

HƯỚNG DẪN THUẬT TOÁN KRUSKAL TÌM CÂY KHUNG NHỎ NHẤT

Phần hướng dẫn trong bài này xem như sinh viên đã nắm được nội dung thuật toán và kỹ thuật tìm cây khung/cây bao trùm đã trình bày trong bài tập thực hành trước: “**Hướng dẫn thuật toán Prime tìm cây khung nhỏ nhất**”. Nếu bạn chưa nắm được hay quên thuật toán Prim thì bạn mở file “**Hướng dẫn thuật toán Prime tìm cây khung nhỏ nhất**” và đọc lại nhé **J**.

Nhắc lại

Trong Prime, chúng ta thực hiện việc mở rộng tập đã xét (ban đầu chỉ gồm một đỉnh, 0 cạnh thành n đỉnh, $n-1$ cạnh) dựa trên các cạnh ngắn nhất nối giữa tập đã xét và tập chưa xét. Tư tưởng chính là thêm vào các cạnh ngắn nhất sao cho không tạo ra chu trình. Như vậy, có trường hợp một cạnh sẽ phải xét đi xét nhiều lần rồi mới được chọn, thậm chí không hề được chọn.

Đối với Kruskal, cũng thêm lần lượt các cạnh vào đồ thị, theo thứ tự từ trọng nhỏ nhất đến trọng lớn nhất (như vậy mỗi cạnh sẽ chỉ được duyệt một lần duy nhất). Ta chỉ bổ sung cạnh vào cây khung nếu việc thêm cạnh này không làm phát sinh ra chu trình.

1. Thuật toán Kruskal

Cho $G=(X,E)$ là một đồ thị có trọng số gồm n đỉnh. Thuật toán Kruskal được dùng để tìm ra cây khung/cây bao trùm ngắn nhất của G như sau:

Bước 1: Duyệt các cạnh của đồ thị G và tạo danh sách các cạnh listEdge.

Hướng dẫn thuật toán Kruskal

Bước 2: Sắp xếp lại danh sách các cạnh listEdge của đồ thị G theo trọng số tăng dần, rồi đánh dấu tất cả các cạnh là chưa xét và khởi tạo $T := \emptyset$. (T là tập cạnh của cây khung hay cây bao trùm của đồ thị G).

Bước 3: Lấy cạnh e chưa xét có trọng số bé nhất trong danh sách listEdge đã sắp xếp, nếu $T \cup \{e\}$ không chứa chu trình thì gán $T := T \cup \{e\}$.

Đánh dấu cạnh e đã xét.

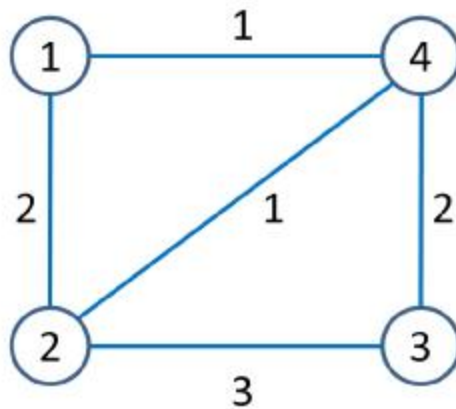
Bước 4: Nếu hết cạnh chưa xét (tức các cạnh đã xét hết) hoặc T có đủ n-1 cạnh thì dừng, ngược lại làm tiếp tục bước 3.

Thuật toán dừng, nếu T không có đủ n-1 cạnh thì đồ thị không liên thông và không có cây khung/bao trùm nhỏ nhất. Ngược lại thì có cây khung/bao trùm.

Chú ý: Trong các thuật toán tìm khung ngắn nhất chúng ta có thể bỏ đi hướng các cạnh và các khuyên; đối với cạnh song song thì có thể bỏ đi và chỉ để lại một cạnh trọng lượng nhỏ nhất trong chúng.

