Google Kickstart '17 Round C Problem B - X Squared

Time limit: 1.0s **Memory limit:** 64M

The hot new toy for this year is called "X Squared". It consists of a square N by N grid of tiles, where N is odd. Exactly $2 \times N - 1$ of the tiles are labeled with an `X`, and the rest are blank (which we will represent with the `.` character). In each move of the game, the player can either choose and exchange two rows of tiles, or choose and exchange two columns of tiles. The goal of the game is to get all of the `X` tiles to be on the two main diagonals of the grid, forming a larger X shape, as in the following example for $N = 5$:

```
X...X
.X.X.
..X..
.X.X.
X...X
```

You are about to play with your X Squared toy, which is not yet in the goal state. You suspect that your devious younger sibling might have moved some of the tiles around in a way that has broken the game. Given the current configuration of the grid, can you determine whether it is possible to win or not?

Input Specification

The first line of the input gives the number of test cases, T. T test cases follow. Each one begins with one line with an integer N, the size of the grid. N more lines with N characters each follow; the jth character on the ith of these lines is `X` if the tile in the ith row and j^{th} column of the grid has an X, or `.` if that tile is blank.

Output Specification

For each test case, output one line containing `Case #x: y`, where `x` is the test case number (starting from 1) and `y` is `POSSIBLE` if it is possible to win, and `IMPOSSIBLE` otherwise.

Limits

$1 \leq T \leq 100$

$N \bmod 2 = 1$ (N is odd.)

The grid has exactly $2 \times N - 1$ `X` tiles and exactly $N^2 - 2 \times N + 1$ `.` tiles.

The grid is not already in the goal state, as described in the problem statement.

Subtask 1 [9/23]

$3 \leq N \leq 5$

Subtask 2 [14/23]

$3 \leq N \leq 55$

Sample Input

```
2
3
..X
XX.
XX.
3
...
XXX
XX.
```

Sample Output

```
Case #1: POSSIBLE
Case #2: IMPOSSIBLE
```

In Sample Case #1, one winning strategy is:

1. Swap the top row with the middle row.
2. Swap the rightmost column with the middle column.

```
..X    XX.    X.X
XX. -> ..X -> .X.
XX.    XX.    X.X
```

In Sample Case #2, no sequence of moves can turn the grid into the desired final configuration.