Solution: Regions Cut by Slashes

Let's solve the Regions Cut By Slashes problem using the Union Find pattern.

We'll cover the following

Statement

An $n \times n$ grid is composed of n, 1×1 squares, where each 1×1 square consists of a "/", "\", or a blank space. These characters divide the square into adjacent regions.

Given the grid represented as a string array, return the number of regions.

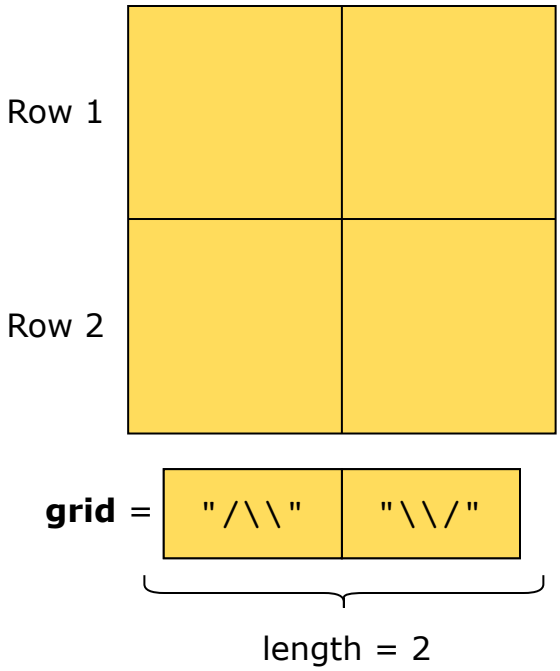
> **Note:**
>
> 1. Backslash characters are escaped, so "\" is represented as "\\".
> 2. A 1×1 square in the grid will be referred to as a box.

Constraints:

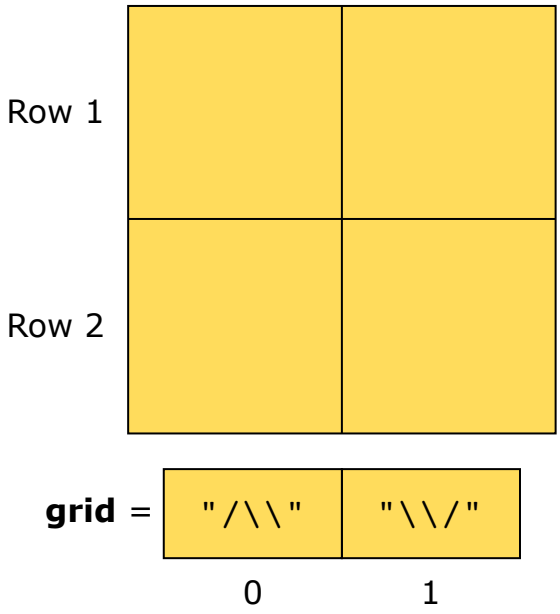
- The grid consists of only "/", "\", or " " characters.
- $1 \leq$ `grid.length` ≤ 30

The following demonstration shows how the grid can be visualized:

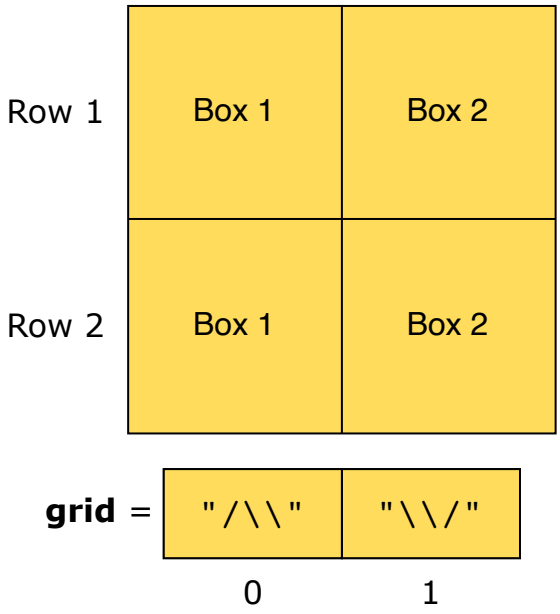
The number of elements in the **grid** array represent the value of n in the nxn grid. In this case, the length is 2, so we have a 2x2 grid.



