UCF "Practice" Local Contest — Aug 23, 2014

Polygon Restoration
filename: `polygon`
(Difficulty Level: `Hard`)

A rectangular polygon is a closed figure with all vertices at points with integer coordinates in the XY-plane, and whose edges are all either horizontal or vertical. The vertices are all distinct, and no two edges intersect, except for neighboring edges intersecting at their common vertex. For the purposes of this problem, every horizontal edge will be adjacent to a vertical edge, and vice versa, so all angles are either 90 or 270 degrees.
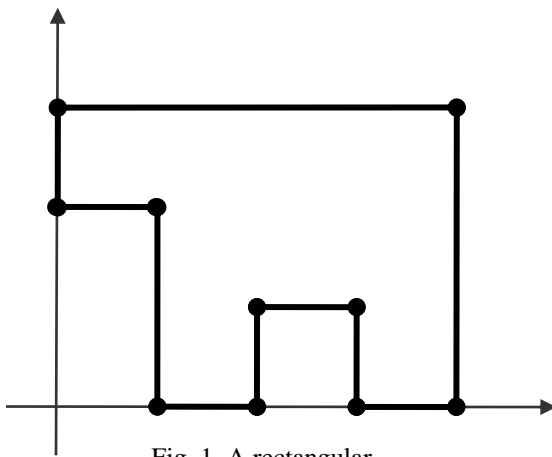

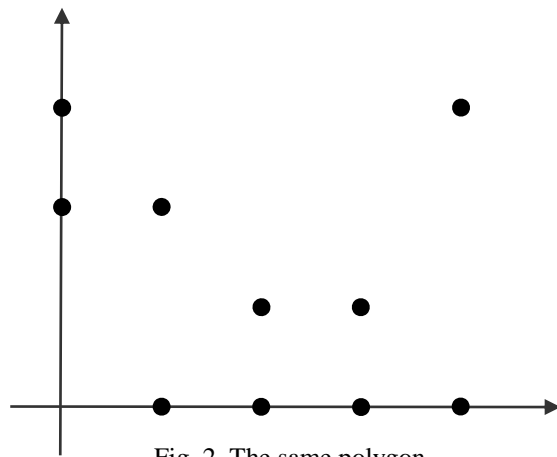Fig. 1. A rectangular
polygon in the XY-plane

Fig. 2. The same polygon,
with all of its edges erased

The Problem:

Dr. O is an avid collector of polygons. He takes particular pride in his collection of pencil-rendered rectangular polygons from the early 19th century. Unfortunately, some sneaky vandal has broken into his collection and erased all the edges, leaving only the vertices of each polygon. You have been called in to try to restore Dr. O's collection to its former glory by redrawing the edges of the rectangular polygons.

The Input:

There will be multiple polygons in the input file. The first line of every polygon description will be an integer N ($4 \leq N \leq 50$), the number of vertices of the polygon. Each of the next N input lines will contain two integers, giving the coordinates of a vertex of a polygon in the form "X Y" ($-10000 \leq X, Y \leq 10000$). Note that these vertices are given in arbitrary order. All vertices will be distinct.

End of input will be indicated by a case with N = 0. This case should not be processed.

The Output:

For each test case, output a single line, formatted as: "Polygon #t:", where t is the test case number (starting from 1), a single space, followed by the vertices of the polygon in counterclockwise order (with a single space separating vertices). Points should be referred to by their number in the order they were given in the input, the first input being vertex 1 (see Sample Output for clarification). The list must start from the vertex with minimum y-coordinate. If there are multiple points with minimum y-coordinate, use the one with minimum x-coordinate. It is guaranteed that a closed rectangular polygon can always be constructed from the given data.

Sample Input:

```
4
0 0
1 1
0 1
1 0
10
0 3
4 0
1 2
2 1
3 0
4 3
0 2
3 1
2 0
1 0
0
```

Sample Output:

```
Polygon #1: 1 4 2 3
Polygon #2: 10 9 4 8 5 2 6 1 7 3
```