Problem H. Half Plane

Input file:	*standard input*
Output file:	*standard output*
Time limit:	12 seconds
Memory limit:	1024 mebibytes

This problem might be well-known in some countries, but how do other countries learn about such problems if nobody poses them.

There are n points on the plane, where the i-th point (x_i, y_i) has value $\mathbf{d}_i \in D$. Two sets D and O are given, with the following properties:

- There exists a special element ε_D in D.
- There exists a special element ε_O in O.
- A binary operation $+ : D \times D \to D$ is given with the following properties:
  - $\forall \mathbf{a}, \mathbf{b} \in D$, $\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}$
  - $\forall \mathbf{a}, \mathbf{b}, \mathbf{c} \in D$, $(\mathbf{a} + \mathbf{b}) + \mathbf{c} = \mathbf{a} + (\mathbf{b} + \mathbf{c})$
  - $\forall \mathbf{x} \in D$, $\mathbf{x} + \varepsilon_D = \varepsilon_D + \mathbf{x} = \mathbf{x}$
- A binary operation $\cdot : O \times D \to D$ is given with the following properties:
  - $\forall \mathbf{a}, \mathbf{b} \in O, \mathbf{x} \in D$, $(\mathbf{a} \cdot \mathbf{b}) \cdot \mathbf{x} = \mathbf{a} \cdot (\mathbf{b} \cdot \mathbf{x})$
  - $\forall \mathbf{a} \in O, \mathbf{x}, \mathbf{y} \in D$, $\mathbf{a} \cdot (\mathbf{x} + \mathbf{y}) = \mathbf{a} \cdot \mathbf{x} + \mathbf{a} \cdot \mathbf{y}$
- A binary operation $\cdot : O \times O \to O$ is given with the following properties:
  - $\forall \mathbf{a}, \mathbf{b}, \mathbf{c} \in O$, $(\mathbf{a} \cdot \mathbf{b}) \cdot \mathbf{c} = \mathbf{a} \cdot (\mathbf{b} \cdot \mathbf{c})$
  - $\forall \mathbf{x} \in O$, $\mathbf{x} \cdot \varepsilon_O = \varepsilon_O \cdot \mathbf{x} = \mathbf{x}$

In this problem, we treat D as the set of all 3×1 matrices over $\mathbb{F}_p$ and O as the set of all 3×3 matrices over $\mathbb{F}_p$, where $p = 10^9 + 7$. That is, you can treat the above operations as the usual matrix addition and matrix multiplication modulo $10^9 + 7$.

Now, m queries are given in the form `a b c o`:

- Let $\mathbf{s} = \varepsilon_D$.
- For all points i with $ax_i + by_i < c$, modify $\mathbf{s}$ to $\mathbf{s} + \mathbf{d}_i$, then modify $\mathbf{d}_i$ to $\mathbf{o} \cdot \mathbf{d}_i$.
- Return $\mathbf{s}$ as the answer of the query.

As a data structure master, you need to perform all queries and find the answer.

Input

The first line of the input contains a single integer n ($1 \le n \le 3 \cdot 10^5$), indicating the number of points.

Each of the following n lines contains **five** integers $x_i, y_i, d_{i0}, d_{i1}, d_{i2}$, indicating the coordinates of the i-th point and its value $\mathbf{d}_i = \begin{bmatrix} d_{i0} \\ d_{i1} \\ d_{i2} \end{bmatrix}$.

The next line of the input contains a single integer m ($1 \le m \le 1.5 \cdot 10^4$), indicating the number of the queries.

Each of the following m lines contains **twelve** integers $a, b, c, o_{00}, o_{01}, o_{02}, o_{10}, \ldots, o_{22}$. Note that the real $\mathbf{o} = \begin{bmatrix} o_{00} & o_{01} & o_{02} \\ o_{10} & o_{11} & o_{12} \\ o_{20} & o_{21} & o_{22} \end{bmatrix}$.

It is guaranteed that:

- $|x_i| \le 10^6$, $|y_i| \le 10^6$.
- $|a_i| \le 10^3$, $|b_i| \le 10^3$, $b_i \ne 0$, $|c_i| \le 10^6$.
- All matrix elements are from 0 to $10^9 + 6$ inclusive.
- For all $1 \le i \le m$ and $1 \le j \le n$, $a_i x_j + b_i y_j \ne c_i$.
- For all $1 \le i \le m$ and $1 \le j \le m$, $\left(\frac{a_i}{b_i}, \frac{c_i}{b_i}\right) \ne \left(\frac{a_j}{b_j}, \frac{c_j}{b_j}\right)$.

Output

For each query, output a single line containing three integers s_0, s_1, s_2, indicating $\mathbf{s} = \begin{bmatrix} s_0 \\ s_1 \\ s_2 \end{bmatrix}$.

Example

standard input	standard output
5 1 1 2 3 4 12 12 4 6 1 1 12 5 1 2 12 1 1 5 5 6 6 2 0 3 3 1 1 4 1 1 2 3 4 5 2 3 4 1 1 400 1 3 4 2 1 2 3 4 5 -1 -1 -10 3 2 1 4 6 5 4 3 2	2 3 4 25 50 40 92 58 139

Note

Note that the solution does not depend on other properties of matrix addition/multiplication than those mentioned in the statements. Defining D and O as sets of matrices is only for testing convenience (since we can't use the graders or interaction libraries).