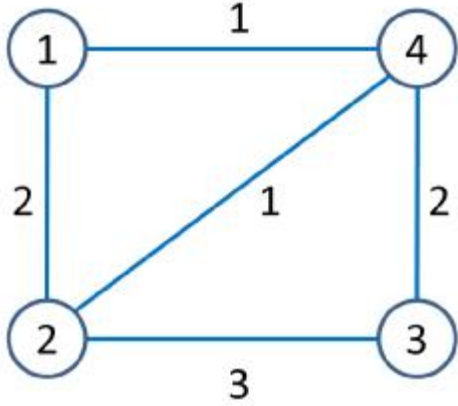
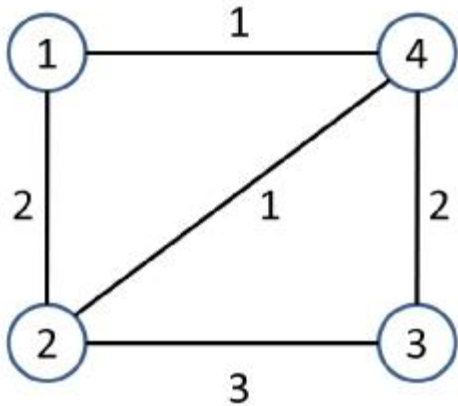
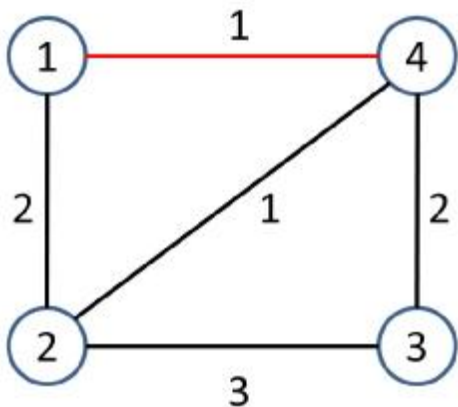
2. Ví dụ thi hành thuật toán Kruskal

Cho đồ thị sau:

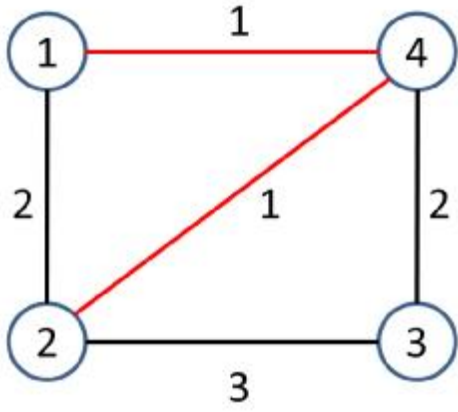
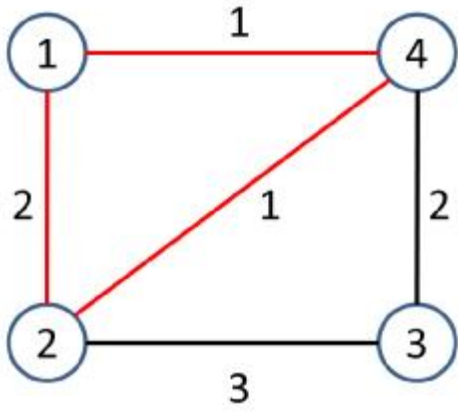
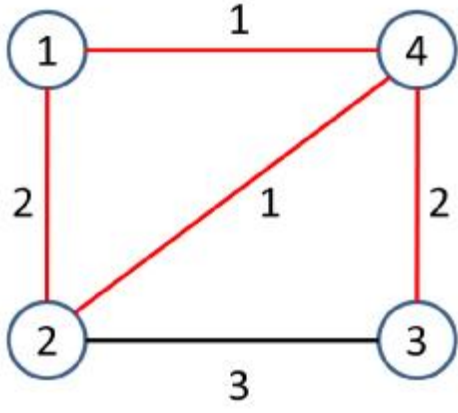


Tìm cây khung ngắn nhất của đồ thị trên dùng thuật toán Kruskal.

