



10397 Connect the Campus

Link submit:

https://uva.onlinejudge.org/index.php?option=onlinejudge&page=show_problem&problem=1338

Solution:

C++: <http://ideone.com/BZ8q5n>

Python: <https://ideone.com/os4go2>

Java: <https://ideone.com/6GMigh>

Tóm tắt đề: Cho N tòa nhà nằm ở các tọa độ (x, y) trên bản đồ, mỗi tòa nhà ở một tọa độ khác nhau, không có 2 tòa nhà cùng một tọa độ. Các tòa nhà được đánh số từ $1 \rightarrow N$. Cho danh sách M cặp tòa nhà đã có kết nối dây cáp với nhau. Nhiệm vụ của bạn là cần bao nhiêu đoạn dây cáp nữa để tất cả các tòa nhà đều có kết nối với nhau.

Input

Có nhiều bộ test cần xử lý cùng một lúc, mỗi bộ test chứa các thông tin sau:

Dòng đầu tiên chứa số lượng tòa nhà N ($1 \leq N \leq 750$).

N dòng tiếp theo mỗi dòng chứa tọa độ (x, y) là tọa độ của từng ngôi nhà giá trị không vượt quá 10000.

Dòng tiếp theo chứa số M ($0 \leq M \leq 1000$) là số lượng các cặp tòa nhà đã được kết nối sẵn dây cáp.

M dòng tiếp theo là danh sách các cặp tòa nhà.

Output

Tìm đường cáp ngắn nhất để tất cả các tòa nhà được kết nối với nhau. Số lẻ thì lấy phía sau dấu chấm 2 chữ số.

4	4.41
103 104	4.41
104 100	
104 103	
100 100	

1	
4 2	
4	
103 104	
104 100	
104 103	
100 100	
1	
4 2	

Giải thích: Trong ví dụ đầu tiên có 4 tòa nhà lần lượt nằm tại các tọa độ:

- Tòa nhà 1 (103, 104)
- Tòa nhà 2 (104, 100)
- Tòa nhà 3 (104, 103)
- Tòa nhà 4 (100, 100)

Chỉ có 2 tòa nhà có kết nối là tòa nhà 4 và tòa nhà 2, các tòa nhà còn lại chưa có kết nối. Vì thế bạn phải tìm đường kết nối sao cho kết quả là nhỏ nhất.

Hướng dẫn giải:

Bạn sẽ tính khoảng cách của tất cả các cặp tòa nhà, trừ những tòa nhà nào đã có kết nối rồi thì bạn đặt chi phí kết nối là 0.

Sau khi đã tính xong thì bạn bỏ toàn bộ các cặp kết nối và chi phí đó vào graph rồi chạy thuật toán Prim.

Cây khung nhỏ nhất là kết quả tìm được.

Độ phức tạp: $O(T * M \log N)$ với T là số lượng bộ dataset cho mỗi test và $O(E \log V)$ là độ phức tạp của thuật toán Prim.