5509. Minimum Deletion Cost to Avoid Repeating Letters

missions (/contest/weekly-contest-205/problems/minimum-deletion-cost-to-avoid-repeating-letters/submissions/)

Contest (/contest/weekly-contest-205/)

Given a string `s` and an array of integers `cost` where `cost[i]` is the cost of deleting the character `i` in `s`.

Return the minimum cost of deletions such that there are no two identical letters next to each other.

Notice that you will delete the chosen characters at the same time, in other words, after deleting a character, the costs of deleting other characters will not change.

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

Example 1:

```
Input: s = "abaac", cost = [1,2,3,4,5]
Output: 3
Explanation: Delete the letter "a" with cost 3 to get "abac" (String without two identical
```

Example 2:

```
Input: s = "abc", cost = [1,2,3]
Output: 0
Explanation: You don't need to delete any character because there are no identical letters
```

Example 3:

```
Input: s = "aabaa", cost = [1,2,3,4,1]
Output: 2
Explanation: Delete the first and the last character, getting the string ("aba").
```

Constraints:

- `s.length == cost.length`
- `1 <= s.length, cost.length <= 10^5`
- `1 <= cost[i] <= 10^4`
- `s` contains only lowercase English letters.

JavaScript

```
/**
 * @param {string} s
 * @param {number[]} cost
 * @return {number}
 */
var minCost = function(s, cost) {

};
```

Custom Testcase Use Example Testcases

Run Submit