# 题目

给你一个下标从0开始的字符串word，字符串只包含小写英文字母。你需要选择一个下标并删除下标处的字符，使得word中剩余每个字母出现频率相同。

如果删除一个字母后，word中剩余所有字母的出现频率都相同，那么返回true，否则返回false。

注意：

字母x的频率是这个字母在字符串中出现的次数。

你必须恰好删除一个字母，不能一个字母都不删除。

示例 1：

输入：word = "abcc"

输出：true

解释：选择下标 3 并删除该字母：word 变成 "abc" 且每个字母出现频率都为 1 。

示例 2：

输入：word = "aazz"

输出：false

解释：我们必须删除一个字母，所以要么 "a" 的频率变为 1 且 "z" 的频率为 2 ，要么两个字母频率反过来。所以不可能让剩余所有字母出现频率相同。

提示：

2 <= word.length <= 100

word 只包含小写英文字母。

# 分析

## 方法一：枚举

思路：

首先遍历输入字符串word，统计每一个字符出现的频率。然后我们按照字母序，遍历所有的字符。如果当前这个字符出现在原字符串中，我们假定要删除这个字符，把这个字符出现的频率减一，统计所有出现字符的频率集合。如果集合大小为1，则说明剩余每个字母出现频率相同，我们直接返回true，反之说明删除当前字符不可行，我们把这个字符的频率加一进行还原。

最后，当我们尝试过所有不同字符后，还没有找到能删除的字符，使得满足要求，我们返回false。

代码：

class Solution {

public:

bool equalFrequency(string word) {

int charCount[26] = {0};

for (char& c : word) {

charCount[c - 'a']++;

}

for (int i = 0; i < 26; i++) {

if (charCount[i] == 0) {

continue;

}

charCount[i]--;

unordered\_set<int> frequency;

for (int f : charCount) {

if (f > 0) {

frequency.insert(f);

}

}

if (frequency.size() == 1) {

return true;

}

charCount[i]++;

}

return false;

}

};