

Algorithms Lab

Exercise – Casino Royale

To cope with the latest round of budget cuts, MI6 asks its agents to use the railway system as their main means of transportation. With many upcoming missions in Central and Eastern Europe, travel planning has become intricate. Since Q is on leave, you have been asked to prepare a schedule for agents traveling on the Pendolino train from Switzerland to Montenegro.

Some missions are deemed more important than others. Therefore every mission has an assigned *priority*. Higher priority missions are to be handled preferentially. Each mission has been assigned exactly one agent. To execute a mission, the assigned agent has to enter the train at the given stop and leave it at her/his mission destination. Furthermore, agents should travel inconspicuously. If too many agents are on the train at the same time, the large consumption of drinks (Château Angelus 1981 and Vesper Martini) may raise suspicion. There is a single train. It starts at a station t_0 . It passes through t_1, t_2, \dots in that order and ends at t_{n-1} .

Your task is to come up with a travel schedule by selecting a subset of missions to be *executed*. The non-selected missions are *deferred*. The goal is to *maximize the sum of priorities* of the executed missions, while respecting the specified limit on the number of agents simultaneously on the train between any two consecutive stops. (Agents that enter the train at a station do not interfere with those leaving the train at that station.)

Input The first line of the input contains the number $t \leq 30$ of test cases. Each of the t test cases is described as follows:

- It starts with a line that contains three integers $n \ m \ \ell$, separated by a space. They denote
 - n , the number of train stops t_0, t_1, \dots, t_{n-1} ($2 \leq n \leq 3 \cdot 10^2$).
 - m , the number of missions ($1 \leq m \leq 10^5$).
 - ℓ , the maximum number of agents that can be on the train between any two consecutive stops ($1 \leq \ell \leq 10^2$).
- The following m lines describe the missions. Each mission is described by three integers $x \ y \ q$, separated by a space and such that $0 \leq x < y \leq n - 1$ and $1 \leq q \leq 2^7$. This means that the assigned agent wants to travel from t_x to t_y and that the mission has priority q .

Output For each test case output a line with one integer k that is the maximum *achievable priority*, defined as follows. A priority k is *achievable*, if there is a set of missions whose priorities sum up to k and such that no more than ℓ agents are on the train between any two consecutive stops.

Points There are four groups of test sets, which are worth 100 points in total.

1. For the first group of test sets, worth 20 points, you may assume that there are no more than $3 \cdot 10^2$ missions ($m \leq 3 \cdot 10^2$).
2. For the second group of test sets, worth 20 points, you may assume that all missions have the same unit priority ($q = 1$).
3. For the third group of test sets, worth 40 points, there are no additional assumptions.
4. For the fourth group of test sets, which is hidden and worth 20 points, there are no additional assumptions.

Corresponding sample test sets are contained in `testi.in/out`, for $i \in \{1, 2, 3\}$.

Sample Input

```
4
4 4 2
2 3 1
0 1 1
0 2 1
1 3 1
6 4 2
0 3 1
1 5 1
3 4 1
0 3 1
6 4 2
0 3 32
1 5 40
3 4 16
0 3 38
5 10 2
0 2 25
1 3 36
0 4 100
3 4 40
1 3 25
2 3 14
3 4 21
1 3 92
3 4 58
0 1 51
```

Sample Output

```
4
3
94
301
```