Asia-Pacific Informatics Olympiad 2012

Saturday, May 12th, 2012

Hosted by

The Japanese Committee for International Olympiad in Informatics (JCIOI)

Task name	**Dispatching**	**Guard**	**Kunai**
Time Limit	1.0 sec	1.0 sec	3.0 sec
Memory Limit	256 MB	256 MB	256 MB
Points	100	100	100
Input	stdin (keyboard)		
Output	stdout (screen)		

Language	**Compiler version**	**Compiler options**
C	gcc version 4.6.3	-m64 -O2 -lm
C++	g++ version 4.6.3	-m64 -O2 -lm
Pascal	fpc version 2.4.4	-O2 -Sd -Sh

Dispatching

In a sect of ninja, ninjas are dispatched to a client, and they are rewarded according to their work.

In this sect, there is one ninja called the Master. Every ninja except the Master has one and only one boss. In order to preserve the confidentiality and to encourage leadership, any instructions concerning their work are always sent by a boss to his/her subordinates. It is forbidden to send instructions by other methods.

You are gathering a number of ninjas and dispatch them to a client. You have to pay salaries to dispatched ninjas. For each ninja, the amount of salary for him/her is fixed. The total amount of salaries paid to them should be within a budget. Moreover, in order to send instructions, you have to choose a ninja as a manager who can send instructions to all dispatched ninjas. When instructions are sent, a ninja who is not dispatched may mediate the transmission. The manager may or may not be dispatched. If the manager is not dispatched, he will not be paid.

You would like to maximize the satisfaction level of the client as much as possible within a budget. The satisfaction level of the client is calculated as the product of the total number of dispatched ninjas and the leadership level of the manager. For each ninja, his/her leadership level is fixed.

Task

Write a program that, given the boss B_i, the amount of salary C_i, the leadership level L_i of each ninja i ($1 \leqq i \leqq N$), and the budget for salaries M, outputs the maximum value of the satisfaction level of the client when the manager and dispatched ninjas are chosen so that all the conditions are fulfilled.

Constraints

$1 \leqq N \leqq 100\,000$	The number of ninjas
$1 \leqq M \leqq 1\,000\,000\,000$	The budget for salaries
$0 \leqq B_i < i$	The boss of each ninja
$1 \leqq C_i \leqq M$	The amount of salary of each ninja
$1 \leqq L_i \leqq 1\,000\,000\,000$	The leadership level of each ninja

Input

Read the following data from the standard input.

- The first line of input contains two space separated integers N, M, where N is the number of ninjas and M is the budget.
- The following N lines describe the boss, salary, leadership level of each ninja. The $(i + 1)$-th line contains three space separated integers B_i, C_i, L_i, describing that the boss of ninja i is ninja B_i, the amount of his/her salary is C_i, and his/her leadership level is L_i. The ninja i is the Master if $B_i = 0$. Since the inequality $B_i < i$ is always satisfied, for each ninja, the number of his/her boss is always smaller than the number of himself/herself.

Output

Write the maximum value of the satisfaction level of the client to the standard output.

Grading

In test cases worth 30% of the full score, $N \leqq 3\,000$.

Sample Input and Output

Sample Input 1	Sample Output 1
5 4 0 3 3 1 3 5 2 2 2 1 2 4 2 3 1	6

If we choose ninja 1 as a manager and ninja 3, 4 as dispatched ninjas, the total amount of salaries is 4 which does not exceed the budget 4. Since the number of dispatched ninjas is 2 and the leadership level of the manager is 3, the satisfaction level of the client is 6. This is the maximum value.

Guard

The Kingdom of APIO is attacked by ninjas. Ninjas are very strong because, when they attack, they are hiding in the shadows and other people cannot see them. The Kingdom was captured except for the APIO castle, where the king lives. In front of the APIO castle, there is a line of N bushes. The bushes are numbered from 1 to N, and K ninjas are hiding in exactly K bushes. There are M guards in the APIO castle. The guard i is watching a sequence of bushes from the bush A_i to the bush B_i. Now, each guard reports to the king whether there is a ninja in the bushes he/she is watching. Since you are a servant of the king, you have to tell him, based on the reports from the guards, in which bush "a ninja is certainly hiding". Here, "a ninja is certainly hiding" in a bush if a ninja is hiding in it for any possible arrangement of ninjas which does not contradict the reports from the guards.

