Problem 3333: Find the Original Typed String II

Problem Information

Difficulty: Hard
Acceptance Rate: 45.73%
Paid Only: No
Tags: String, Dynamic Programming, Prefix Sum

Problem Description

Alice is attempting to type a specific string on her computer. However, she tends to be clumsy and **may** press a key for too long, resulting in a character being typed **multiple** times.

You are given a string `word`, which represents the **final** output displayed on Alice's screen. You are also given a **positive** integer `k`.

Return the total number of _possible_ original strings that Alice _might_ have intended to type, if she was trying to type a string of size **at least** `k`.

Since the answer may be very large, return it **modulo** `109 + 7`.

Example 1:

Input: word = "aabbccdd", k = 7

Output: 5

Explanation:

The possible strings are: `"aabbccdd"`, `"aabbccd"`, `"aabbcdd"`, `"aabccdd"`, and `"abbccdd"`.

Example 2:

Input: word = "aabbccdd", k = 8

Output: 1

Explanation:

The only possible string is `"aabbccdd"`.

Example 3:

Input: word = "aaabbb", k = 3

Output: 8

Constraints:

* `1 <= word.length <= 5 * 105` * `word` consists only of lowercase English letters. * `1 <= k <= 2000`

Code Snippets

C++:

```
class Solution {
public:
int possibleStringCount(string word, int k) {

}
};
```

Java:

```
class Solution {
public int possibleStringCount(String word, int k) {

}
}
```

Python3:

```
class Solution:
def possibleStringCount(self, word: str, k: int) -> int:
```