Task: TUG
Tug of War

english

BOI 2015, day 2. Available memory: 256 MB.   *1.05.2015*

Tug of war is a very popular sport in Byteland. The rules are easy: two teams pull a rope in opposite directions. The annual Byteland tug of war charity is taking place, and a lot of contestants have signed up. As the fair play commissioner, your job is to divide the contestants into two teams, such that the game can go on for a long time.

Since a total of $2n$ contestants have signed up, each team will consist of n contestants. The rope has n spots on the left side and n spots on the right side. The tug of war elite of Byteland are a picky bunch: each contestant has exactly one spot on the left side of the rope and one spot on the right side that he or she wants to use. Furthermore, you know the strength of each contestant.

The organizer has now asked you the following: Given an integer k, is it possible to create two teams, such that each team has n contestants, each contestant uses a spot that he or she wants to use (of course no two contestants share a spot), and the sums of strengths of the two teams differ by at most k?

Input

The first line of input contains a positive integer n, specifying the number of spots on each side of the rope, and an integer $k \leq 20n$, specifying the maximum difference of teams' strengths. For simplicity, we number the contestants from 1 to $2n$.

Each of the following $2n$ lines describes one contestant: the i-th of these lines contains three positive integers l_i, r_i and s_i ($1 \leq l_i, r_i \leq n$, $1 \leq s_i \leq 20$), which specify that contestant i has strength s_i and wants to use either spot l_i on the left side of the rope or spot r_i on the right side of the rope.

Output

In the first and only line of output your program should write either `YES` or `NO`, depending on whether it is possible to create two teams satisfying the requirements stated above.

Examples

For the input data:

```
4 1
1 1 1
2 1 2
2 2 8
1 2 2
3 3 5
3 3 2
4 4 1
4 4 2
```

a correct result is:

```
YES
```

For the input data:

```
2 5
1 1 1
1 2 4
2 2 1
2 1 4
```

a correct result is:

```
NO
```

Explanation to the examples: In the first example we can assign contestants 1, 3, 6 and 7 to the left side (which results in a team of strength $1 + 8 + 2 + 1 = 12$) and contestants 2, 4, 5 and 8 to the right side (which results in a team of strength $2 + 2 + 5 + 2 = 11$). The difference of strengths between teams is 1.

In the second example both players of strength 4 have to be in the same team, thus the minimal difference of strengths between teams is 6.

Grading

Subtask	Conditions	Points
1	$n \leq 10$	18
2	$n \leq 2000$	30
3	$n \leq 30\,000$, $s_i = 1$	23
4	$n \leq 30\,000$	29