Problem H. Humbertov and the Maze

Source file name:	Maze.c, Maze.cpp, Maze.java, Maze.py
Input:	`Standard`
Output:	`Standard`
Time / Memory limit:	`2/3/4 (C++/Java/Python) second(s) / 128 megabytes`
Author(s):	`Hugo Humberto Morales & Gabriel Gutiérrez - UTP Colombia`

One of Professor **Humbertov Moralov**'s passions is the analysis of mazes. He is now interested in identifying the distance of the farthest pair of cells in the largest region of cells within the maze that are walkable between them.

For simplicity, you can consider the maze as a grid (a matrix) of cells. A cell can be a wall or a passage (a cell through which you can walk). At any moment, Professor Moralov can move to a new cell from his current position if they share a side and both are passage cells.

Since you are a student in the Data Structures course, Professor Humbertov Moralov needs your help to solve this challenge. You must write a program to calculate the distance of the farthest pair of walkable cells in the largest region of cells inside the maze through which you can walk between them.

Note: In the case where the maximum region size is shared by several regions, you must calculate which of them has the maximum distance between the farthest pair of cells.

It is guaranteed that for any pair of cells within a passage cell region (a region through which you can walk), there is a unique path between them.

Input

The input begins with a positive integer T ($1 \leq T \leq 10$), denoting the number of test cases.

Each test case starts with a line containing two positive integers H and W (High, Wide, $3 \leq H, W \leq 1000$). The test case continues with H lines, each of which contains W characters. Each character represents the status of a cell in the maze as follows:

1. '`.`' - to indicate that it is a passage cell (a cell through which you can walk).
2. '`#`' - to indicate that it is a wall cell.

Output

For each test case, print a line with the following format: `Case idCase: R T D`, where `idCase` is replaced by the test case number, `R` is replaced by the number of isolated regions of walkable passage cells in the maze, `T` is replaced by the size of the largest independent region of walkable cells, and `D` is the minimum distance between the farthest pair of cells within the largest walkable region. For clarity, refer to the input and output examples below.

Example

Input	Output
3 11 11 #.......#.. ..#.#.#...# #.#.#.##### ..#.#.#.... .#########. ....####.####. ....#.##.#. ###.#.#..#. .#..#.#.##. ...##.#.... 11 11 #...#...#.. ..#.#.#...# #.#.#.##### ..#.#.#.... .#########. .#........ .####.####. ....#.##.#. ###.#.#..#. .#..#.#.##. ...##.#.... 11 11 #...#...#.. ..#.#.#...# #.#.#.##### ..#.#.#.... .########## .#........ .####.####. ....#.####. ###.#.#..#. .#..#.#.##. ...##.#....	Case 1: 1 69 39 Case 2: 3 31 22 Case 3: 4 25 23

Use fast I/O methods