5483. Make The String Great

My Submissions (/contest/weekly-contest-201/problems/make-the-string-great/submissions/)

Back to Contest (/contest/weekly-contest-201/)

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Easy

Given a string `s` of lower and upper case English letters.

A good string is a string which doesn't have **two adjacent characters** `s[i]` and `s[i + 1]` where:

- `0 <= i <= s.length - 2`
- `s[i]` is a lower-case letter and `s[i + 1]` is the same letter but in upper-case or **vice-versa**.

To make the string good, you can choose **two adjacent** characters that make the string bad and remove them. You can keep doing this until the string becomes good.

Return *the string* after making it good. The answer is guaranteed to be unique under the given constraints.

Notice that an empty string is also good.

Example 1:

```
Input: s = "leEeetcode"
Output: "leetcode"
Explanation: In the first step, either you choose i = 1 or i = 2, both will result "leEeet
```

Example 2:

```
Input: s = "abBAcC"
Output: ""
Explanation: We have many possible scenarios, and all lead to the same answer. For example
"abBAcC" --> "aAcC" --> "cC" --> ""
"abBAcC" --> "abBA" --> "aA" --> ""
```

Example 3:

```
Input: s = "s"
Output: "s"
```

Constraints:

- `1 <= s.length <= 100`
- `s` contains only lower and upper case English letters.

JavaScript

```javascript
/**
 * @param {string} s
 * @return {string}
 */
const makeGood = function (s) {
    if (s.length == 1) return s;
    let stack = [s[0]];
    for (let i = 1; i < s.length; i++) {
        if (stack.length == 0){
            stack.push(s[i]);
        }  else {
            let end = stack[stack.length - 1];
            if ((isUpperCase(end) && isLowerCase(s[i]) && isEqual(end, s[i]))
|| (isLowerCase(end) && isUpperCase(s[i]) && isEqual(end, s[i]))) {
                stack.pop();
                continue;
            }
            stack.push(s[i]);
        }
    }
    return stack.join("");
};


const isEqual = (s1, s2) => {
    if (s1.toLowerCase() == s2.toLowerCase()) {
        return true;
    }
    return false;
};

const isUpperCase = (character) => {
    if (character == character.toUpperCase()) {
        return true;
    }
    return false;
};

const isLowerCase = (character) => {
    if (character == character.toLowerCase()) {
        return true;
    }
    return false;
};
```

☐ **Custom Testcase** Use Example Testcases

Run Submit

Submission Result: Accepted (/submissions/detail/378183681/)

More Details > (/submissions/detail/378183681/)

Share your acceptance!

1