

Problem 1624: Largest Substring Between Two Equal Characters

Problem Information

Difficulty: Easy

Acceptance Rate: 68.27%

Paid Only: No

Tags: Hash Table, String

Problem Description

Given a string `s`, return _the length of the longest substring between two equal characters, excluding the two characters._ If there is no such substring return `-1`.

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

Example 1:

Input: s = "aa" **Output:** 0 **Explanation:** The optimal substring here is an empty substring between the two 'a's.

Example 2:

Input: s = "abca" **Output:** 2 **Explanation:** The optimal substring here is "bc".

Example 3:

Input: s = "cbzxy" **Output:** -1 **Explanation:** There are no characters that appear twice in s.

Constraints:

* `1 <= s.length <= 300` * `s` contains only lowercase English letters.

Code Snippets

C++:

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

Java:

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

Python3:

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