

Like pachinko? Here is another one. It's not exactly a traditional pachinko, but it's also a let-the-ball-hit-things game.

In the machine, there are  $n$  non-overlapping non-vertical bars, shown below. Here non-overlapping means for every pair of bars, the two segments do not intersect, do not have common end-point, and do not partially overlap (they may overlap when projected to  $x$ -axis, though).



At the  $i$ -th step, the ball will be transferred to  $(x_i, y_i)$ , then start to fall vertically, hopefully it'll hit a bar and earn some scores. A ball who hit the  $i$ -th bar will earn a score of  $s_i$ . If the ball directly drops on the floor (with  $y = 0$ ), it will not score.

The most interesting part of the machine is: **if the  $i$ -th bar is hit during this step, it will disappear at that moment and re-appear after  $d_i$  steps**. For example, if a bar with  $d_i = 3$  is hit in the 5-th step, then it'll be missing during step 6 and 7, and will re-appear in step 8.

## Input

There will be at most 5 test cases. Each test case begins with one integer  $n$  ( $1 \leq n \leq 10^5$ ), the number of bars. Each of the next lines contains 6 integers  $x_1, y_1, x_2, y_2, s, d$  ( $0 \leq x_1 < x_2 \leq 10^9$ ,  $1 \leq y_1, y_2 \leq 200000$ ,  $1 \leq s \leq 1000$ ,  $1 \leq d \leq 5$ ), describing one bar. No two bars can have any common point (i.e. no intersection, can't touch each other etc).

The next line contains  $b$  ( $1 \leq b \leq 10^5$ ), the number of balls. In the next  $b$  lines, the  $i$ -th line describes the ball appear in the  $i$ -th step. Each line contains two integers  $(x', y')$ , that means the ball will appear at  $(x_i, y_i) = (x' \text{ XOR } a, y' \text{ XOR } a)$ , where  $a$  is the current score before the ball falls (which will be zero at the beginning of each test case). It is guaranteed that  $x_i$  and  $y_i$  are non-negative integers and will not be precisely on a bar.

## Output

For each test case, print the case number in the first line and the scores after each step. There should be one empty line after each test case.

### Explanation for Sample 1

- Step 1: ball (3,5) will hit the first bar, *score* = 1
- Step 2: ball (3,5) will hit the second bar, *score* = 9
- Step 3 (bar 2 appear again): ball (1,5) will hit the ground, *score* = 0
- Step 4: ball (3,3) will hit the second bar, *score* = 9
- Step 5 (bar 1&2 appear again): ball (3,9) will hit the first bar again, *score* = 1

## Sample Input

```
2
0 4 4 4 1 4
2 2 6 2 9 1
5
3 5
2 4
11 15
9 9
16 26
3
0 6 10 7 1 5
2 4 8 3 10 5
4 2 6 2 100 5
4
5 7
4 6
14 12
106 104
```

## Sample Output

Case 1:

```
1
10
10
19
20
```

Case 2:

```
1
11
111
111
```