Task

Write a program that, given information of the guards and the reports from them, determines all bushes where "a ninja is certainly hiding".

Constraints

$1 \leqq N \leqq 100\,000$	The number of bushes
$1 \leqq K \leqq N$	The number of hidden ninjas
$1 \leqq M \leqq 100\,000$	The number of guards

Input

Read the following data from the standard input.

- The first line of input contains three space separated integers N, K, M, where N is the number of bushes, K is the number of hidden ninjas, and M is the number of guards.
- The following M lines contain the information of the guards and the reports from them. The i-th line of them contains three space separated integers A_i, B_i, C_i ($A_i \leqq B_i$), describing that the guard i is watching from the bush A_i to the bush B_i. The integer C_i is either 0 or 1. If $C_i = 0$, there is no ninja from the bush A_i to the bush B_i. If $C_i = 1$, there is at least one ninja from the bush A_i to the bush B_i.

For each input, it is guaranteed that there is at least one arrangement of ninjas which does not contradict the reports from the guards.

Output

If there is a bush where "a ninja is certainly hiding", output the numbers of the bushes where "a ninja is certainly hiding" to the standard output. The numbers of bushes should be written in ascending order, and one line of output should contain only one number. Therefore, if there are X bushes where "a ninja is certainly hiding", the output consists of X lines. If there is no bush where "a ninja is certainly hiding", output '`-1`' to the standard output.

Grading

In test cases worth 10% of the full score, $N \leqq 20$, $M \leqq 100$.
In test cases worth 50% of the full score, $N \leqq 1\,000$, $M \leqq 1\,000$.

Sample Input and Output

Sample Input 1	Sample Output 1

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

In this example, there are two possible arrangements of ninjas satisfying the conditions; three ninjas are hiding in the bush 1, 3, 5, or three ninjas are hiding in the bush 2, 3, 5.

Since a ninja is hiding in the bush 3 and 5 for any possible arrangements, we should output 3 and 5. Concerning the bush 1, there is a possible arrangement of ninjas where a ninja is hiding in the bush 1. But there is also a possible arrangement of ninjas where no ninja is hiding in the bush 1. Therefore, we should not output 1. By the same reason, we should not output 2.

Sample Input 2	Sample Output 2
`5 1 1` `1 5 1`	`-1`

In this example, there is no bush where "a ninja is certainly hiding". Therefore, we should output '`-1`'.

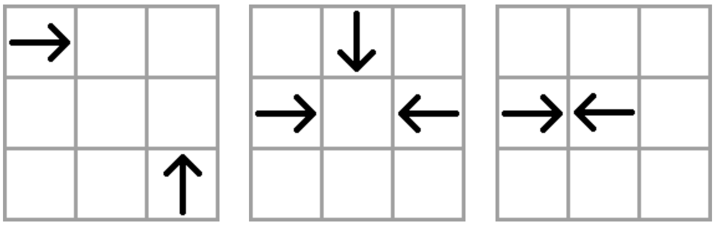
Kunai

Kunai is an acuate weapon used by ninjas whose shape is similar to a knife. Ninjas were attacking their enemies by throwing kunais against them.

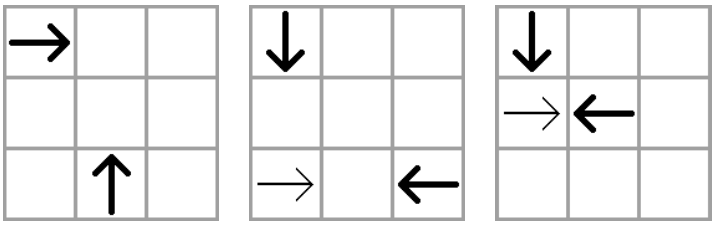
There are N ninjas in a grid of squares with W columns and H rows. Every ninja is in the center of a square, and no two ninjas share the same square. Each ninja has a kunai, and looks toward one of the four directions; up, down, left, or right. At time 0, every ninja threw his/her kunai to the direction he/she is looking toward.

Every kunai proceeds straight with speed 1. If more than one kunais come to the same place at the same time, they clash each other and disappear. The size of a kunai is so small that we can ignore it. Also, since ninjas can move quickly, they will not be hit by kunais. Each kunai continues to proceed along its direction without losing its speed unless it is clashed with another kunai.

In the following figures, the arrows represent kunais. The direction of an arrow describes the direction of a kunai. In these figures, all thick arrows will clash.



