Codeforces Beta Round #63 (Div. 2)

A. Young Physicist

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

A guy named Vasya attends the final grade of a high school. One day Vasya decided to watch a match of his favorite hockey team. And, as the boy loves hockey very much, even more than physics, he forgot to do the homework. Specifically, he forgot to complete his physics tasks. Next day the teacher got very angry at Vasya and decided to teach him a lesson. He gave the lazy student a seemingly easy task: You are given an idle body in space and the forces that affect it. The body can be considered as a material point with coordinates (0; 0; 0). Vasya had only to answer whether it is in equilibrium. "Piece of cake" — thought Vasya, we need only to check if the sum of all vectors is equal to 0. So, Vasya began to solve the problem. But later it turned out that there can be lots and lots of these forces, and Vasya can not cope without your help. Help him. Write a program that determines whether a body is idle or is moving by the given vectors of forces.

Input

The first line contains a positive integer n ($1 \le n \le 100$), then follow n lines containing three integers each: the x_i coordinate, the y_i coordinate and the z_i coordinate of the force vector, applied to the body ($-100 \le x_i, y_i, z_i \le 100$).

Output

Print the word "YES" if the body is in equilibrium, or the word "NO" if it is not.

Sample test(s)

input
3 4 1 7 -2 4 -1 1 -5 -3
output
NO

input
3 3 -1 7 -5 2 -4 2 -1 -3
output
YES

B. Bets

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In Chelyabinsk lives a much respected businessman Nikita with a strange nickname "Boss". Once Nikita decided to go with his friend Alex to the Summer Biathlon World Cup. Nikita, as a very important person, received a token which allows to place bets on each section no more than on one competitor.

To begin with friends learned the rules: in the race there are n sections of equal length and m participants. The participants numbered from 1 to m. About each participant the following is known:

- l_i — the number of the starting section,
- r_i — the number of the finishing section ($l_i \le r_i$),
- t_i — the time a biathlete needs to complete an section of the path,
- c_i — the profit in roubles. If the i-th sportsman wins on one of the sections, the profit will be given to the man who had placed a bet on that sportsman.

The i-th biathlete passes the sections from l_i to r_i inclusive. The competitor runs the whole way in $(r_i - l_i + 1) \cdot t_i$ time units. It takes him exactly t_i time units to pass each section. In case of the athlete's victory on k sections the man who has betted on him receives $k \cdot c_i$ roubles.

In each section the winner is determined **independently** as follows: if there is at least one biathlete running this in this section, then among all of them the winner is the one who has ran this section in minimum time (spent minimum time passing this section). In case of equality of times the athlete with the smaller index number wins. If there are no participants in this section, then the winner in this section in not determined. We have to say that in the summer biathlon all the participants are moving at a constant speed.

We should also add that Nikita can bet on each section and on any contestant running in this section.

Help the friends find the maximum possible profit.

Input

The first line contains two integers n and m ($1 \le n, m \le 100$). Then follow m lines, each containing 4 integers l_i, r_i, t_i, c_i ($1 \le l_i \le r_i \le n$, $1 \le t_i, c_i \le 1000$).

Output

Print a single integer, the maximal profit in roubles that the friends can get. In each of n sections it is not allowed to place bets on more than one sportsman.

Sample test(s)

input
4 4 1 4 20 5 1 3 21 10 3 3 4 30 3 4 4 20

output
60

input
8 4 1 5 24 10 2 4 6 15 4 6 30 50 6 7 4 20

output
105

Note

In the first test the optimal bet is: in the 1-2 sections on biathlete 1, in section 3 on biathlete 3, in section 4 on biathlete 4. Total: profit of 5 rubles for 1 section, the profit of 5 rubles for 2 section, profit of 30 rubles for a 3 section, profit of 20 rubles for 4 section. Total profit 60 rubles.

In the second test the optimal bet is: on 1 and 5 sections on biathlete 1, in the 2-4 sections on biathlete 2, in the 6-7 sections on athlete 4. There is no winner in the 8 section. Total: profit of 10 rubles for 1 section, the profit of 15 rubles for 2,3,4 section, profit of 10 rubles for a 5 section, profit of 20 rubles for 6, 7 section. Total profit 105 rubles.

C. Game

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In one school with Vasya there is a student Kostya. Kostya does not like physics, he likes different online games. Every day, having come home, Kostya throws his bag in the farthest corner and sits down at his beloved computer. Kostya even eats glued to the game. A few days ago Kostya bought a new RPG game "HaresButtle", which differs from all other games in this genre. It has a huge number of artifacts. As we know, artifacts are divided into basic and composite ones. Only the basic artifacts are available on sale. More powerful composite artifacts are collected from some number of basic artifacts.

After the composing composite artifact, all the components disappear.

Kostya is the head of the alliance, so he has to remember, what artifacts has not only himself, but also his allies. You must identify by sequence of artifacts purchased by Kostya and his allies, how many and which artifacts has been collected by each of them. It is believed that initially no one has any artifacts.

