Problem 3325: Count Substrings With K-Frequency Characters I

Problem Information

Difficulty: Medium
Acceptance Rate: 55.19%
Paid Only: No
Tags: Hash Table, String, Sliding Window

Problem Description

Given a string `s` and an integer `k`, return the total number of substrings of `s` where **at least one** character appears **at least** `k` times.

Example 1:

Input: s = "abacb", k = 2

Output: 4

Explanation:

The valid substrings are:

* `"aba"` (character `'a'` appears 2 times). * `"abac"` (character `'a'` appears 2 times). * `"abacb"` (character `'a'` appears 2 times). * `"bacb"` (character `'b'` appears 2 times).

Example 2:

Input: s = "abcde", k = 1

Output: 15

Explanation:

All substrings are valid because every character appears at least once.

Constraints:

* `1 <= s.length <= 3000` * `1 <= k <= s.length` * `s` consists only of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
int numberOfSubstrings(string s, int k) {

}
};
```

Java:

```
class Solution {
public int numberOfSubstrings(String s, int k) {

}
}
```

Python3:

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