

Carol and Carla are roommates. Yesterday they threw a big party and today they have a partially eaten cake that they want to divide. Since people were careless when cutting themselves a slice, the cake is now shaped as a prism with its top and bottom faces being the same simple convex polygon.

To add some fun to the process of dividing the cake, the girls came up with the following game. Carol chooses a vertex v of the top face of the cake. Carla chooses another vertex w of the top face that is not adjacent to v . Then, they cut the cake into two pieces by extending downwards the segment vw , so as to obtain two separate pieces of cake, each in the shape of a prism. Finally, Carol chooses the piece that she prefers, and Carla gets the other one. Carla immediately saw that this system gives Carol an advantage. Carla wants to know exactly how much of an advantage Carol has.

You are given a polygon that represents both the top and bottom faces of the cake. The height of the cake is 2, so the volume of a piece of cake is 2 times the area of its top face. Assuming the cake is divided as explained, and that both girls make their decisions to maximize the volume of the piece they have at the end, compute the volume of the piece each girl will get.

Input

The input contains several test cases; each test case is formatted as follows. The first line contains an integer N representing the number of vertices of the polygonal top face of the cake ($4 \leq N \leq 10^5$). Each of the next N lines describes a vertex of the polygon with two integers X and Y , indicating the coordinates of the vertex in the XY plane ($-10^8 \leq X, Y \leq 10^8$). Vertices are given in counter clockwise order and define a simple convex polygon. No three points in the input are collinear.

Output

For each test case in the input, output a line with two integers representing the volume of the piece Carol and Carla will get, in that order, if both make their decisions optimally.

Sample Input

```
5
0 0
3 0
3 1
2 2
0 1
6
0 1
1 0
2 0
3 1
2 2
0 2
4
-100000000 -100000000
100000000 -100000000
100000000 100000000
-100000000 100000000
4
-99999995 -100000000
100000000 -100000000
100000000 99999995
-100000000 100000000
```

Sample Output

```
7 2
6 3
400000000000000000 400000000000000000
39999999999999975 399999980000000025
```