

4581 Minimal Bounding Rectangle

Given N points on the plane, namely, $(x[0], y[0]), (x[1], y[1]), \dots, (x[N - 1], y[N - 1])$, a **divider line** is one that passes through two of these points such that all the N points are on one side of the line or on the line itself. Given any divider line $L(P, P')$ that passes through the two points P and P' , a unique bounding rectangle $R(P, P')$ of minimal area can be found that contains all the N points inside the rectangle or on its sides, such that one side of $R(P, P')$ is on $L(P, P')$. Your problem is to write a computer program that will determine all such bounding rectangles of minimal area such that one side is on a divider line, and report the the smallest area among all such rectangles.

For example, given the six points $A(0, 0)$, $B(0, 2)$, $C(1, 1)$, $D(2, 2)$, $E(3, 1)$, and $F(4, 0)$. The divider line passing through AB defines a bounding rectangle of minimal area 8 square units. The divider line passing through AF defines a bounding rectangle of minimal area 8 square units. The divider line passing through DF defines a bounding rectangle of minimal area 12 square units. Finally the divider line passing through BD defines a bounding rectangle of minimal area 8 square units. Thus the bounding rectangle of smallest area has 8 square units.

Input

Input shall consist of several data sets. Each data set will be given in several lines. The first line of the data set will contain the value of N . This will be followed by N lines, each line containing the x and y coordinates of one point, separated by spaces. The value of N will not exceed 500. The coordinates x and y will be integers not exceeding 1000 in absolute value. Data for the next data set will immediately follow the data for the previous data set.

An input value of $N = 0$ signifies the end of input.

Output

For each data set, print one line of output of the form 'Data set N : A ', where N is the data set number, starting from 1, and A is the area of the bounding rectangle that is the smallest, given to four decimal places. Note that two sides of a bounding rectangle may meet at a point with non-integer coordinates.

Sample Input

```
6
0 0
0 2
1 1
2 2
3 1
4 0
3
0 1
1 0
1 1
0
```

Sample Output

Data set 1: 8.0000

Data set 2: 1.0000