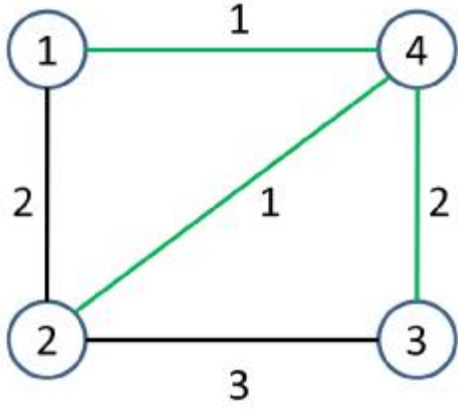
Hướng dẫn thuật toán Kruskal

	<p>Bước 1: Duyệt các cạnh của đồ thị G và tạo danh sách các cạnh $listEdge$.</p> <p>Đồ thị G có 5 cạnh $\Rightarrow listEdge$ có 5 cạnh.</p> <p>$listEdge = \{(1,2); (1,4); (2,3); (2,4); (3,4)\}$</p>
	<p>Bước 2: Sắp xếp lại danh sách các cạnh $listEdge$ của đồ thị G theo trọng số tăng dần, rồi đánh dấu tất cả các cạnh là chưa xét và khởi tạo $T := \emptyset$.</p> <p>Sắp xếp lại $listEdge$ theo trọng số tăng dần $\Rightarrow listEdge = \{(1,4); (2,4); (1,2); (3,4); (2,3)\}$</p> <p>Đánh dấu các cạnh chưa xét là màu đen.</p> <p>$T := \emptyset$.</p>
	<p>Bước 3: Lấy cạnh e chưa xét có trọng số bé nhất trong danh sách $listEdge$ đã sắp xếp, nếu $T \cup \{e\}$ không chứa chu trình thì gán $T := T \cup \{e\}$. Đánh dấu cạnh e đã xét.</p> <p>Lấy cạnh e chưa xét có trọng số bé nhất trong danh sách $listEdge \Rightarrow$ lấy cạnh $(1,4)$.</p> <p>$T \cup \{(1,4)\}$ không tạo chu trình $\Rightarrow T := T \cup \{(1,4)\} = \{(1,4)\}$.</p> <p>Đánh dấu cạnh $(1,4)$ đã xét bằng màu đỏ.</p>

Hướng dẫn thuật toán Kruskal

	<p>Bước 4: Nếu hết cạnh chưa xét (tức các cạnh đã xét hết) hoặc T có đủ $n-1$ cạnh thì dừng, ngược lại làm tiếp tục bước 3.</p> <p>T chỉ mới có 1 cạnh $< n-1 = 4 - 1 = 3$ \Rightarrow quay lại bước 3.</p> <p>Bước 3 (lần 2): Lấy cạnh e chưa xét có trọng số bé nhất trong danh sách listEdge \Rightarrow lấy cạnh (2,4).</p> <p>$T \cup \{(2,4)\}$ không tạo chu trình $\Rightarrow T = T \cup \{(2,4)\} = \{(1,2); (2,4)\}$.</p> <p>Đánh dấu cạnh (2,4) đã xét bằng màu đỏ.</p>
	<p>Bước 4 (lần 2): T chỉ mới có 2 cạnh $< n-1 = 4 - 1 = 3$ \Rightarrow quay lại bước 3.</p> <p>Bước 3 (lần 3): Lấy cạnh e chưa xét có trọng số bé nhất trong danh sách listEdge \Rightarrow lấy cạnh (1,2).</p> <p>$T \cup \{(1,2)\}$ tạo chu trình 1 \rightarrow 2 \rightarrow 4 \rightarrow 1 \Rightarrow Không thể lấy cạnh (1,2). $\Rightarrow T = \{(1,2); (2,4)\}$.</p> <p>Đánh dấu cạnh (1,2) đã xét bằng màu đỏ.</p>
	<p>Bước 4 (lần 3): T chỉ mới có 2 cạnh $< n-1 = 4 - 1 = 3$ \Rightarrow quay lại bước 3.</p> <p>Bước 3 (lần 4): Lấy cạnh e chưa xét có trọng số bé nhất trong danh sách listEdge \Rightarrow lấy cạnh (3,4).</p> <p>$T \cup \{(3,4)\}$ không tạo chu trình $\Rightarrow T = T \cup \{(3,4)\} = \{(1,2); (2,4); (3,4)\}$.</p> <p>Đánh dấu cạnh (3,4) đã xét bằng màu đỏ</p>

