Sherlock and Valid String

by darkshadows

Problem | Submissions | Leaderboard | Discussions | Editorial

You know my powers, my dear Watson, and yet at the end of three months I was forced to confess that I had at last met an antagonist who was my intellectual equal.

A "valid" string is a string S such that for all distinct characters in S each such character occurs the same number of times in S.

For example, `aabb` is a valid string because the frequency of both characters `a` and `b` is 2, whereas `aabbc` is not a valid string because the frequency of characters `a`, `b`, and `c` is not the same.

Watson gives a string S to Sherlock and asks him to remove some characters from the string such that the new string is a "valid" string.

Sherlock wants to know from you if it's possible to be done with less than or equal to one removal.

Input Format

The first and only line contains S, the string Watson gives to Sherlock.

Constraints

- $1 \leq |S| \leq 10^5$
- String S contains lowercase letters only ($a - z$).

Output Format

Output `YES` if string S can be converted to a "valid" string by removing less than or equal to one character.
Else, output `NO`.

Sample Input

```
aabbcd
```

Sample Output

```
NO
```

Explanation

2 is the minimum number of removals required to make it a valid string. It can be done in following two ways:

Remove `c` and `d` to get `aabb`.
Or remove `a` and `b` to get `abcd`.

Submissions: 7872
Max Score: 100
Difficulty: Hard

Rate This Challenge:

More

Current Buffer (saved locally, editable) C#

```
1 using System;
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
class Solution {
    static void Main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution */


        string s = Console.ReadLine();



            Dictionary<char, int> frec = s.ToCharArray().GroupBy(x => x)
                                .ToDictionary(x => x.Key, x => x.Count());

            Dictionary<int, int> values = new Dictionary<int, int>();

            foreach (KeyValuePair<char, int> kvp in frec)
            {
                //Console.WriteLine(kvp.Key + " " + kvp.Value);
                if (values.ContainsKey(kvp.Value))
                {
                    values[kvp.Value]++;
                }
                else
                {
                    values[kvp.Value] = 1;
                }
            }


            if (values.Count == 1)
            {
                Console.WriteLine("YES");
            }
            else if (values.Count > 2)
            {
                Console.WriteLine("NO");
            }
            else if(values.Count==2)
            {
                //foreach (KeyValuePair<int, int> kvp in values)
                //{
                //    Console.WriteLine(kvp.Key + " " + kvp.Value);
                //}
                List<int> lista = values.Values.ToList();
                //foreach (int v in lista)
                //{
                //    Console.Write(v + " ");
                //}
                if (lista[0] != 1 && lista[1] != 1)
                {
                    Console.WriteLine("NO");
                }
                else
                {
                    Console.WriteLine("YES");
                }

            }



    }
}
```

Line: 12 Col: 12

Upload Code as File Test against custom input Run Code Submit Code

Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5

- Test Case #3
- Test Case #4
- Test Case #5
- Test Case #6
- Test Case #7
- Test Case #8
- Test Case #9
- Test Case #10
- Test Case #11
- Test Case #12
- Test Case #13
- Test Case #14
- Test Case #15

Next Challenge