Problem E
Onion Layers

Source file name: `onion.c`, `onion.cpp`, `onion.java` *or* `onion.pas`

Dr. Kabal, a well recognized biologist, has recently discovered a liquid that is capable of curing the most advanced diseases. The liquid is extracted from a very rare onion that can be found in a country called Onionland. But not all onions of Onionland are worth to take to the lab for processing. Only those onions with an odd number of layers contain the miraculous liquid. Quite an odd discovery!

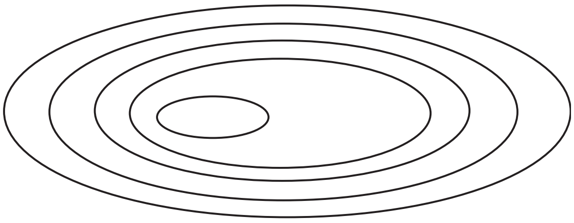

Figure 1: Onion from Onionland

Dr. Kabal has hired a lot of research assistants to collect and analyse onions for him. Since he does not want to share his discovery with the world yet, he didn't tell the assistants to look for onions with an odd number of layers. Instead, each assistant was given the task of collecting onions, and selecting points from each of the layer's outer borders, so that an approximation of the layer structure of the onion can be reconstructed later. Dr. Kabal told the assistants that the next step will be a "complicated analysis" of these points. In fact, all he will do is simply to use the points to count the number of layers in each of the onions, and select the ones with an odd number of layers.

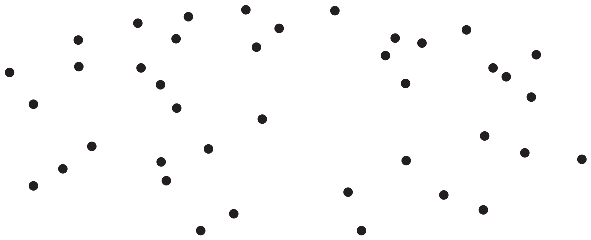

Figure 2: Points collected by an assistant

It is clear that the approximation obtained by Dr. Kabal, from the points collected, might have a different *shape* than the original onion. For instance, only some of the points of the onion shown in Figure 1 would be extracted in the process, giving rise to a set of points as shown in Figure 2. With these points Dr. Kabal will try to approximate the original layers of the onion, obtaining something like what is shown in Figure 3. The approximation procedure followed by Dr. Kabal (whose result is shown in Figure 3) is simply to recursively find nested convex polygons such that at the end every point belongs to precisely one of the polygons. The assistants have been told to select points in such a way that the *number of layers in the* *approximation, if done in this recursive manner, will be the same as in the original onion*, so that is fine with Dr. Kabal. The assistants are also aware that they need at least three points to approximate a layer, even the innermost one.

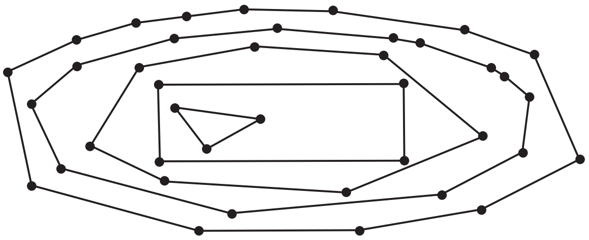

Figure 3: Dr. Kabal's approximation

Your task is to write a program that, given a set of points collected by an assistant (as shown in Figure 2), determines if the respective onion should be taken to the laboratory.

Input

The input contains several test cases. Each test case consists of an integer $3 \leq N \leq 2000$ in a single line, indicating the number of points collected by the assistants. Following, there are N lines, each containing two integers $-2000 \leq X, Y \leq 2000$ corresponding to the coordinates of each point. The input is finished by a problem with $N = 0$ points, which should not be processed.

The input must be read from standard input.

Output

There should be one line of output for each test case in the input. For each test case print the string

```
Take this onion to the lab!
```

if the onion should be taken to the laboratory or

```
Do not take this onion to the lab!
```

if the onion should not be taken to the laboratory.

The output must be written to standard output.

Sample input

```
7
0 0
0 8
1 6
3 1
6 6
8 0
8 8
11
2 6
3 2
6 6
0 0
0 11
1 1
1 9
7 1
7 9
8 10
8 0
0
```

Output for the sample input

```
Do not take this onion to the lab!
Take this onion to the lab!
```