

# Thuật toán đường đi ngắn nhất Dijkstra

Thuật toán

**1)** Tạo một tập hợp *sptSet* (tập hợp cây đường đi ngắn nhất) theo dõi các đỉnh có trong cây đường đi ngắn nhất, tức là khoảng cách tối thiểu từ nguồn được tính toán và hoàn thiện. Ban đầu, bộ này trống.

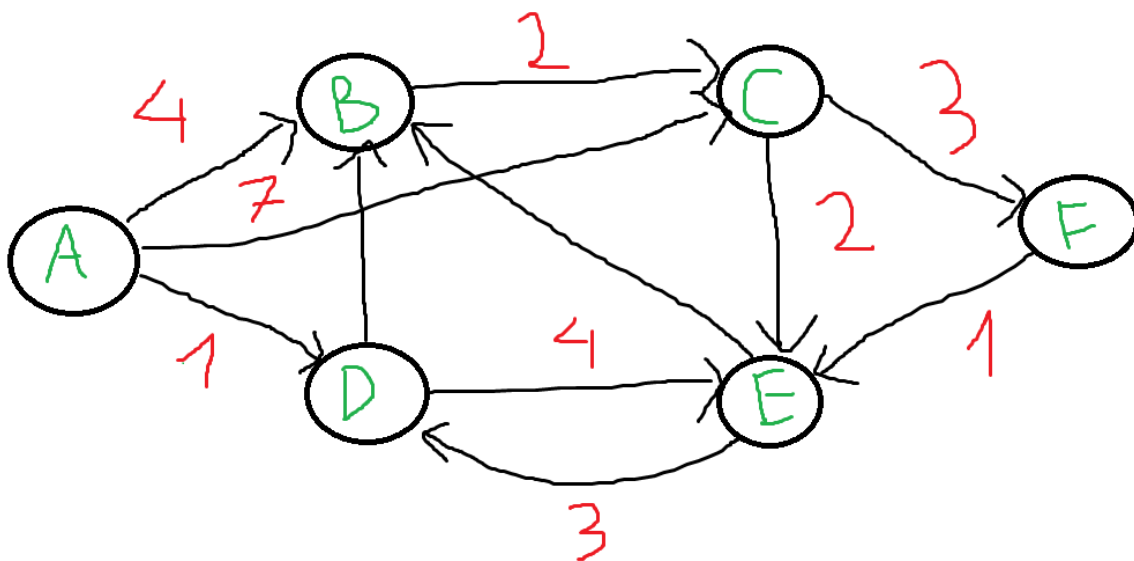
**2)** Gán giá trị khoảng cách cho tất cả các đỉnh trong đồ thị đầu vào. Khởi tạo tất cả các giá trị khoảng cách dưới dạng INFINITE. Gán giá trị khoảng cách là 0 cho đỉnh nguồn để nó được chọn đầu tiên.

**3)** Trong khi *sptSet* không bao gồm tất cả các đỉnh

.... **a)** Chọn một đỉnh *u* không có trong *sptSet* và có giá trị khoảng cách nhỏ nhất.

.... **b)** Đưa *u* vào *sptSet*.

.... **c)** Cập nhật giá trị khoảng cách của tất cả các đỉnh liền kề của *u*. Để cập nhật các giá trị khoảng cách, hãy lặp lại qua tất cả các đỉnh liền kề. Đối với mọi đỉnh liền kề *v*, nếu tổng giá trị khoảng cách của *u* (từ nguồn) và trọng số của cạnh *uv* nhỏ hơn giá trị khoảng cách của *v*, thì cập nhật giá trị khoảng cách của *v*.



Đỉnh	A	B	C	D	E	F
Số bước						
Khởi tạo	0	( $\infty$ , -)	( $\infty$ , -)	( $\infty$ , -)	( $\infty$ , -)	( $\infty$ , -)
1	-	(4, A)	(7, A)	(1, -A)	( $\infty$ , -)	( $\infty$ , -)
2	-	(3, D)	(7, A)	-	(5, D)	( $\infty$ , -)
3	-	-	(5, B)	-	(5, D)	( $\infty$ , -)
4	-	-	-	-	(5, D)	(8, C)
5	-	-	-	-	-	(8, C)
6	-	-	-	-	-	-

-**Note:** những điểm đc đánh dấu màu đỏ sẽ là những điểm mà ta sẽ bắt đầu từ đỉnh đó trong bước tiếp theo(vì nó có trọng số là min)

-Ta có đường đi ngắn nhất từ đỉnh A tới các đỉnh là:

+A-A:0

+A-D-B:3

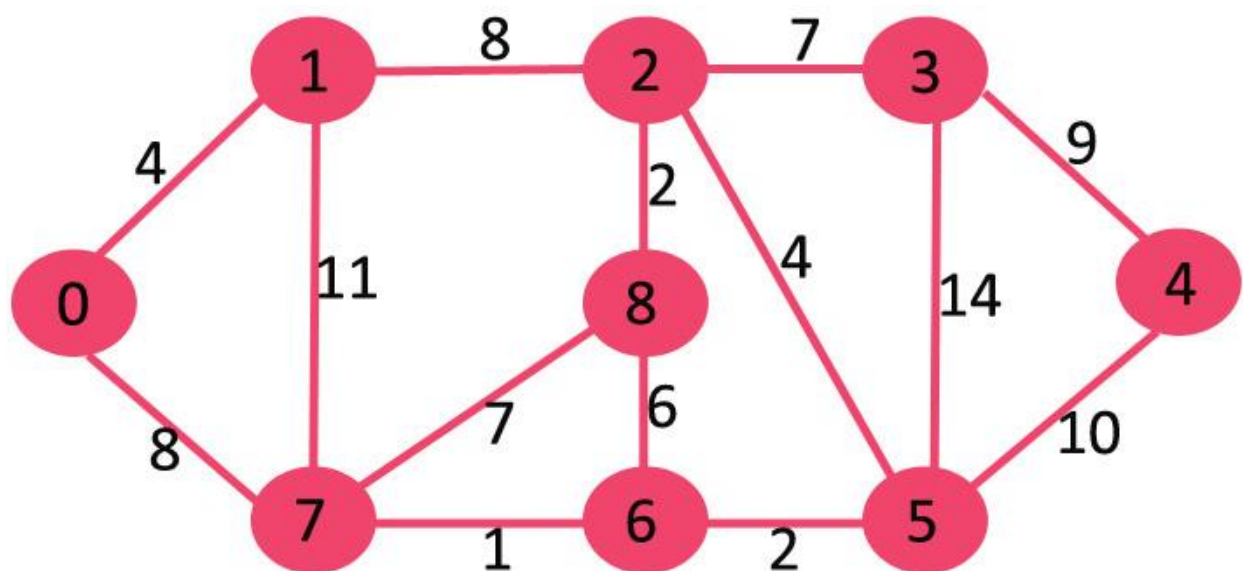
+A-D-B-C:5

+A-D=1

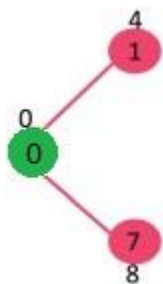
+A-D-E:5

+A-D-B-C-F:8

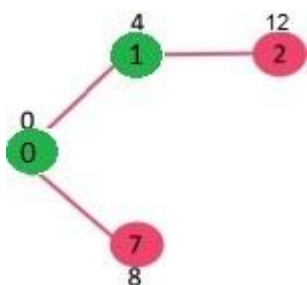
Ví dụ sau(1 ví dụ bằng hình ảnh e lấy trên mạng để thêm phần khác quan):



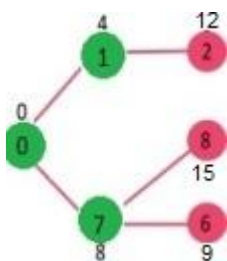
Tập *sptSet* ban đầu trống và khoảng cách được gán cho các đỉnh là {0, INF, INF, INF, INF, INF, INF, INF, INF} trong đó INF chỉ ra vô hạn. Bây giờ chọn đỉnh có giá trị khoảng cách tối thiểu. Đỉnh 0 được chọn, đưa nó vào *sptSet*. Vì vậy, *sptSet* trở thành {0}. Sau khi bao gồm 0 vào *sptSet*, hãy cập nhật các giá trị khoảng cách của các đỉnh liền kề của nó. Các đỉnh liền kề của 0 là 1 và 7. Giá trị khoảng cách của 1 và 7 được cập nhật là 4 và 8. Đồ thị con sau đây hiển thị các đỉnh và giá trị khoảng cách của chúng, chỉ các đỉnh có giá trị khoảng cách hữu hạn mới được hiển thị. Các đỉnh có trong SPT được hiển thị bằng màu xanh lá cây.



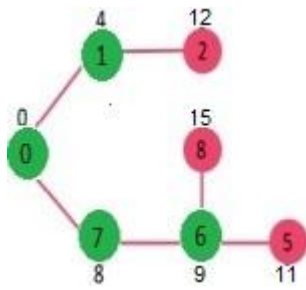
Chọn đỉnh có giá trị khoảng cách nhỏ nhất và chưa có trong SPT (không có trong sptSET). Đỉnh 1 được chọn và thêm vào sptSet. Vì vậy, sptSet bây giờ trở thành {0, 1}. Cập nhật giá trị khoảng cách của các đỉnh liên kề là 1. Giá trị khoảng cách của đỉnh 2 trở thành 12.



Chọn đỉnh có giá trị khoảng cách nhỏ nhất và chưa có trong SPT (không có trong sptSET). Đỉnh 7 được chọn. Vì vậy, sptSet bây giờ trở thành {0, 1, 7}. Cập nhật giá trị khoảng cách của các đỉnh liên kề là 7. Giá trị khoảng cách của đỉnh 6 và 8 trở thành hữu hạn (tương ứng là 15 và 9).



Chọn đỉnh có giá trị khoảng cách nhỏ nhất và chưa có trong SPT (không có trong sptSET). Đỉnh 6 được chọn. Vì vậy, sptSet bây giờ trở thành {0, 1, 7, 6}. Cập nhật giá trị khoảng cách của các đỉnh liên kề là 6. Giá trị khoảng cách của đỉnh 5 và 8 được cập nhật.



Chúng ta lặp lại các bước trên cho đến khi *sptSet* bao gồm tất cả các đỉnh của đồ thị đã cho. Cuối cùng, chúng ta nhận được Cây đường dẫn ngắn nhất (SPT) sau đây.