Each index in the **grid** list represents a row.

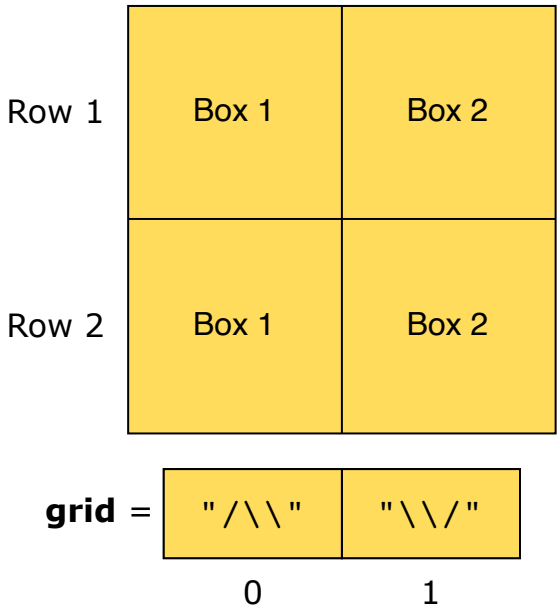


Each character in a row represents a slash or space character inside the respective box in the row.

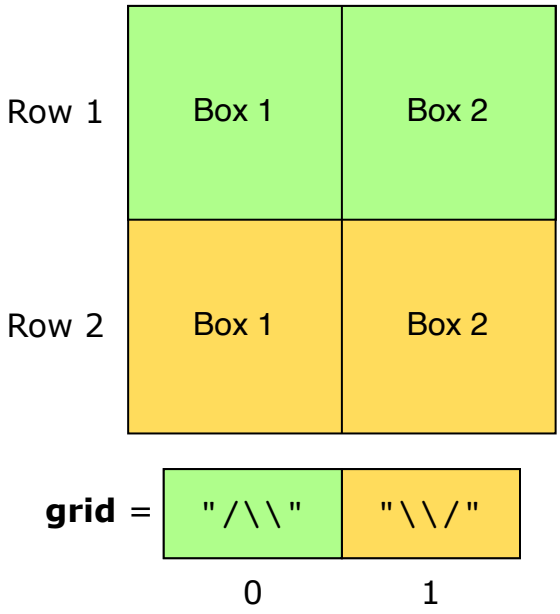


grid = | "/\\" | "\\/" |
|---|---|
| 0 | 1 |

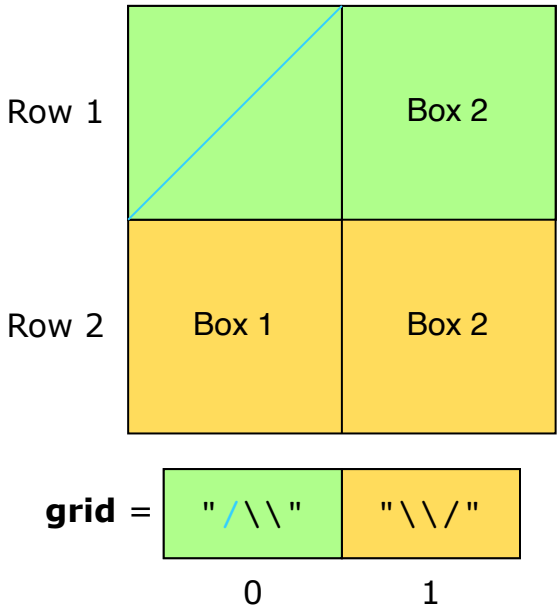
Let's start to fill the grid.



We are at row 1.



