

Problem 1332: Remove Palindromic Subsequences

Problem Information

Difficulty: Easy

Acceptance Rate: 76.85%

Paid Only: No

Tags: Two Pointers, String

Problem Description

You are given a string `s` consisting **only** of letters `'a` and `'b` . In a single step you can remove one **palindromic subsequence** from `s` .

Return _the**minimum** number of steps to make the given string empty_.

A string is a **subsequence** of a given string if it is generated by deleting some characters of a given string without changing its order. Note that a subsequence does **not** necessarily need to be contiguous.

A string is called **palindrome** if is one that reads the same backward as well as forward.

Example 1:

Input: s = "ababa" **Output:** 1 **Explanation:** s is already a palindrome, so its entirety can be removed in a single step.

Example 2:

Input: s = "abb" **Output:** 2 **Explanation:** "_a_ bb" -> "_bb_" -> "". Remove palindromic subsequence "a" then "bb".

Example 3:

****Input:**** s = "baabb" ****Output:**** 2 ****Explanation:**** _baa_ b _b_ -> _b_ -> "". Remove palindromic subsequence "baab" then "b".

****Constraints:****

* `1 <= s.length <= 1000` * `s[i]` is either `'a'` or `'b'`.

Code Snippets

C++:

```
class Solution {
public:
    int removePalindromeSub(string s) {
        }
};
```

Java:

```
class Solution {
    public int removePalindromeSub(String s) {
        }
}
```

Python3:

```
class Solution:
    def removePalindromeSub(self, s: str) -> int:
```