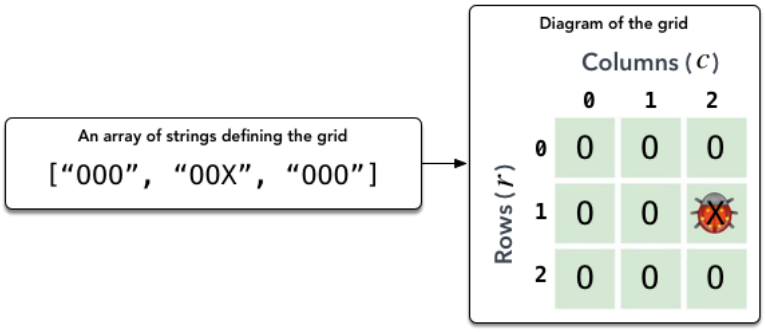
Find the Bug

by shashank21j

Problem Submissions Leaderboard Discussions

Consider an $n \times n$ grid where the top-left coordinate is $(0, 0)$ and the bottom-right coordinate is $(n-1, n-1)$. We define the contents of the grid as an array of n strings of length n, where the index r of a string corresponds to a *row* in the grid and the index c of a character in a string corresponds to a column. Each string consists of the characters `0` and/or `X`, where an `0` denotes an empty cell and an `X` denotes a cell containing a bug.

For example, if $grid = [\texttt{"000"}, \texttt{"00X"}, \texttt{"000"}]$, then the grid has a bug at location $(1, 2)$ and looks like this:



Given an array of strings defining a grid with 1 bug in it, print the bug's location in the format `r,c` (where r is the row and c is the column).

Input Format

The first line contains an integer denoting n (the length and width of the grid).
Each line r of the n subsequent lines contains a string of n characters describing row r in the grid.

Constraints

- $1 \leq n \leq 10^3$

Output Format

Print the bug's location in the format `r,c`, where r is its row and c is its column.

Note: If using the code stubs in the editor, return an array of two integers where index 0 contains the value of r and index 1 contains the value of c.

Sample Input 0

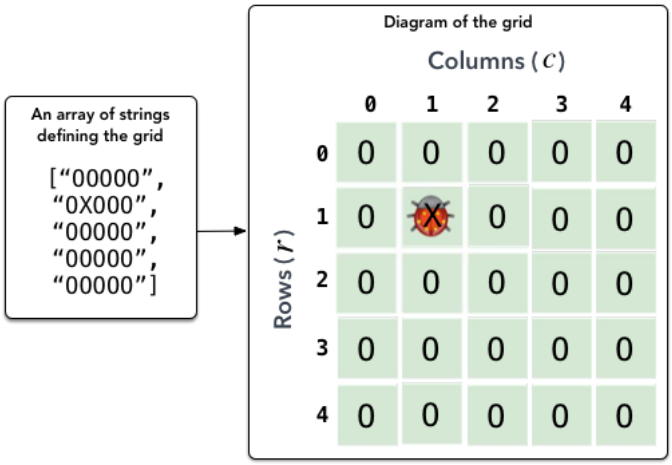
```
5
00000
0X000
00000
00000
00000
```

Sample Output 0

```
1,1
```

Explanation 0

The diagram below depicts the array and grid:



The bug is located at the intersection of row $r = 1$ and column $c = 1$, so we print `1,1`.

Contest ends in a day

Submissions: 1709
Max Score: 10
Difficulty: Easy

Rate This Challenge:

More

Current Buffer (saved locally, editable) C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
class Solution {

    static int[] findTheBug(string[] grid){
        // Complete this function
        for (int i = 0; i < grid.Length; i++)
            {
                for (int j = 0; j < grid[i].Length; j++)
                {
                    if (grid[i][j] == 'X')
                    {
                        return new int[] { i, j };
                    }
                }
            }
            return new int[] { -1, -1 };
    }

    static void Main(String[] args) {
        int n = Convert.ToInt32(Console.ReadLine());
        string[] grid = new string[n];
        for(int grid_i = 0; grid_i < n; grid_i++){
           grid[grid_i] = Console.ReadLine();
        }
        // Return an array containing [r, c]
        int[] result = findTheBug(grid);
        Console.WriteLine(String.Join(",", result));


```

```
    }
}
```

Line: 19 Col: 41

Upload Code as File Test against custom input Run Code Submit Code