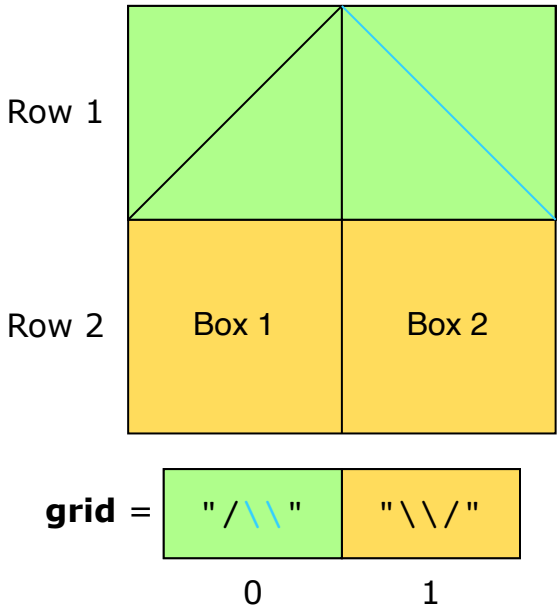
Row 1, box 1 contains a "/", so we fill this box with the "/" character.



grid = `"/\\"` `"\\/"`

0 1

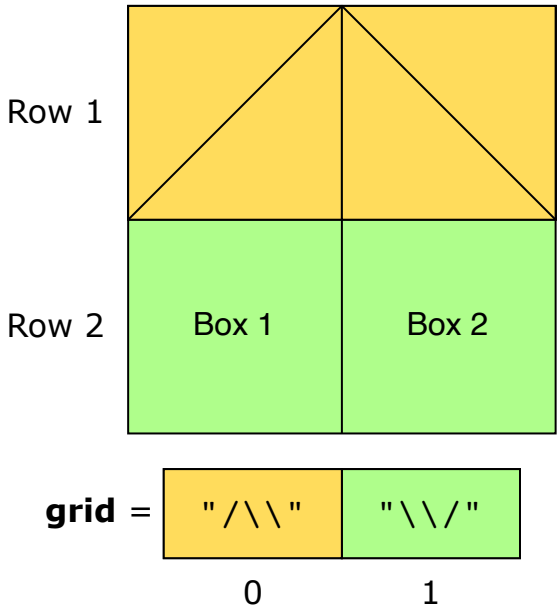
Row 1, box 2 contains a "\", so we fill this box with the "\" character.



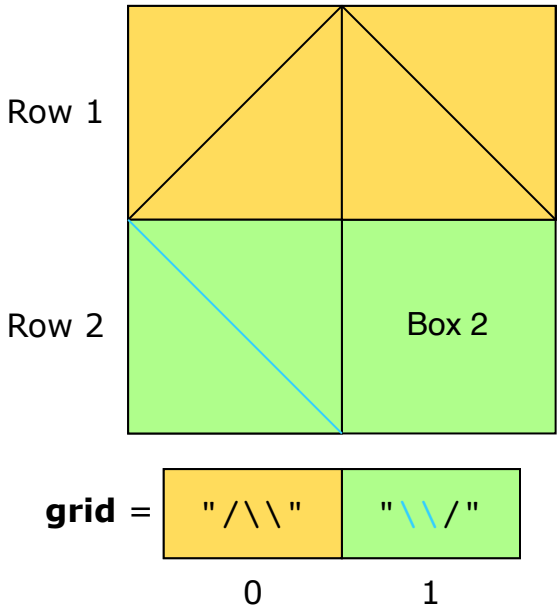
grid = `"/\\"` `"\\/"`

0 1

We are now at row 2.



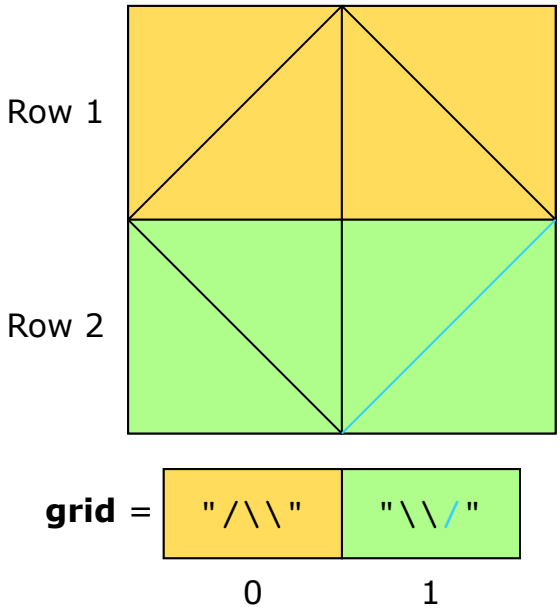
Row 2, box 1 contains a "\", so we fill this box with the "\" character.



