# H: **Hip-***n*

*Source file name:* `hipn.c, hipn.cpp, hipn.java,` *or* `hipn.py`
*Author:* Rodrigo Cardoso

Hip-*n* is a game in which two players take turns by placing tokens on the free cells of a non-empty $n \times n$ checkerboard. The game is lost by the first player placing four tokens identifying the vertices of a square: they can be of any size and tipped at any angle. The game ends in a tie when the board is full of tokens and no player has lost.

The following figure depicts a $6 \times 6$ checkerboard and three examples of squares: the first player putting four tokens on the vertices of any of these squares loses the game. Of course, there are many more options for losing a game in the $6 \times 6$ checkerboard.
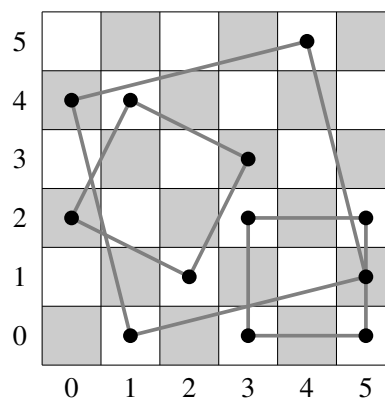


Your task is to create a program that decides the outcome of a Hip-*n* game described as a sequence of plays, by identifying the player that loses or recognizing a tie.

## Input

The input consists of several test cases. It ends when there are no more cases to test.

The first line of each test case contains an integer $n$ ($1 \le n \le 200$) indicating the number of rows and columns of the checkerboard. The next line contains $n^2$ distinct pairs of blank-separated integers $r$ and $c$ in the checkerboard ($0 \le r < n$ and $0 \le c < n$): each such a pair identifies the placement of a token at row $r$ and column $c$ by the corresponding player. You can assume that player 1 makes the first move, player 2 the second one, player 1 the third one, and so on.

*The input must be read from standard input.*

## Output

For each test case, print a single line with 0 if the game ends in a tie, 1 if player 1 loses, and 2 if player 2 loses.

*The output must be written to standard output.*

| Sample Input | Sample Output |
|---|---|
| 3 | 0 |
| 1 0 1 1 2 1 0 2 0 1 2 0 0 0 1 2 2 2 | 1 |
| 3 | 2 |
| 1 0 1 1 2 1 0 2 0 1 2 0 1 2 0 0 2 2 | |
| 3 | |
| 1 0 2 2 2 1 0 0 1 2 0 2 1 1 2 0 0 1 | |