G. Shooting Gallery

time limit per test: 5 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Berland amusement park shooting gallery is rightly acknowledged as one of the best in the world. Every day the country's best shooters master their skills there and the many visitors compete in clay pigeon shooting to win decent prizes. And the head of the park has recently decided to make an online version of the shooting gallery. During the elaboration process it turned out that the program that imitates the process of shooting effectively, is needed. To formulate the requirements to the program, the shooting gallery was formally described. A 3D Cartesian system of coordinates was introduced, where the X axis ran across the gallery floor along the line, along which the shooters are located, the Y axis ran vertically along the gallery wall and the positive direction of the Z axis matched the shooting direction. Let's call the XOY plane a shooting plane and let's assume that all the bullets are out of the muzzles at the points of this area and fly parallel to the Z axis. Every clay pigeon can be represented as a rectangle whose sides are parallel to X and Y axes, and it has a positive z-coordinate. The distance between a clay pigeon and the shooting plane is always different for every target. The bullet hits the target if it goes through the inner area or border of the rectangle corresponding to it. When the bullet hits the target, the target falls down vertically into the crawl-space of the shooting gallery and cannot be shot at any more. The targets are tough enough, that's why a bullet can not pierce a target all the way through and if a bullet hits a target it can't fly on. In input the simulator program is given the arrangement of all the targets and also of all the shots in the order of their appearance. The program should determine which target was hit by which shot. If you haven't guessed it yet, you are the one who is to write such a program.

Input

The first line contains an integer n $(1 \le n \le 10^5)$ — the number of targets. Each of the subsequent n lines contains the description of a target. The target is described by five integers x_l, x_r, y_l, y_r, z, that determine it's location in space $(0 \le x_l < x_r \le 10^7, 0 \le y_l < y_r \le 10^7, 0 < z \le 10^7)$. The next line contains an integer m $(1 \le m \le 10^5)$, determining the number of shots. Then in m lines shots are described. Every shot is determined by the coordinates of a bullet on the shooting plane (x, y) $(0 \le x, y \le 10^7$, the coordinates of bullets are integers). The shots are given in the order of their firing. The intervals between shots are large enough, and a target falls very quickly, that's why assume that a falling target can not be an obstruction for all the shots following the one that hit it.

Output

For every shot in the single line print the number of the target which the shot has hit, or 0, if the bullet did not hit any target. The targets are numbered starting from 1 in the order in which they were given in the input data.

Examples

input

```
2
1 4 1 4 1
2 5 2 6 2
4
0 0
3 3
4 5
3 5
```

output

```
0
1
2
0
```