YONSEI UNIVERSITY

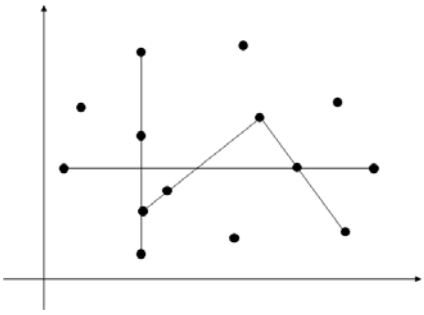
CSI3108-01 2015. 09. 10

Programming HW#2

Max 35 points

Due on Sept. 17 (Thursday), 2015, by 5 pm

Given a set of n points in the 2-dimensional space, first find every line segment that connects 3 or more distinct points in the set, and then compute the number of crossing points made by these line segments.



Input

The test cases consist of the following format. In the first line, the number of test cases is given. From the next line, each test case is provided in n+1 lines. The first line of each test case has a single integer n, where $5 \le n \le 10{,}000$. The next n lines have the x- and y-coordinates of n points (two integers per line). Note that the x- and y-coordinates of a point are positive integers, and there are no crossing points made by 2 or more collinear line segments.

Output

For each test case, print out the number k of crossing points in the first line made by the line segments of 3 or more collinear points. The next k lines should show the x- and y-coordinates of a crossing point per line. These k numbers should be printed in the lexicographical order. Note that if there is no crossing point, print '0' only in a separate line. Place a blank between two adjacent numbers for printing and print also a single blank line between the outputs of two test cases.

Sample Input

```
20      // the no of test cases.

6       // the no n of points in test case #1
1 1     // point #1     x=1, y=1
1 2     // point #2     x=1, y=2
1 3
1 4
2 2
3 3
        // place '\n' between test cases.
7       // test case #2
1 1
1 2
1 3
2 1
2 2
3 1
3 3
```

```
7       // test case #3
1 1
1 2
1 3
1 5
2 4
3 4
4 4

...
```

Sample Output

```
1       // the no of crossing points in test case #1.
1 1

4       // the no of crossing points in test case #2.
1 1     // four crossing points in lexicographic order.
1 3
2 2
3 1

0       // if crossing point doesn't exist, print '0.'

...
```