

Tìm cây khung nhỏ nhất với Kruskal

1. Thuật toán Kruskal

Cho $G=(X,E)$ là một đồ thị liên thông có trọng gồm n đỉnh. Thuật toán Kruskal được dùng để tìm ra cây khung ngắn nhất của G .

Bước 1: Sắp xếp các cạnh theo thứ tự trọng lượng tăng dần và khởi tạo $T := \emptyset$.

Bước 2: Lấy cạnh e ở đầu danh sách đã sắp xếp (có trọng nhỏ nhất). Nếu $T + \{e\}$ không chứa chu trình thì gán $T := T + \{e\}$. Loại cạnh e khỏi danh sách.

Bước 3: Nếu T đủ $n-1$ phần tử thì dừng, ngược lại làm tiếp tục bước 2.

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.

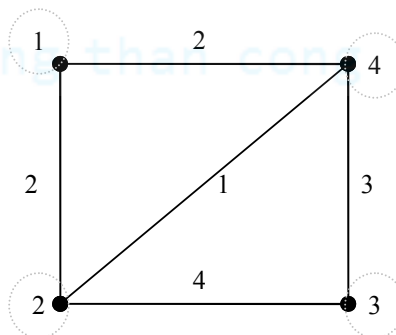
So sánh với Prim: Trong Prim, chúng ta thực hiện việc mở rộng tập đã xét (ban đầu chỉ gồm một đỉnh, 0 cạnh thành n đỉnh, $n-1$ cạnh) dựa trên các cạnh ngắn nhất nối giữa tập đã xét và tập chưa xét. Như vậy, có trường hợp một cạnh sẽ phải xét đi xét nhiều lần rồi mới được chọn, thậm chí không hề được 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).

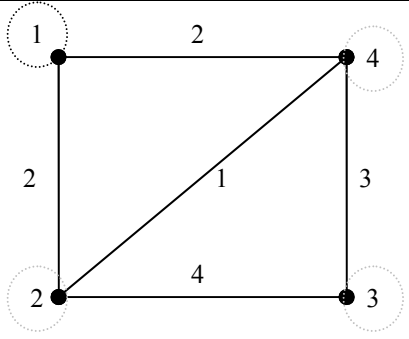
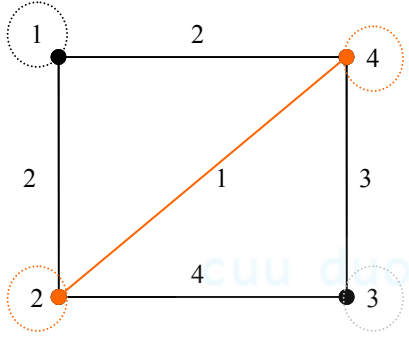
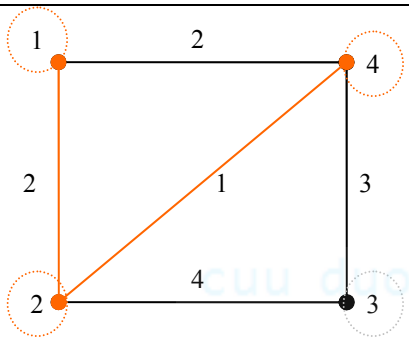
Ta chỉ bổ sung cạnh vào cây khung nếu việc thêm cạnh này không làm phát sinh ra chu trình.

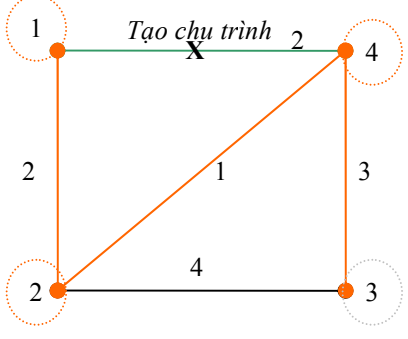
2. Ví dụ Kruskal

Cho đồ thị sau:



Tìm cây khung ngắn nhất của đồ thị.

