Spanning Trees

Let $G = (V, E)$ be a connected, undirected graph. Write an efficient algorithm that computes the following.

- Computes the weight of a minimum spanning tree of G.
- Given a new edge (u, v, w) where $u, v \in V$ and w is the weight of the edge, compute the minimum spanning tree for the new graph $G' = (V, E \cup \{(u, v, w)\})$.

Input Format

The first line of each test will be three space separated positive integers $|V|$, $|E|$ and q denoting number of vertices and number of edges in input graph G and the number of queries respectively. Let the vertices of graph G be labelled with $\{0, \ldots, |V| - 1\}$.

Each of the following $|E| + q$ lines will contain three space separated positive integers u,v and w where (u, v) denotes the edge and w denotes the weight of that edge. The first $|E|$ lines denotes the edge of the original graph whereas the next q lines are the queries.

Constraints

- $1 \leq |V| \leq 10^4$
- $1 \leq |E| \leq 10^2$
- $1 \leq q \leq 10^8$

Output Format

First, print the weight of a minimum spanning tree of the original graph $G = (V, E)$. For each query u, v, w, print the weight of the minimum spanning tree of $G' = (V, E \cup \{(u, v, w)\})$.

Sample Input 0

```
6 5 2
0 1 2
1 2 2
1 3 2
2 4 2
3 5 2
2 3 2
2 3 1
```

Sample Output 0

```
10
10
9
```