Problem 2272: Substring With Largest Variance

Problem Information

Difficulty: Hard
Acceptance Rate: 45.97%
Paid Only: No
Tags: Array, Dynamic Programming

Problem Description

The **variance** of a string is defined as the largest difference between the number of occurrences of **any** `2` characters present in the string. Note the two characters may or may not be the same.

Given a string `s` consisting of lowercase English letters only, return _the**largest variance** possible among all **substrings** of_ `s`.

A **substring** is a contiguous sequence of characters within a string.

Example 1:

Input: s = "aababbb" **Output:** 3 **Explanation:** All possible variances along with their respective substrings are listed below: - Variance 0 for substrings "a", "aa", "ab", "abab", "aababb", "ba", "b", "bb", and "bbb". - Variance 1 for substrings "aab", "aba", "abb", "aabab", "ababb", "aababbb", and "bab". - Variance 2 for substrings "aaba", "ababbb", "abbb", and "babb". - Variance 3 for substring "babbb". Since the largest possible variance is 3, we return it.

Example 2:

Input: s = "abcde" **Output:** 0 **Explanation:** No letter occurs more than once in s, so the variance of every substring is 0.

Constraints:

* `1 <= s.length <= 104` * `s` consists of lowercase English letters.

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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