



# CSTREET - Cobbled streets

Link submit: <http://www.spoj.com/problems/CSTREET/>

Solution:

C++	<a href="https://ideone.com/hUneK8">https://ideone.com/hUneK8</a>
Java	<a href="https://ideone.com/q37Wfw">https://ideone.com/q37Wfw</a>
Python	<a href="https://ideone.com/b9P0ht">https://ideone.com/b9P0ht</a>

Tóm tắt đề:

Cho bạn đường đi đến các tòa nhà và chi phí của nó, hãy tìm các đoạn đường đi sao cho đi được hết tất cả các thành phố với chi phí nhỏ nhất rồi nhân thêm với chi phí p.

Input:

Số đầu tiên là số testcase  $1 \leq t \leq 100$ . Mỗi testcase có các thông tin sau:

- Dòng đầu chứa một số nguyên p là chi phí để tính sau khi có được cây khung nhỏ nhất.
- Dòng thứ hai gồm một số nguyên n ( $1 \leq n \leq 1000$ ) là số lượng tòa nhà.
- Dòng thứ ba chứa một số nguyên m ( $1 \leq m \leq 300000$ ) cho biết số lượng đường đi đến các tòa nhà.
- m dòng tiếp theo, mỗi dòng gồm ba số nguyên a, b, c cho biết đường đi từ tòa nhà a đến b có chi phí là c.

Output:

In ra một số duy nhất là trọng số cây khung nhỏ nhất tìm được.

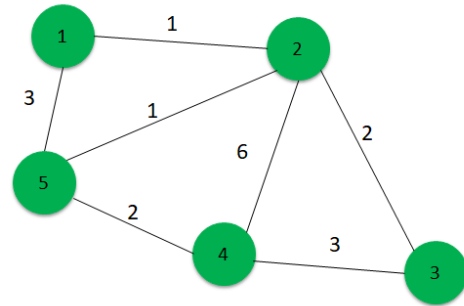
Ví dụ:

1 2 5 7 1 2 1 2 3 2 2 4 6 5 2 1 5 1 3 4 5 2 3 4 3	12
---	----

### Hướng dẫn giải:

Đồ thị trong ví dụ được biểu thị như hình bên. Để tìm cây khung nhỏ nhất bạn cần đi qua các cạnh sau:

- 1 – 2: 1
- 2 – 5: 1
- 2 – 3: 2
- 5 – 4: 2



Tổng chi phí là  $6 * 2$  (chi phí p) = 12.

Bài này bạn chỉ cần cài đặt Prim cơ bản sau đó tính chi phí cây khung nhỏ nhất rồi nhân với p là ra kết quả.

**Độ phức tạp:**  $O(t * m \log n)$  với t là số test case, m là số cạnh và n là số đỉnh.