	<p>Bước 1: Sắp xếp các cạnh theo thứ tự trọng lượng tăng dần và khởi tạo $T := \emptyset$.</p> <p>Gọi lstEdge là tập các cạnh đã sắp thứ tự trọng tăng:</p> $\text{lstEdge} = \{(2,4); (1,2); (1,4); (3,4); (2,3)\}$ <p>$T = \emptyset$.</p>
	<p>Bước 2: Lấy cạnh e ở đầu danh sách đã sắp xếp (có trọng nhỏ nhất). Nếu $T + \{e\}$ không chứa chu trình thì gán $T := T + \{e\}$. Loại cạnh e khỏi danh sách.</p> <p>Vì cạnh $(2,4)$ bổ sung vào T ($T = \emptyset$) không tạo thành chu trình nên: $T = \{(2,4)\}$.</p> <p>Danh sách cạnh:</p> $\text{lstEdge} = \{(1,2); (1,4); (3,4); (2,3)\}$
	<p>Bước 3: Nếu T đủ $n-1$ phần tử thì dừng, ngược lại làm tiếp tục bước 2.</p> <p>T chỉ có 1 phần tử $< n - 1 = 4 - 1 = 3$ nên thuật toán chưa dừng.</p> <p>Bước 2 (lần 2):</p> <p>Cạnh $(1,2)$ không tạo chu trình khi thêm vào T, vậy: $T = \{(2,4); (1,2)\}$</p> $\text{lstEdge} = \{(1,4); (3,4); (2,3)\}$ <p>Bước 3 (lần 2): T chưa đủ nên tiếp tục.</p>

	<p><u>Bước 2 (lần 3):</u></p> <p>Cạnh (1,4) khi thêm vào T tạo thành chu trình nên không xét cạnh này. Cạnh tiếp theo (3,4) thỏa mãn: $T = \{(2,4); (1,2); (3,4)\}$</p> <p style="text-align: center;">$lstEdge = \{(2,3)\}$</p> <p><u>Bước 3 (lần 3):</u> T đủ nên dừng thuật toán</p>
---	---

Hàm tìm cây khung là một phương thức của lớp Graph. Graph KrushAlg()

```
{
    Graph sT // cây khung cần tìm
    // Nếu đồ thị hiện tại biểu diễn dưới dạng ma trận kề thì chuyển sang dạng danh sách cạnh.
    // Sắp xếp các cạnh theo thứ tự tăng dần.
    while (số cạnh của cây khung sT chưa đủ)
    {
        // chọn cạnh e bé nhất chưa xét
        if (eMinIndex < nEdgeCount )
        {
            //Kiểm tra xem cạnh này có tạo thành chu trình khi thêm vào không
            if (IsCircle(eMinIndex) == false)
            {
                // thêm cạnh có chỉ mục eMinIndex vào cây khung
                sT.AddEdge()
            }
            eMinIndex++;
        }
        else // Số cạnh của cây khung chưa đủ nhưng đã xét hết các cạnh của đồ thị
        {
            //Dừng thuật toán
        }
    }
}
```

Bài tập thực hành

Cây khung nhỏ nhất với Kruskal

I. Quy định

Thời gian làm bài: **1 tuần** (deadline xem trên moodle)

Loại bài tập: **cá nhân**

Phong cách lập trình: **hướng đối tượng**

Cấu trúc bên trong thư mục **MSSV** bao gồm các thư mục

- **Source**: thư mục chứa toàn bộ source code.

- **Release**: thư mục chứa file thực thi (**MSSV.exe**).

Nén toàn bộ thư mục thành file **MSSV.zip** hoặc **.rar**

Nộp bài lên moodle.

Lưu ý: tất cả các bài làm sai qui định sẽ không được chấm tức là 0 điểm.

II. Đề bài

Sử dụng *thuật toán Kruskal* để tìm *cây khung nhỏ nhất* trong đồ thị vô hướng có trọng số. Chương trình được viết dưới dạng Console với 2 tham số dòng lệnh lần lượt là đường dẫn *tập tin đầu vào* và đường dẫn *tập tin đầu ra*:

Ví dụ tham số dòng lệnh:

MSSV.exe input.txt Output.txt

Hoặc

MSSV.exe C:\input.txt D:\Output.txt

Lưu ý: đường dẫn/tên tập tin đầu vào và đầu ra có thể thay đổi (không cố định)

Cấu trúc file dữ liệu đầu vào:

- Dòng đầu tiên: **số đỉnh đồ thị (N)**
- N dòng tiếp theo: **ma trận kề** của đồ thị với quy ước:
 - ✓ $A[i][j] = W$: trọng số của đường nối trực tiếp từ i đến j
 - ✓ $A[i][j] = 0$: không có đường nối trực tiếp từ i đến j

