

[387 First Unique Character in a String \(link\)](#)

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: $s = \text{"leetcode"}$
Output: 0

Example 2:

Input: $s = \text{"loveleetcode"}$
Output: 2

Example 3:

Input: $s = \text{"aabb"}$
Output: -1

Constraints:

- $1 \leq s.length \leq 10^5$
- s consists of only lowercase English letters.

(scroll down for solution)

Solution

Language: *cpp*

Status: Accepted

```
#include <string>
#include <unordered_map>

using namespace std;

class Solution {
public:
    int firstUniqChar(string s) {
        unordered_map<char, int> frequency;

        // Подсчитываем частоту каждого символа
        for (char ch : s) {
            frequency[ch]++;
        }

        // Ищем первый неповторяющийся символ
        for (int i = 0; i < s.length(); ++i) {
            if (frequency[s[i]] == 1) {
                return i;
            }
        }

        return -1; // Если неповторяющийся символ не найден
    }
};
```