# NOI '18 P1 - Return

**Time limit:** 4.0s **Memory limit:** 512M

You are given an undirected connected graph with $n$ vertices labeled from $1$ to $n$, and $m$ edges, where the $i$-th edge has length $l_i$ and altitude $a_i$.

You are given $Q$ queries. In each query, you are given a starting vertex $v$, a water level $p$. A water level $p$ indicates that there is water on the edges that have $a_i \le p$. You can drive a car starting from $v$ only through the edges that have **no water**, and then get out of the car at any vertex and walk to vertex $1$. You want to find the minimum sum of lengths of edges that you need to **walk** through in a path.

In some tests, the queries are forced online. See more details in the Input Specification/Constraints.

## Input Specification

The first line contains an integer $T$, the number of test cases.

For each test case,

- The first line contains two non-negative numbers $n$, the number of vertices, and $m$, the number of edges.
- Each of the next $m$ lines contains four integers $u, v, l, a$, indicating that the $i$-th edge is between vertices $u$ and $v$, and has length $l$ and altitude $a$.
- The next line contains three integers, $Q, K, S$. $K$ is a constant used in calculating the input each time, $S$ is the highest possible $p$ in all queries.
- Each of the next $Q$ lines contains two integers $v_0$ and $p_0$. Let `lastans` denote the answer in the last query ($0$ if this is the first query). Then in this query, you are starting from vertex $v = (v_0 + K \times \texttt{lastans} - 1) \pmod{n} + 1$ and water level $p = (p_0 + K \times lastans) \pmod{S+1}$. It is guaranteed that $1 \le v_0 \le n$, $0 \le p_0 \le S$.

## Output Specification

For each query, output the answer on its own line.

## Constraints

For all test files, $T \le 3$. For all test cases in a file:

- $n \le 2\times 10^5, m \le 4\times 10^5, Q \le 4 \times 10^5, K \in \{0, 1\}, 1 \le S \le 10^9$.
- For all edges, $1 \le u, v \le n$, $l \le 10^4$, $a \le 10^9$.
- The graph is connected.

The following terminology is used in the constraint table:

- Graph Property:
  - Bamboo: $m = n - 1$, and for each edge we have $u + 1 = v$.
  - Tree: $m = n - 1$
- Altitude:

    - One kind: For all edges, $a = 1$.
- Forced Online:
    - Yes: $K = 1$
    - No: $K = 0$

"no guarantee" means that there is no guarantee on the property of test data.

| $n$ | $m$ | $Q =$ | No. | Graph Property | Altitute | Forced Online |
|---|---|---|---|---|---|---|
| $\le 1$ | $\le 0$ | $0$ | $1$ | no guarantee | one kind | No |
| $\le 6$ | $\le 10$ | $10$ | $2$ | | | |
| $\le 50$ | $\le 150$ | $100$ | $3$ | | | |
| $\le 100$ | $\le 300$ | $200$ | $4$ | | | |
| $\le 1500$ | $\le 4000$ | $2000$ | $5$ | | | |
| $\le 200\,000$ | $\le 400\,000$ | $100\,000$ | $6$ | | | |
| $\le 1500$ | $= n - 1$ | $2000$ | $7$ | Bamboo | no guarantee | |
| | | | $8$ | | | |
| | | | $9$ | | | |
| $\le 200\,000$ | | $100\,000$ | $10$ | Tree | | |
| | | | $11$ | | | Yes |
| | $\le 400\,000$ | | $12$ | no guarantee | | No |
| | | | $13$ | | | |
| | | | $14$ | | | |
| $\le 1500$ | $\le 4000$ | $2000$ | $15$ | | | Yes |
| | | | $16$ | | | |
| $\le 200\,000$ | $\le 400\,000$ | $100\,000$ | $17$ | | | |
| | | | $18$ | | | |
| | | $400\,000$ | $19$ | | | |
| | | | $20$ | | | |

## Sample Input 1

```
1
4 3
1 2 50 1
2 3 100 2
3 4 50 1
5 0 2
3 0
2 1
4 1
3 1
3 2
```

## Sample Output 1

```
0
50
200
50
150
```

## Explanation for Sample Output 1

Day $1$ has no rain. so you can just travel directly.

Days $2$, $3$, $4$ have the same situation: the edges between $(1, 2)$ and $(3, 4)$ have water.

On day $2$, starting from $2$, you can only go to $3$, which doesn't help returning, so you have to walk.

On day $3$, the only edge out of $4$ has water, so you again have to walk.

On day $4$, you can first take the car to $2$, and then walk home.

On day $5$, all edges have water, so you have to walk.

## Sample Input 2

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

## Sample Output 2

```
0
2
3
1
```

## Explanation for Sample Output 2

This test case is forced online.

On day $1$ the answer is $0$, so day $2$ has $v = (5 + 0 - 1) \mod 5 + 1 = 5$, $p = (2 + 0) \mod (3 + 1) = 2$.

On day $2$, the answer is $2$, so day $3$ has $v = (2 + 2 - 1) \mod 5 + 1 = 4$, $p = (0 + 2) \mod (3 + 1) = 2$.

On day $3$, the answer is $3$, so day $4$ has $v = (4 + 3 - 1) \mod 5 + 1 = 2$, $p = (0 + 3) \mod (3 + 1) = 3$.