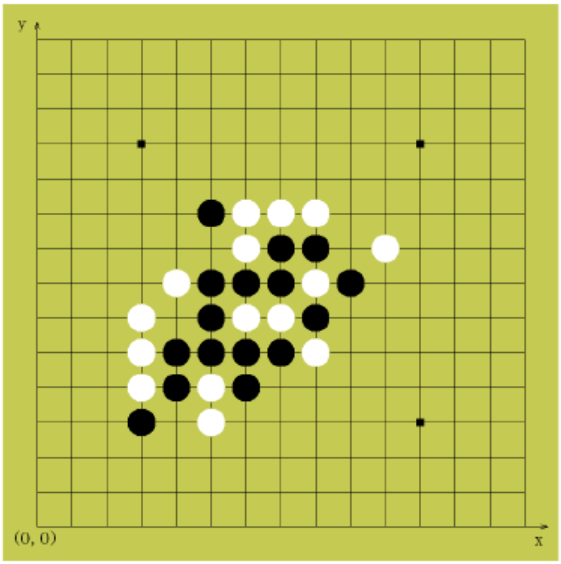
4836 Gomoku

You are probably not familiar with the title, "Gomoku", but you must have played it a lot. Gomoku is an abstract strategy board game and is also called Five in a Row, or GoBang. It is traditionally played with go pieces (black and white stones) on a go board (19×19 intersections). Nowadays, standard chessboard of Gomoku has 15×15 intersections. Black plays first, and players alternate in placing a stone of their color on an empty intersection. The winner is the first player to get an unbroken row of five or more stones horizontally, vertically, or diagonally.



For convenience, we coordinate the chessboard as illustrated above. The left-bottom intersection is (0,0). And the bottom horizontal edge is x-axis, while the left vertical line is y-axis.

I am a fan of this game, actually. However, I have to admit that I don't have a sharp mind. So I need a computer program to help me. What I want is quite simple. Given a chess layout, I want to know whether someone can win within 3 moves, assuming both players are clever enough. Take the picture above for example. There are 31 stones on it already, 16 black ones and 15 white ones. Then we know it is white turn. The white player must place a white stone at (5,8). Otherwise, the black player will win next turn. After that, however, the white player also gets a perfect situation that no matter how his opponent moves, he will win at the 3-rd move.

So I want a program to do similar things for me. Given the number of stones and positions of them, the program should tell me whose turn it is, and what will happen within 3 moves.

Input

The input contains no more than 20 cases.

Each case contains $n + 1$ lines which are formatted as follows.

n
$x_1\ y_1\ c_1$
$x_2\ y_2\ c_2$
.....
$x_n\ y_n\ c_n$

The first integer n indicates the number of all stones. $n \leq 222$ which means players have enough space to place stones. Then n lines follow. Each line contains three integers: x_i and y_i and c_i. x_i and y_i are coordinates of the stone, and c_i means the color of the stone. If $c_i = 0$ the stone is white. If $c_i = 1$ the stone is black. It is guaranteed that $0 \leq x_i, y_i \leq 14$, and $c_i = 0$ or 1. No two stones are placed at the same position. It is also guaranteed that there is no five in a row already, in the given cases.

The input is ended by $n = 0$.

Output

For each test case:

First of all, the program should check whose turn next. Let's call the player who will move next "Mr. Lucky". Obviously, if the number of the black stone equals to the number of white, Mr. Lucky is the black player. If the number of the black stone equals to one plus the numbers of white, Mr. Lucky is the white player. If it is not the first situation or the second, print '`Invalid.`'

A valid chess layout leads to four situations below:

1. Mr. Lucky wins at the 1st move. In this situation, print :

   `Place` $TURN$ `at` (x, y) `to win in 1 move.`

   "$TURN$" must be replaced by '`black`' or '`white`' according to the situation and (x, y) is the position of the move. If there are different moves to win, choose the one where x is the smallest. If there are still different moves, choose the one where y is the smallest.

2. Mr. Lucky's opponent wins at the 2nd move. In this situation, print:

   `Lose in 2 moves.`

3. Mr. Lucky wins at the 3rd move. If so, print:

   `Place` $TURN$ `at` (x, y) `to win in 3 moves.`

   "$TURN$" should replaced by '`black`' or '`white`', (x, y) is the position where the Mr. Lucky should place a stone at the 1st move. After he place a stone at (x, y), no matter what his opponent does, Mr. Lucky will win at the 3rd step. If there are multiple choices, do the same thing as described in situation 1.

4. Nobody wins within 3 moves. If so, print:

   `Cannot win in 3 moves.`

Sample Input

```
31
3 3 1
3 4 0
3 5 0
3 6 0
4 4 1
4 5 1
4 7 0
5 3 0
5 4 0
5 5 1
```

```
5 6 1
5 7 1
5 9 1
6 4 1
6 5 1
6 6 0
6 7 1
6 8 0
6 9 0
7 5 1
7 6 0
7 7 1
7 8 1
7 9 0
8 5 0
8 6 1
8 7 0
8 8 1
8 9 0
9 7 1
10 8 0
1
7 7 1
1
7 7 0
0
```

Sample Output

```
Place white at (5,8) to win in 3 moves.
Cannot win in 3 moves.
Invalid.
```