The Courier

Byteland is a scarcely populated country, and residents of different cities seldom communicate with each other. There is no regular postal service and throughout most of the year a one-man courier establishment suffices to transport all freight. However, on Christmas Day there is somewhat more work for the courier than usual, and since he can only transport one parcel at a time on his bicycle, he finds himself riding back and forth among the cities of Byteland.

The courier needs to schedule a route which would allow him to leave his home city, perform the individual orders in arbitrary order (i.e. travel to the city of the sender and transport the parcel to the city of the recipient, carrying no more than one parcel at a time), and finally return home. All roads are bi-directional, but not all cities are connected by roads directly; some pairs of cities may be connected by more than one road. Knowing the lengths of all the roads and the errands to be performed, determine the length of the shortest possible cycling route for the courier.

Input

The input begins with the integer t, the number of test cases. Then t test cases follow.

Each test case begins with a line containing three integers: n m b, denoting the number of cities in Byteland, the number of roads, and the number of the courier's home city, respectively ($1 <= b <= n <= 100$, $1 <= m <= 10000$). The next m lines contain three integers each, the i-th being u_i v_i d_i, which means that cities u_i and v_i are connected by a road of length d_i ($1 <= u_i$, $v_i <= 100$, $1 <= d_i <= 10000$). The following line contains integer z - the number of transport requests the courier has received ($1<=z<=5$). After that, z lines with the description of the orders follow. Each consists of three integers, the j-th being u_j v_j b_j, which signifies that b_j parcels should be transported (individually) from city u_j to city v_j. The sum of all b_j does not exceed 12.

Output

For each test case output a line with a single integer - the length of the shortest possible bicycle route for the courier.

Example

Sample input:
```
1
5 7 2
1 2 7
1 3 5
1 5 2
2 4 10
2 5 1
3 4 3
3 5 4
3
1 4 2
5 3 1
5 1 1
```

Sample output:
43