Hướng dẫn thuật toán Kruskal

	<p>Bước 4 (lần 3): T chỉ mới có 3 cạnh = $n-1 = 4 - 1 = 3$ ẽ dừng.</p> <p>Vậy cây khung/bao trùm là màu xanh lá cây tương ứng với những cặp cạnh trong $T = \{(1,2); (2,4); (3,4)\}$</p> <p>Khi đó tổng trọng số của cây khung/cây bao trùm là $1+1+2 = 4$.</p>
-----------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3. So sánh khác biệt Kruskal với Prim

Khác với Prime, Kruskal không cần kiểm tra đồ thị liên thông trước khi thi hành thuật toán. Nếu quá trình thi hành thuật toán tìm được cây khung/bao trùm thì đồ thị liên thông và ngược lại là đồ thị không liên thông.

Khác với Prime là mở rộng tập đỉnh đã xét, Kruskal kiểm tra nếu khi thêm cạnh đang xét vào cây khung mà **không làm phát sinh chu trình** thì sẽ chọn cạnh này. Việc kiểm tra chu trình như thế này sẽ tốn chi phí. Ta có thể áp dụng cách sau để làm giảm chi phí kiểm tra chu trình:

1. Dùng một mảng Nhãn có kích thước bằng số đỉnh của đồ thị. Giá trị Nhãn[i] tại đỉnh i được khởi tạo bằng với chính chỉ số i của đỉnh.

Ví dụ: Với đồ thị G ở trên ta có

Đỉnh	1	2	3	4
Nhãn	1	2	3	4

2. Khi chọn một cạnh (u,v) được thêm vào cây khung/cây bao trùm, ta sửa **nhãn của tất cả các đỉnh** có cùng giá trị với **nhãn của đỉnh v** thành **nhãn của đỉnh u**.

Hướng dẫn thuật toán Kruskal

Ví dụ: sau khi thêm cạnh (1, 4) vào cây khung/cây bao trùm hay tập T, ta sửa nhãn của tất cả các đỉnh có cùng giá trị với nhãn của đỉnh 4 (là 4) thành nhãn của đỉnh 1 (là 1).

Đỉnh	1	2	3	4
Nhãn	1	2	3	<u>1</u>

Sau đó, nếu như ta chọn tiếp cạnh (2, 4), ta sửa nhãn của tất cả các đỉnh có cùng giá trị với nhãn của đỉnh 4 (là 1) thành nhãn của đỉnh 2 (là 2).

Đỉnh	1	2	3	4
Nhãn	<u>2</u>	2	3	<u>2</u>

- Sau đó, khi chọn cạnh để thêm vào cây khung/cây bao trùm thì chỉ chọn cạnh có nhãn hai đỉnh là khác nhau \Rightarrow không tạo thành chu trình.

Ví dụ: Khi ta xét tới cạnh (1,2) thì ta không chọn vì đỉnh 1 và đỉnh 2 có cùng nhãn là 2, do đó nếu ta chọn cạnh (1,2) sẽ tạo ra chu trình \Rightarrow không chọn cạnh (1,2) đưa vào cây khung/cây bao trùm.

Nói thêm: Sau khi ta đã chọn đủ $n-1$ cạnh, ta sẽ có mảng nhãn chỉ chứa một giá trị duy nhất.

-----**HẾT**-----