

Problem 387: First Unique Character in a String

Problem Information

Difficulty: Easy

Acceptance Rate: 64.47%

Paid Only: No

Tags: Hash Table, String, Queue, Counting

Problem Description

Given a string `s`, find the **first** non-repeating character in it and return its index. If it **does not** exist, return `-1`.

Example 1:

Input: `s = "leetcode"`

Output: `0`

Explanation:

The character `'l'` at index 0 is the first character that does not occur at any other index.

Example 2:

Input: `s = "loveleetcode"`

Output: `2`

Example 3:

Input: `s = "aabb"`

Output: `-1`

****Constraints:****

*`1` <= s.length <= 105` *`s` consists of only lowercase English letters.

Code Snippets

C++:

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

Java:

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

Python3:

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