L. 2D Geometry

There are n **distinct** points on a 2-dimension plane. The coordinates of the i-th point is (x_i, y_i).

If there are three points A, B and C which form a triangle ABC with **positive area**, Bobo can remove them simultaneously from the plane. Also, if there are multiple triangles with positive area, Bobo can choose to remove any of them. Find the minimum number of points left on the plane if he can perform the operation for **any number of times**.

Input

The input consists of several test cases terminated by end-of-file. For each test case,

The first line contains an integer n.

For the following n lines, the i-th line contains two integers x_i and y_i.

- $1 \le n \le 2 \times 10^5$
- $0 \le x_i, y_i \le 10^9$ for each $1 \le i \le n$
- $(x_i, y_i) \ne (x_j, y_j)$ for each $1 \le i < j \le n$
- In each input, the sum of n does not exceed 2×10^5.

Output

For each test case, output an integer which denotes the minimum number of points left.

Sample Input

```
3
0 0
0 1
0 2
3
0 0
0 1
1 0
6
0 0
0 1
0 2
0 3
1 1
1 2
```

Sample Output

```
3
0
0
```

Note

For the third test case, if Bobo chooses to remove the triangle $\{(0,1),(1,1),(1,2)\}$ first, there will be no other triangles to remove. Alternatively, Bobo can remove the triangle $\{(0,0),(0,1),(1,1)\}$ first and then $\{(0,2),(0,3),(1,2)\}$.