Các đỉnh được đánh chỉ số từ 0

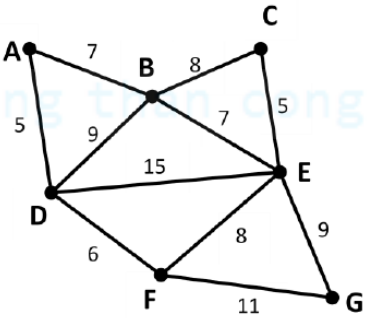
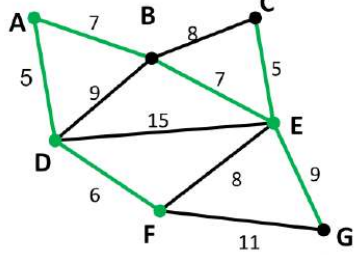
Cấu trúc file dữ liệu đầu ra:

- Nếu tìm được cây khung nhỏ nhất
 - Dòng đầu chứa số nguyên k là *trọng lượng* cây khung cực tiểu
 - Dòng tiếp theo là các *cạnh* (u, v) thuộc cây khung này,
- Nếu không tìm được ghi là “NULL” (**KHÔNG** dấu “”)

Lưu ý:

- Cách xuất các cạnh theo qui ước (u,v) và các cạnh cách nhau bởi dấu “;”
- Thứ tự trong một cạnh: đỉnh u nhỏ hơn đỉnh v
- Thứ tự giữa các cạnh: sắp theo thứ tự từ **nhỏ đến lớn** theo **đỉnh u** , nếu **đỉnh u** bằng nhau thì xét đến **đỉnh v** (cũng theo thứ tự từ **nhỏ đến lớn**)

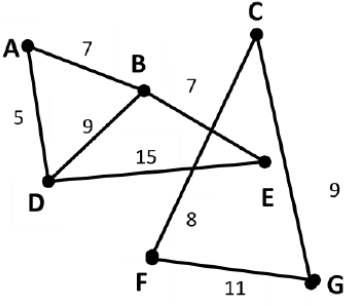
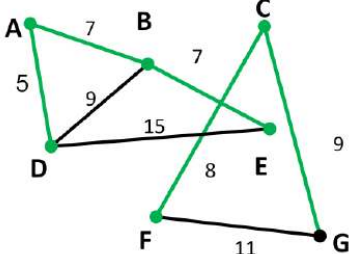
Ví dụ:

Tập tin đầu vào	Đồ thị	Tập tin đầu ra
<pre> 7 0 7 0 5 0 0 0 7 0 8 9 7 0 0 0 8 0 0 5 0 0 5 9 0 0 15 6 0 0 7 5 15 0 8 9 0 0 0 6 8 0 11 0 0 0 0 9 11 0 </pre>	 <p>Với $A \equiv 0, B \equiv 1, C \equiv 2, D \equiv 3, E \equiv 4, F \equiv 5, G \equiv 6$, A là gốc</p>	<pre> 39 (0,1); (0,3); (1,4);(2,4);(3,5); (4,6) </pre>  <p>Đường màu xanh biểu diễn cây khung cực tiểu</p>

IV. Mở rộng (cộng điểm)

Đối với trường hợp đồ thị không liên thông (không tìm thấy cây khung cho đồ thị này) thì xuất ra tất cả cây khung của mỗi thành phần liên thông của đồ thị. Khi đó, thay vì xuất ra “NULL” thì xuất ra từng cây khung nhỏ nhất như qui định ở trên (theo thứ tự *trọng lượng* tăng dần)

Ví dụ:

Tập tin đầu vào	Đồ thị	Tập tin đầu ra
<pre> 7 0 7 0 5 0 0 0 7 0 0 9 7 0 0 0 0 0 0 0 8 9 5 9 0 0 15 0 0 0 7 0 15 0 0 0 0 0 8 0 0 0 11 0 0 9 0 0 11 0 </pre>	 <p>Với $A \equiv 0, B \equiv 1, C \equiv 2,$ $D \equiv 3, E \equiv 4, F \equiv 5, G \equiv 6,$ A là gốc</p>	<pre> 17 (2,5);(2,6) 19 (0,1);(0,3);(1,4) </pre>  <p>Đường màu xanh biểu diễn cây khung cực tiểu</p>

cuu duong than cong . com

cuu duong than cong . com