Input

The first line has 4 natural numbers: k ($1 \le k \le 100$) — the number of Kostya's allies, n ($1 \le n \le 50$) — the number of basic artifacts, m ($0 \le m \le 50$) — the number of composite artifacts, q ($1 \le q \le 500$) — the number of his friends' purchases. The following n lines contain the names of basic artifacts. After them m lines contain the descriptions of composite artifacts in the following format:

```
<Art. Name>: <Art. №1> <Art. №1 Number>, <Art. №2> <Art. №2 Number>, ... <Art. №X> <Art. №X Number>
```

All the numbers are natural numbers not exceeding 100 ($1 \le X \le n$).

The names of all artifacts are different, they are composed of lowercase Latin letters, and the length of each name is from 1 to 100 characters inclusive. All the words in the format of the description of a composite artifact are separated by exactly one space. It is guaranteed that all components of the new artifact are different and have already been met in the input data as the names of basic artifacts.

Next, each of the following q lines is characterized by the number a_i, the number of a friend who has bought the artifact ($1 \le a_i \le k$), and the name of the purchased basic artifact. Let's assume that the backpacks of the heroes are infinitely large and any artifact bought later can fit in there.

It is guaranteed that after the i-th purchase no more than one opportunity to collect the composite artifact appears. If such an opportunity arose, the hero must take advantage of it.

Output

The output file should consist of k blocks. The first line should contain number b_i — the number of different artifacts the i-th ally has. Then the block should contain b_i lines with the names of these artifacts and the number of these artifacts. At that the lines should be printed in accordance with the lexicographical order of the names of the artifacts. In each block all the artifacts must be different, and all the numbers except the b_i should be positive.

Sample test(s)

input

```
2 3 2 5
desolator
refresher
perseverance
vanguard: desolator 1, refresher 1
maelstorm: perseverance 2
1 desolator
2 perseverance
1 refresher
2 desolator
2 perseverance
```

output

```
1
vanguard 1
2
desolator 1
maelstorm 1
```

D. Dot

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Anton and Dasha like to play different games during breaks on checkered paper. By the 11th grade they managed to play all the games of this type and asked Vova the programmer to come up with a new game. Vova suggested to them to play a game under the code name "dot" with the following rules:

- On the checkered paper a coordinate system is drawn. A dot is initially put in the position (x, y).
- A move is shifting a dot to one of the pre-selected vectors. Also each player can once per game symmetrically reflect a dot relatively to the line $y = x$.
- Anton and Dasha take turns. Anton goes first.
- The player after whose move the distance from the dot to the coordinates' origin exceeds d, loses.

Help them to determine the winner.

Input

The first line of the input file contains 4 integers x, y, n, d ($-200 \le x, y \le 200, 1 \le d \le 200, 1 \le n \le 20$) — the initial coordinates of the dot, the distance d and the number of vectors. It is guaranteed that the initial dot is at the distance less than d from the origin of the coordinates. The following n lines each contain two non-negative numbers x_i and y_i ($0 \le x_i, y_i \le 200$) — the coordinates of the i-th vector. It is guaranteed that all the vectors are nonzero and different.

Output

You should print "`Anton`", if the winner is Anton in case of both players play the game optimally, and "`Dasha`" otherwise.

Sample test(s)

input
0 0 2 3 1 1 1 2
output
Anton

input
0 0 2 4 1 1 1 2
output
Dasha

Note

In the first test, Anton goes to the vector (1;2), and Dasha loses. In the second test Dasha with her first move shifts the dot so that its coordinates are (2;3), and Anton loses, as he has the only possible move — to reflect relatively to the line $y = x$. Dasha will respond to it with the same move and return the dot in position (2;3).

E. Subsegments

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Programmer Sasha has recently begun to study data structures. His coach Stas told him to solve the problem of finding a minimum on the segment of the array in $O(\log n)$, which Sasha coped with. For Sasha not to think that he had learned all, Stas gave him a new task. For each segment of the fixed length Sasha must find the maximum element of those that occur on the given segment exactly once. Help Sasha solve this problem.

Input

The first line contains two positive integers n and k ($1 \le n \le 10^5$, $1 \le k \le n$) — the number of array elements and the length of the segment.

Then follow n lines: the i-th one contains a single number a_i ($-10^9 \le a_i \le 10^9$).

Output

Print $n-k+1$ numbers, one per line: on the i-th line print of the maximum number of those numbers from the subarray $a_i\ a_{i+1}\ \ldots\ a_{i+k-1}$ that occur in this subarray exactly 1 time. If there are no such numbers in this subarray, print "Nothing".

Sample test(s)

input

```
5 3
1
2
2
3
3
```

output

```
1
3
2
```

input

```
6 4
3
3
3
4
4
2
```

output

```
4
Nothing
3
```