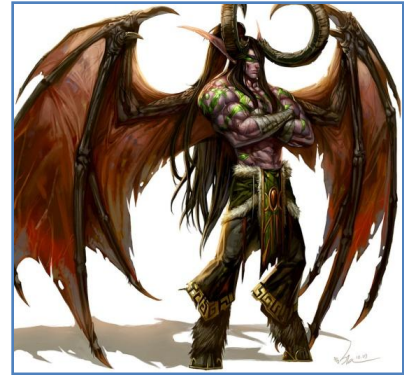


1426 – Blind Escape

Illidan Stormrage, the blind Warcraft hero is thrown to a maze for punishment. Being a blind hero, he cannot see anything but he can sense which direction is north. He memorized the map of the maze when he was a child. But he doesn't know his exact location now.



Assume that the maze is laid out on a grid, and each grid location is either blocked or free. He has four possible moves: **North**, **South**, **East** and **West**. He wants to find the shortest sequence of moves that will guarantee his escape (his initial location can be any free cell in the grid). He is said to be escaped if he is outside the maze and once he is out of the maze, further moves are irrelevant. And of course if he tries to walk into a wall, he will simply stay in the same spot.

As he cannot see anything, for any move, he will only stop if he bumps into a wall or he is out of the maze. Now your task is to help him by finding the shortest sequence of moves.

Input

Input starts with an integer **T** (≤ 200), denoting the number of test cases.

Each case starts with a line containing two integers **M** and **N** ($1 \leq M, N \leq 12$) denoting the number of rows and columns of the grid respectively. Each of the next **M** lines contains **N** characters (either '.' or '#') denoting the maze. '.' means free location and '#' means a wall. Assume that there is at least one free location in the grid.

Output

For each case, print the case number first. If it's impossible to do so, print **"Impossible"**. Otherwise, print the shortest possible sequence of moves that guarantees his escape. Show the sequence by the first letters of the moves. As there can be many solutions, print the one that comes lexicographically earliest. See the samples for details.

Sample Input	Output for Sample Input
2 4 8 ##### #...#.. ##....## ##.##### 3 4 #### #..# ####	Case 1: ESWWS Case 2: Impossible