Check if frequencies can be equal

Check if frequencies can be equal

Medium Accuracy: 16.67% Submissions: 75K+ Points: 4

Internship Alert!
New month-> Fresh Chance to top the leaderboard and get SDE Internship!

Given a string **s** which contains only lower alphabetic characters, check if it is possible to remove **at most one** character from this string in such a way that frequency of each distinct character becomes same in the string. Return **true** if it is possible to do else return **false**.

Note: The driver code print 1 if the value returned is true, otherwise 0.

Example 1:

Input:
s = "xyyz"
Output:
1
Explanation:
Removing one 'y' will make frequency of each character to be 1.

Example 2:

Input:
s = "xxxxyyzz"
Output:
0
Explanation:
Frequency can not be made same by removing at most one character.

→ Approach:-

x x x x y y z z

Step 1:-

↓
freq. array.

Step 2: Now iterate through each character if it's count is greater than 0 & subtract 1 from its freq., If after subtraction all the freq. are same that means it can be done.

Time Complexity: O(N)
Space Complexity: O(1)

```
class Solution{
public:
        int getIdx(char ch)
        {
            return (ch - 'a');
        }
        bool allSame(int freq[], int N)
        {
            int same;
            int i;
            for (i = 0; i < N; i++)
            {
                if (freq[i] > 0)
                {
                    same = freq[i];
                    break;
                }
            }
            for (int j = i+1; j < N; j++)
                if (freq[j] > 0 && freq[j] != same)
                    return false;

            return true;
        }
        bool sameFreq(string str)
        {
            int M = 26;
            int l = str.length();

            //  fill frequency array
            int freq[M] = {0};
            for (int i = 0; i < l; i++)
                freq[getIdx(str[i])]++;
            if (allSame(freq, M))
                return true;
            for (char c = 'a'; c <= 'z'; c++)
            {
                int i = getIdx(c);
                if (freq[i] > 0)
                {
                    freq[i]--;
                    if (allSame(freq, M))
                        return true;
                    freq[i]++;
                }
            }
            return false;
        }
};
```