

Fracture Ray

Input file: **standard input**
Output file: **standard output**
Time limit: 10 seconds
Memory limit: 512 megabytes

A monster is attacking our city! Our hero, Little Rabbit, comes to rescue the city. Little Rabbit has a very powerful skill — Fracture Ray. The rays can penetrate the monster and cause huge damage to it. But the monster also has a defensive skill — Hall of Mirrors. Several mirrors will be placed on the battleground, which can reflect the rays. However, since the rays are too powerful, each mirror can only reflect the ray once, and it will break down afterward.

We can regard the battleground as a two-dimensional plane, which can be described by Cartesian coordinates. There are n double-sided mirrors on the plane, which can be considered as segments that are parallel to the x -axis or y -axis. The coordinates of their endpoints are all integers.

Little Rabbit will emit m rays. Each ray starts from a point and has an initial direction which is at an angle of 45° to the x -axis and y -axis. When the ray hits a mirror, it will be reflected specularly (the angle of reflection equals the angle of incidence). And it will also break the mirror down, which means the mirror will disappear. Little Rabbit will emit the rays one by one. Only if a ray goes out of range, which means it won't hit any more mirrors, Little Rabbit will emit the next ray. It's guaranteed that the ray won't go through any points whose coordinates are integers. See the input format for more information.

Little Rabbit has the confidence to defeat the monster. However, he wonders which mirrors the i -th ray will hit.

Input

The first line of the input contains an integer T ($1 \leq T \leq 2$) — the number of test cases.

The first line of each test case contains two integers n, m ($1 \leq n, m \leq 10^5$) — the number of mirrors and rays.

Then in the next n lines, the i -th line contains four integers x_1, y_1, x_2, y_2 ($0 \leq x_1, y_1, x_2, y_2 \leq 10^9$), which means the endpoints of the i -th mirror are (x_1, y_1) and (x_2, y_2) . It's guaranteed that $x_1 = x_2$ or $y_1 = y_2$, and $(x_1, y_1) \neq (x_2, y_2)$. It's also guaranteed that the intersection of every pair of mirrors is an empty set or only a point.

Then in the next m lines, the i -th line contains four integers x, y, z, d ($0 \leq x, y \leq 10^9$, $0 \leq z \leq 1$, $0 \leq d \leq 3$), indicating the start point and the initial direction of the i -th ray.

The first three integers x, y, z indicate the start point of the ray.

- $z = 0$: the ray starts at $(x + 0.5, y)$.
- $z = 1$: the ray starts at $(x, y + 0.5)$.

The last integer d indicates the initial direction of the ray. We use a vector to describe the direction.

- $d = 0$: the initial direction $\vec{s} = (1, -1)$.
- $d = 1$: the initial direction $\vec{s} = (-1, -1)$.
- $d = 2$: the initial direction $\vec{s} = (-1, 1)$.
- $d = 3$: the initial direction $\vec{s} = (1, 1)$.

It's guaranteed that the start point of a ray won't be on the segment of any mirror.

Output

The output of the x -th test case begins with *Case #x*: in a single line.

Then in the next m lines, the first integer of the i -th line is y , which means the i -th ray hits y mirrors in total. Then there follows y integers, which means the i -th ray hits the a_1 -th, a_2 -th, \dots , a_y -th mirror in order. These y integers must be given in the order of the ray hitting the mirror.

Example

standard input	standard output
1	Case #1:
4 2	2 1 3
1 0 6 0	1 4
5 4 4 4	
1 3 3 3	
5 6 4 6	
7 2 1 1	
0 2 0 3	