

Problem 1062: Longest Repeating Substring

Problem Information

Difficulty: Medium

Acceptance Rate: 63.32%

Paid Only: Yes

Tags: String, Binary Search, Dynamic Programming, Rolling Hash, Suffix Array, Hash Function

Problem Description

Given a string `s`, return _the length of the longest repeating substrings_. If no repeating substring exists, return `0`.

Example 1:

Input: s = "abcd" **Output:** 0 **Explanation:** There is no repeating substring.

Example 2:

Input: s = "abbaba" **Output:** 2 **Explanation:** The longest repeating substrings are "ab" and "ba", each of which occurs twice.

Example 3:

Input: s = "aabcaabdaab" **Output:** 3 **Explanation:** The longest repeating substring is "aab", which occurs 3 times.

Constraints:

* `1 <= s.length <= 2000` * `s` consists of lowercase English letters.

Code Snippets

C++:

```
class Solution {  
public:  
    int longestRepeatingSubstring(string s) {  
  
    }  
};
```

Java:

```
class Solution {  
public int longestRepeatingSubstring(String s) {  
  
}  
}
```

Python3:

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