Challenge Question 2

Cut-off Date: Until it is solved. 30 points

Note that this a challenge question. The first correct answer will get all the points. Once a right answer is accepted, no points will be given to other submissions. You need to present and argue why your approach is right.

Maximum Planar Subset

A chord is a line segment that joins two points on a curve. You are given a set C which contains n chords of a circle. No two chords in the set C share an common endpoint. The endpoints of these chords are numbered, in a clockwise fashion, from 0 to $2n-1$ around the circle. Figure 1(a) provides an example.

Your task is to output the maximum number of chords in C that do not intersect each other. Figure 1(b) provides a solution that contains the maximum number of non-intersecting chords. This problem is known as the *Maximum Planar Subset* problem.

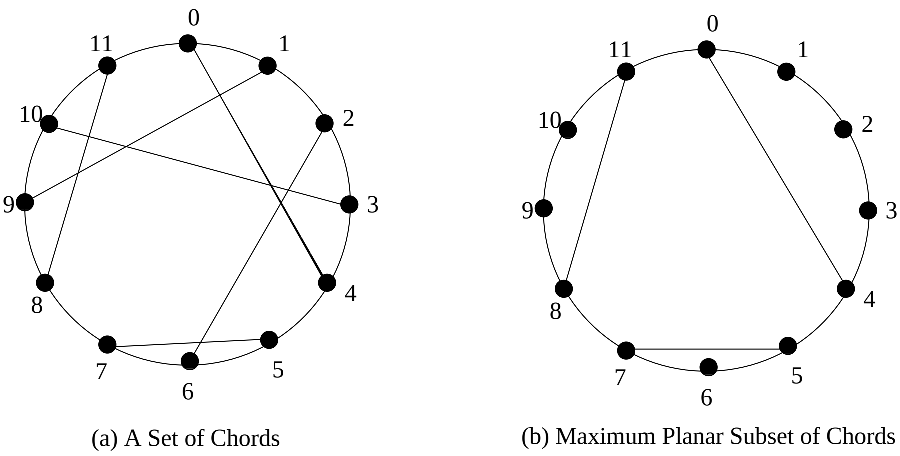

Figure 1: An Example

The Input

The input consists of several circles with different configurations. The configuration of each circle is specified by an integer $2n, 1 \leq n \leq 100$, on a single line, denoting the number of vertices on the circle. This is followed by n lines, each containing two integers, a and b, where $0 \leq a, b \leq 2n-1$, denoting the two endpoints of a chord. A single 0 (zero) in the input line at the beginning of a new configuration signifies the end of the input file.

The Output

For each circle configuration, print the number of chords in the maximum planar subset.

Sample Input

```
12
0 4
1 9
2 6
3 10
5 7
8 11
4
0 2
1 3
0
```

Sample Output

```
3
1
```