9               $d[v] \leftarrow d[u] + w(u,v)$ 
10              $p[v] \leftarrow u$ 
11
12  for each edge  $(u,v) \in E$ 
13      if  $d[v] > d[u] + w(u,v)$ 
14          return false // negative cycle
15  return true

```

Thuật toán Bellman-Ford

Giải thích

```

12   for each edge  $(u,v) \in E$ 
13       if  $d[v] > d[u] + w(u,v)$ 
14           return false // negative cycle
15   return true

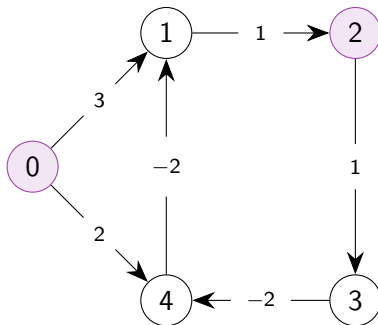
```

- Dòng 12 \rightarrow 15: tìm chu trình âm của đồ thị.
 - ▶ Nếu tìm thấy trả về false (*đồ thị có chu trình âm và không tìm thấy đường đi ngắn nhất*).
 - ▶ Ngược lại, trả về true (*đồ thị không có chu trình âm và tìm thấy đường đi ngắn nhất*).

Thuật toán Bellman-Ford

Ví dụ 5

Cho đồ thị G gồm 5 đỉnh. Tìm đường đi ngắn nhất từ $0 \rightsquigarrow 2$.



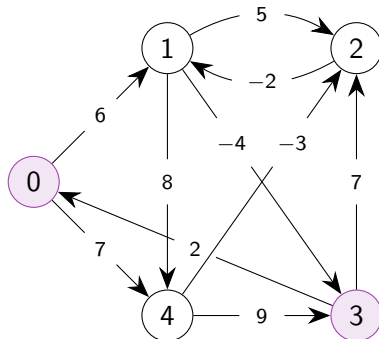
Hình 10: Đồ thị G .

Ứng dụng của thuật toán Bellman-Ford

- Tìm đường đi ngắn nhất giữa 2 đỉnh của đồ thị có trọng số âm.
- Phát hiện chu trình âm trong đồ thị.

Bài tập

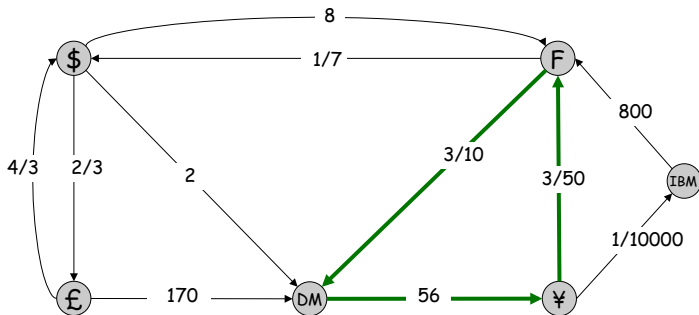
- 1 Cho đồ thị G gồm 5 đỉnh. Tìm đường đi ngắn nhất từ 0 \rightsquigarrow 3.



Hình 11: Đồ thị G .

Bài tập

- 2 Bài toán chuyển đổi ngoại tệ: cho n loại tiền tệ và tỷ giá giữa các loại tiền tệ. Kiểm tra giá trị có bị lệch hay không?



- 3 Trong ví dụ 3, không cần lặp đến $k = |V| - 1$ lần để tìm đường đi ngắn nhất từ s đến các đỉnh khác. Cải tiến thuật toán Bellman-Ford.

Tài liệu tham khảo



ADRIAN BONDY, U.S.R. MURTY, *Graph Theory*, Springer, 2008.



KENNETH H. ROSEN, *Discrete Mathematics and its Applications*, 7th Edition, McGraw-Hill, 2011.



NGUYỄN ĐỨC NGHĨA, NGUYỄN TÔ THÀNH, *Toán rời rạc*, NXB Đại học Quốc gia Hà Nội, 2003.



NGUYỄN CAM, CHU ĐỨC KHÁNH, *Lý thuyết đồ thị*, NXB Đại học Quốc gia Tp Hồ Chí Minh, 2008.



REINHARD DIESTEL, *Graph Theory*, Springer, 2005.