

Problem 2423: Remove Letter To Equalize Frequency

Problem Information

Difficulty: Easy

Acceptance Rate: 18.80%

Paid Only: No

Tags: Hash Table, String, Counting

Problem Description

You are given a **0-indexed** string `word`, consisting of lowercase English letters. You need to select **one** index and **remove** the letter at that index from `word` so that the **frequency** of every letter present in `word` is equal.

Return `true` if it is possible to remove one letter so that the frequency of all letters in `word` are equal, and `false` otherwise.

Note:

The **frequency** of a letter `x` is the number of times it occurs in the string. You **must** remove exactly one letter and cannot choose to do nothing.

Example 1:

Input: `word = "abcc"` **Output:** `true` **Explanation:** Select index 3 and delete it: `word` becomes `"abc"` and each character has a frequency of 1.

Example 2:

Input: `word = "aazz"` **Output:** `false` **Explanation:** We must delete a character, so either the frequency of `"a"` is 1 and the frequency of `"z"` is 2, or vice versa. It is impossible to make all present letters have equal frequency.

Constraints:

* `2 <= word.length <= 100` * `word` consists of lowercase English letters only.

Code Snippets

C++:

```
class Solution {  
public:  
    bool equalFrequency(string word) {  
  
    }  
};
```

Java:

```
class Solution {  
    public boolean equalFrequency(String word) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def equalFrequency(self, word: str) -> bool:
```