

75 Days of Code Day 24 Problem no:2390. Removing Stars From a String (leetcode)

Type : Stacks

You are given a string s, which contains stars *.

In one operation, you can:

Choose a star in s.

Remove the closest non-star character to its left, as well as remove the star itself.

Return the string after all stars have been removed.

Note:

The input will be generated such that the operation is always possible.

It can be shown that the resulting string will always be unique.

Example 1:

Input: s = "leetcod*e"**

Output: "lecoe"

Explanation: Performing the removals from left to right:

- The closest character to the 1st star is 't' in "leetcod*e". s becomes "lee*cod*e".**

- The closest character to the 2nd star is 'e' in "lee*cod*e". s becomes "lecod*e".

- The closest character to the 3rd star is 'd' in "lecod*e". s becomes "lecoe".

There are no more stars, so we return "lecoe".

Example 2:

Shubham Agrahari

Input: s = "erase*****"

Output: ""

Explanation: The entire string is removed, so we return an empty string

Solution using Stack

- Simply following LIFO (last in first out) if it finds * it pos the previous element else push new element

```
28 // Output:
29 // Explanation: The entire string is removed, so we return an empty string
30
31 function removeStars(s: string): string {
32     let chars = s.split("");
33     let ansarr = [];
34     for (let index = 0; index < chars.length; index++) {
35         if (chars[index] === "*" && ansarr.length > 0) {
36             ansarr.pop();
37         } else if (chars[index] !== "*") {
38             ansarr.push(chars[index]);
39         }
40     }
41
42     return ansarr.join("");
43 }
44
45 let ans = removeStars("leet**cod*e");
46 console.log("Answer :", ans);
47
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL GITLENS

[Running] node "c:\Users\Shubham\Desktop\75daysOfCode\75DaysOfCode\day24.js"

Answer : lecoe

