Problem 3456: Find Special Substring of Length K

Problem Information

Difficulty: Easy
Acceptance Rate: 35.06%
Paid Only: No
Tags: String

Problem Description

You are given a string `s` and an integer `k`.

Determine if there exists a substring of length **exactly** `k` in `s` that satisfies the following conditions:

1. The substring consists of **only one distinct character** (e.g., `"aaa"` or `"bbb"`). 2. If there is a character **immediately before** the substring, it must be different from the character in the substring. 3. If there is a character **immediately after** the substring, it must also be different from the character in the substring.

Return `true` if such a substring exists. Otherwise, return `false`.

Example 1:

Input: s = "aaabaaa", k = 3

Output: true

Explanation:

The substring `s[4..6] == "aaa"` satisfies the conditions.

* It has a length of 3. * All characters are the same. * The character before `"aaa"` is `'b'`, which is different from `'a'`. * There is no character after `"aaa"`.

Example 2:

Input: s = "abc", k = 2

Output: false

Explanation:

There is no substring of length 2 that consists of one distinct character and satisfies the conditions.

Constraints:

* `1 <= k <= s.length <= 100` * `s` consists of lowercase English letters only.

Code Snippets

C++:

```cpp
class Solution {
public:
bool hasSpecialSubstring(string s, int k) {


}
};
```

Java:

```java
class Solution {
public boolean hasSpecialSubstring(String s, int k) {


}
}
```

Python3:

```python
class Solution:
def hasSpecialSubstring(self, s: str, k: int) -> bool:
```