

MMCC '15 P2 - Akame

Time limit: 7.0s **Memory limit:** 1G

Akame is training on an infinite 2D Cartesian plane. There is an infinitely long rigid and vertical string at each integer x -coordinate. She takes out her katana, Murasame, and makes N infinitely long slashes on the plane. Each slash can be seen as a line with the equation $Ax + By = C$, for some A , B , and C . A slash will never be a vertical line.

Each string Akame slashes will immediately be broken into two smaller strings. For example, a string originally from $(5, -\infty)$ to $(5, \infty)$ severed by the line $0x + 1y = 3$ will be broken into the two strings $(5, -\infty)$ to $(5, 3)$ and $(5, 3)$ to $(5, \infty)$. A string is considered disconnected if it is of a finite length. For example, the string from $(1, 9)$ to $(1, \infty)$ is not of finite length, but the string from $(8, -3)$ to $(8, 4)$ is.

There are M points on strings Akame wants to disconnect. A point is said to be disconnected if it is either slashed or the string it's on is disconnected. To measure her performance, Akame would like to know the earliest moment (that is, after which slash) each point is disconnected. The slashes are numbered from 1 to N .

Input Specification

The first line of input will have N and M , separated by a single space.

The next N lines of input will each describe one of Akame's slashes, with A , B , and C separated by spaces.

The next M lines of input will each describe a point Akame wants to disconnect, with X_i and Y_i .

Output Specification

There should be M lines of output. The i^{th} line should contain the smallest index of the slash that after it is performed, the i^{th} point is disconnected. If the i^{th} point is never disconnected, print `-1` instead.

Constraints

$$1 \leq N \leq 500\,000$$

$$1 \leq M \leq 500\,000$$

$$0 \leq A, |C| \leq 30\,000$$

$$1 \leq |B| \leq 30\,000$$

$$0 \leq |X_i|, |Y_i| \leq 10^9$$

Sample Input 1

```
3 5
0 1 3
0 1 5
0 1 7
1 2
1 3
1 4
1 5
1 6
```

Sample Output 1

```
-1
1
2
2
3
```

Sample Input 2

```
10 10
81 -36 72
64 -69 24
23 47 47
83 36 18
1 25 77
12 -81 -3
13 -90 -34
23 19 -34
19 23 78
50 -96 82
56 59
85 47
46 -23
1 13
-74 -69
23 99
-98 72
-31 -65
-78 76
9 -69
```

Sample Output 2

```
2
3
4
-1
2
-1
4
2
4
-1
```