

# Roads

[English](#)

[Vietnamese](#)

N cities named with numbers 1 ... N are connected with one-way roads. Each road has two parameters associated with it: the road length and the toll that needs to be paid for the road (expressed in the number of coins). Bob and Alice used to live in the city 1. After noticing that Alice was cheating in the card game they liked to play, Bob broke up with her and decided to move away - to the city N. He wants to get there as quickly as possible, but he is short on cash. We want to help Bob to find the shortest path from the city 1 to the city N that he can afford with the amount of money he has.

## Input

The input begins with the number t of test cases. Then t test cases follow. The first line of the each test case contains the integer K,  $0 \leq K \leq 10000$ , maximum number of coins that Bob can spend on his way. The second line contains the integer N,  $2 \leq N \leq 100$ , the total number of cities. The third line contains the integer R,  $1 \leq R \leq 10000$ , the total number of roads. Each of the following R lines describes one road by specifying integers S, D, L and T separated by single blank characters : S is the source city,  $1 \leq S \leq N$  D is the destination city,  $1 \leq D \leq N$  L is the road length,  $1 \leq L \leq 100$ . T is the toll (expressed in the number of coins),  $0 \leq T \leq 100$  Notice that different roads may have the same source and destination cities.

## Output

For each test case, output a single line contain the total length of the shortest path from the city 1 to the city N whose total toll is less than or equal K coins. If such path does not exist, output -1.

## Example

**Input:**

```
2
5
6
7
1 2 2 3
2 4 3 3
3 4 2 4
1 3 4 1
4 6 2 1
3 5 2 0
5 4 3 2
0
4
4
1 4 5 2
1 2 1 0
2 3 1 1
3 4 1 0
```

**Output:**

```
11
```

