

Problem 2730: Find the Longest Semi-Repetitive Substring

Problem Information

Difficulty: Medium

Acceptance Rate: 38.06%

Paid Only: No

Tags: String, Sliding Window

Problem Description

You are given a digit string `s` that consists of digits from 0 to 9.

A string is called **semi-repetitive** if there is **at most** one adjacent pair of the same digit. For example, `"0010"`, `"002020"`, `"0123"`, `"2002"`, and `"54944"` are semi-repetitive while the following are not: `"00101022"` (adjacent same digit pairs are 00 and 22), and `"1101234883"` (adjacent same digit pairs are 11 and 88).

Return the length of the **longest semi-repetitive substring** of `s`.

Example 1.

Input: `s = "52233"`

Output: 4

Explanation:

The longest semi-repetitive substring is "5223". Picking the whole string "52233" has two adjacent same digit pairs 22 and 33, but at most one is allowed.

Example 2.

Input: `s = "5494"`

****Output:**** 4

****Explanation:****

`s` is a semi-repetitive string.

****Example 3:****

****Input:**** s = "1111111"

****Output:**** 2

****Explanation:****

The longest semi-repetitive substring is "11". Picking the substring "111" has two adjacent same digit pairs, but at most one is allowed.

****Constraints:****

* `1` <= s.length <= 50 * ``0' <= s[i] <= '9`

Code Snippets

C++:

```
class Solution {
public:
    int longestSemiRepetitiveSubstring(string s) {

    }

};
```

Java:

```
class Solution {
    public int longestSemiRepetitiveSubstring(String s) {

    }

}
```

Python3:

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