

5744. Rotating the Box

My Submissions (/contest/biweekly-contest-52/problems/rotating-the-box/submissions/)

Back to Contest (/contest/biweekly-contest-52/)

You are given an  $m \times n$  matrix of characters `box` representing a side-view of a box. Each cell of the box is one of the following:

- A stone `'#'`
- A stationary obstacle `'*'`
- Empty `'.'`

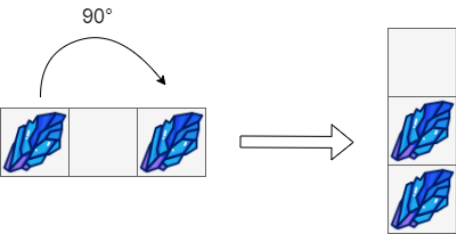
The box is rotated **90 degrees clockwise**, causing some of the stones to fall due to gravity. Each stone falls down until it lands on an obstacle, another stone, or the bottom of the box. Gravity **does not** affect the obstacles' positions, and the inertia from the box's rotation **does not** affect the stones' horizontal positions.

It is **guaranteed** that each stone in `box` rests on an obstacle, another stone, or the bottom of the box.

Return an  $n \times m$  matrix representing the box after the rotation described above.

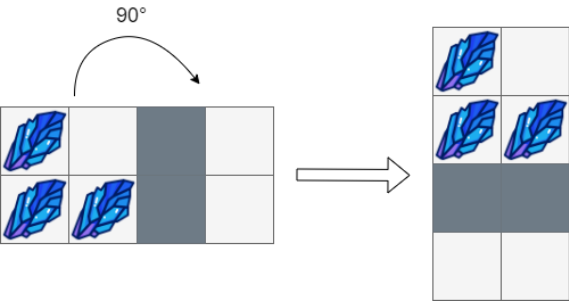
|                    |        |
|--------------------|--------|
| User Accepted:     | 0      |
| User Tried:        | 0      |
| Total Accepted:    | 0      |
| Total Submissions: | 0      |
| Difficulty:        | Medium |

Example 1:



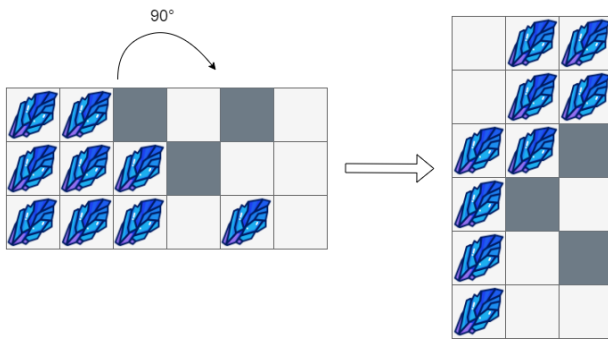
```
Input: box = [["#", ".", "#"]]
Output: [
  [".",
    "#",
    "#"]]
```

Example 2:



```
Input: box = [
  ["#", ".", "*", "."],
  ["#", "#", "*", "."]]
Output: [
  ["#", "."],
  ["#", "#"],
  ["*", "*"],
  [".", "."]]
```

Example 3:



**Input:** box = [
 ["#", "#", "\*", ".", "\*", "."],
 ["#", "#", "#", "\*", " ", "."],
 ["#", "#", "#", ".", "#", "."]]

**Output:** [
 [".", "#", "#"],
 [".", "#", "#"],
 ["#", "#", "\*"],
 ["#", "\*", "."],
 ["#", ".", "\*"],
 ["#", ".", "."]]

#### Constraints:

- m == box.length
- n == box[i].length
- 1 <= m, n <= 500
- box[i][j] is either '#', '\*', or '.'.

JavaScript



```

1 const rotateTheBox = (b) => {
2   b = rotate90Clock(b);
3   let n = b.length;
4   let m = b[0].length;
5   for (let i = n - 1; ~i; i--) {
6     for (let j = 0; j < m; j++) {
7       for (let d = i; d + 1 < n && b[d][j] == '#' && b[d + 1][j] == '.'; d++) {
8         [b[d][j], b[d + 1][j]] = [b[d + 1][j], b[d][j]];
9       }
10    }
11  }
12  return b;
13 };
14
15 const rotate90Clock = (matrix) => {
16   let n = matrix.length;
17   let m = matrix[0].length;
18   let res = initialize2DArrayNew(m, n);
19   for (let i = 0; i < n; i++) {
20     for (let j = 0; j < m; j++) {
21       res[j][n - i - 1] = matrix[i][j];
22     }
23   }
24   return res;
25 };
26
27 const initialize2DArrayNew = (m, n) => {
28   let data = [];
29   for (let i = 0; i < m; i++) {
30     let tmp = new Array(n).fill(0);
31     data.push(tmp);
32   }
33 }

```


```
32     }  
33     return data;  
34 };
```

☐ Custom Testcase[Use Example Testcases](#)[Run](#)[Submit](#)**Submission Result: Accepted** (</submissions/detail/493549890/>) ?[More Details > \(/submissions/detail/493549890/\)](/submissions/detail/493549890/)

Share your acceptance!

[1](#)

Copyright © 2021 LeetCode

[Help Center \(/support/\)](/support/) | [Jobs \(/jobs/\)](/jobs/) | [Bug Bounty \(/bugbounty/\)](/bugbounty/) | [Students \(/student/\)](/student/) | [Terms \(/terms/\)](/terms/) | [Privacy Policy \(/privacy/\)](/privacy/) [United States \(/region/\)](/region/)