On the other hand, in each of the following figures, a thick arrow will not clash with another thick arrow. In the second and the third figure, a thin arrow clashes with a thick arrow. Because clashed arrows will disappear, a thick arrow will not clash with another thick arrow in each of these figures.



Task

Count the number of squares in the $W \times H$ grid where kunais pass through after a sufficient amount of time passed.

Constraints

$1 \leqq N \leqq 100\,000$	The number of ninjas
$1 \leqq W \leqq 1\,000\,000\,000,\ 1 \leqq H \leqq 1\,000\,000\,000$	The size of the grid
$1 \leqq X_i \leqq W,\ 1 \leqq Y_i \leqq H$	The coordinates of ninjas

Input

Read the following data from the standard input.

- The first line of input contains two space separated integers W, H, which describe the size of the grid.
- The second line of input contains an integer N, the number of ninjas.
- The i-th line ($1 \leqq i \leqq N$) of the following N lines contains three space separated integers X_i, Y_i, D_i, describing that the position of the ninja i is in the X_i-th column from left and Y_i-th row from above. No two ninjas share the same position. The direction of the ninja i is described by the value of D_i.
  - When $D_i = 0$, the ninja i is looking toward the right.
  - When $D_i = 1$, the ninja i is looking toward the up.
  - When $D_i = 2$, the ninja i is looking toward the left.
  - When $D_i = 3$, the ninja i is looking toward the down.

Output

Output the number of squares in the $W \times H$ grid where kunais pass through after a sufficient amount of time passed to the standard output.

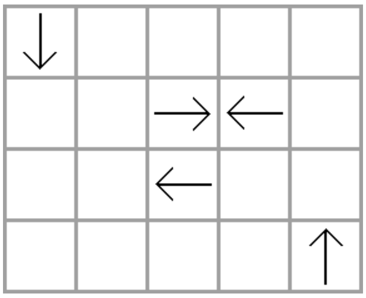
Grading

In test cases worth 10% of the full score, $N \leqq 1\,000,\ W \leqq 1\,000,\ H \leqq 1\,000$.
In test cases worth 40% of the full score, $N \leqq 1\,000$.

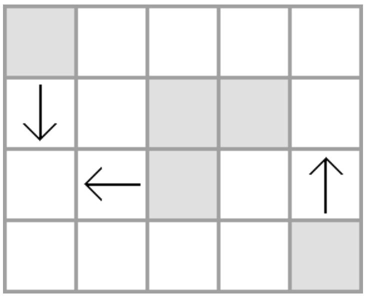
Sample Input and Output

Sample Input 1	Sample Output 1
5 4 5 3 3 2 3 2 0 4 2 2 5 4 1 1 1 3	11

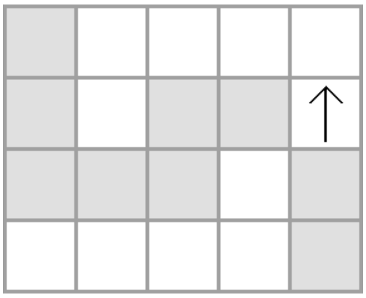
In this example, the grid at time 0 is described as follows.



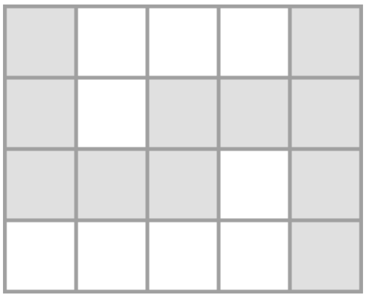
The kunai thrown by the ninja i is denoted by the kunai i. At time 0.5, the kunai 2 and the kunai 3 will clash each other and disappear. The following figure describes the grid at time 1. Here gray squares denote the squares where kunais already passed through.



At time 2, the kunai 1 and the kunai 5 will clash each other and disappear. The grid at time 2 is described as follows.



No more kunais are clashed in a square after time 2. The grid after a sufficient amount of time passed is described as follows.



Finally, the number of squares in the grid where kunais pass through is 11. Therefore, we should output 11.

Sample Input 2	Sample Output 2
`7 6`	`29`
`12`	
`3 2 3`	
`6 3 2`	
`7 1 3`	
`1 5 0`	
`3 6 1`	
`6 6 1`	
`4 5 2`	
`1 3 0`	
`6 5 2`	
`5 1 2`	
`6 4 3`	
`4 1 3`	