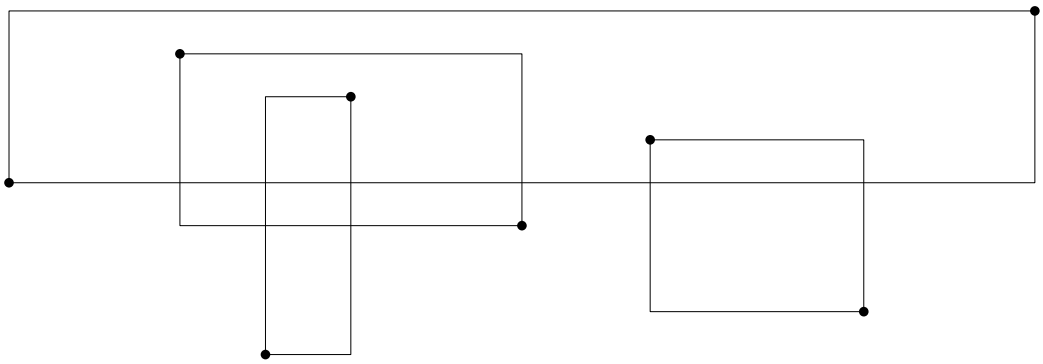
Thin Rectangles

An axis-aligned rectangle in the plane can be represented by its two opposite corners. In other words, any pair of points in the plane defines an axis-aligned rectangle with the points as opposite corners. The figure below shows 4 rectangles among 28 axis-aligned rectangles that are defined by 28 pairs of 8 points.



The *thinness* of a rectangle is defined to be the fraction a/b, where a is the length of the short side and b is the length of the long side of the rectangle. Thus, a square has thinness value 1 and any other rectangle has thinness value smaller than 1.

You are given N points in the plane. Write a program that returns the perimeter of an axis-aligned rectangle with smallest thinness value among all axis-aligned rectangles defined by pairs of input points.

[Input]
The first line contains the total number of test cases, T ($T \leq 60$), given in the input. Then the test cases follow. In each test case, the first line has one integer number N ($0 \leq N \leq 100{,}000$).
In each of the next N lines, two integers a, b ($-1{,}000{,}000{,}000 \leq a, b \leq 1{,}000{,}000{,}000$) are given, where a and b correspond to the x-coordinate and the y-coordinate of an input point, respectively. There are no two input points with the same x-coordinate or the same y-coordinate.

There are two types of input sets.

- Set 1: $N \leq 1{,}000$
- Set 2: $N \leq 100{,}000$

[Output]
For each test case, you should print the perimeter of an axis-aligned rectangle with smallest thinness value among all axis-aligned rectangles defined by pairs of input points. If there are more than one rectangle with smallest thinness, print the smallest perimeter among them.

[I/O Example]

Input

```
3
3
0 0
2 2
1 1
5
10 7
1 2
6 0
5 10
0 5
8
0 0
8 2
15 1
12 -1
4 3
20 -3
6 -4
24 4
```

Output

```
4
22
42
```