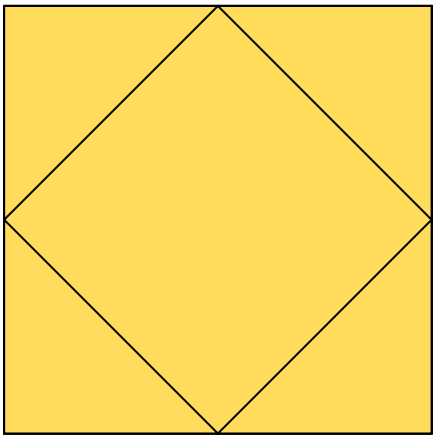
grid = `"/\\"` `"\\/"`

0 1

Row 2, box 2 contains a "/", so we fill this box with the "/" character.

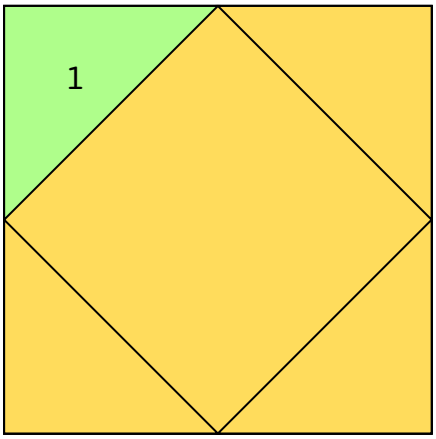


Let's label the regions.

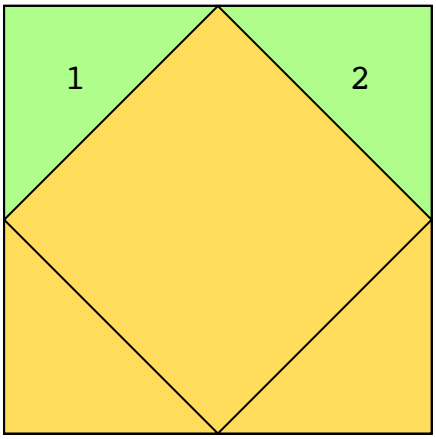


grid =

"/\\"	"\\/"

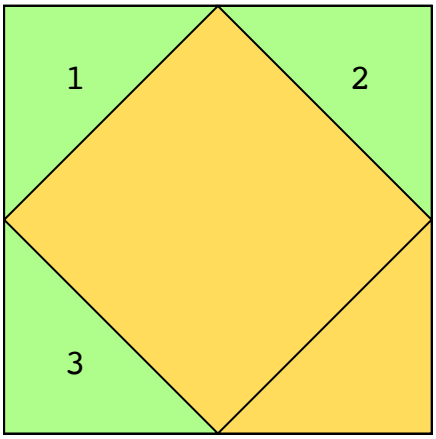


grid = | "/\\" | "\\/" |

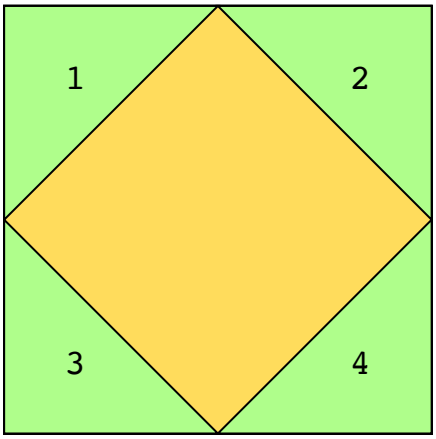


grid =

"/\\"	"\\/"



grid = | "/\\" | "\\/" |



grid = | "/\\" | "\\/" |

There are five regions separated by slashes in this grid.