1730. Shortest Path to Get Food

Medium 279 11 Add to List Share

You are starving and you want to eat food as quickly as possible. You want to find the shortest path to arrive at any food cell.

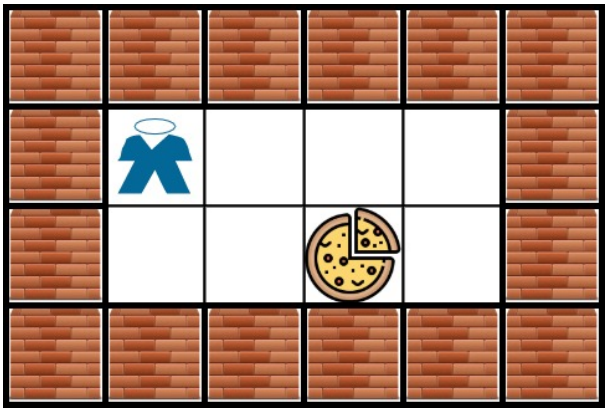
You are given an `m x n` character matrix, `grid`, of these different types of cells:

- `'*'` is your location. There is **exactly one** `'*'` cell.
- `'#'` is a food cell. There may be **multiple** food cells.
- `'O'` is free space, and you can travel through these cells.
- `'X'` is an obstacle, and you cannot travel through these cells.

You can travel to any adjacent cell north, east, south, or west of your current location if there is not an obstacle.

Return *the **length** of the shortest path for you to reach **any** food cell*. If there is no path for you to reach food, return `-1`.

Example 1:



```
Input: grid = [["X","X","X","X","X","X"],["X","*","O","O","O","X"],
["X","O","O","#","O","X"],["X","X","X","X","X","X"]]
Output: 3
Explanation: It takes 3 steps to reach the food.
```

Example 2:

```
in directions)
                {
                    var
item.Item1 + dir[0];
                    var
item.Item2 + dir[1];

                    if(
&& x < m && y >= 0 && y
                    {

                    }
                    if(
[y] == '#')
                    {


 foodFound = true;

                    }
                    els
if(grid[x][y] == 'O')
                    {

 q.Enqueue((x,y));

[y] = 'v';

                    }
                }
                if(food
                {
                    bre
                }
            }
            if(foodFoun
```

Testcase Run Code Result

Accepted Runtime: 102 ms

Your input [["X","X","X","X
["X","*","O","O",

Output 3

Expected 3

Console Use Example Testcase