

Sherlock and Valid String



blem 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 $\boldsymbol{\mathcal{S}}$ 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

- $1 \le |S| \le 10^5$
- String \boldsymbol{S} contains lowercase letters only $(\boldsymbol{a}-\boldsymbol{z})$.

Output Format

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

Sample Input

aabbcd

Sample Output

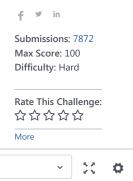
NO

Explanation

 ${f 2}$ 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.



```
justing system,
 2
    using System.Collections.Generic;
 3
    using System.Linq;
 4
   using System.Text;
 5 ▼ class Solution {
 6
        static void Main(String[] args) {
            /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be
 7
    named Solution */
 8
 9
10
             string s = Console.ReadLine();
11
12
13
                 Dictionary<char, int> frec = s.ToCharArray().GroupBy(x \Rightarrow x)
14
15
                                      .ToDictionary(x => x.Key, x => x.Count());
16
                 Dictionary<int, int> values = new Dictionary<int, int>();
17
18
                 foreach (KeyValuePair<char, int> kvp in frec)
19
20 1
21
                     //Console.WriteLine(kvp.Key + " " + kvp.Value);
                                                                                                                   22
                     if (values.ContainsKey(kvp.Value))
23
24
                         values[kvp.Value]++;
25
26
                     else
27 ▼
28
                         values[kvp.Value] = 1;
29
30
                 }
31
32
33
                 if (values.Count == 1)
34
35
                     Console.WriteLine("YES");
36
37
                 else if (values.Count > 2)
38
39
                     Console.WriteLine("NO");
40
41
                 else if(values.Count==2)
42
43
                     //foreach (KeyValuePair<int, int> kvp in values)
44
                     //{
                           Console.WriteLine(kvp.Key + " " + kvp.Value);
45
46
                     //}
47
                     List<int> lista = values.Values.ToList();
48
                     //foreach (int v in lista)
49
                     //{
                           Console.Write(v + " ");
50
                     //
51
52
                     if (lista[0] != 1 && lista[1] != 1)
53 •
54
                         Console.WriteLine("NO");
55
56
                     else
57
                         Console.WriteLine("YES");
58
59
60
61
62
63
64
65
        }
66
                                                                                                      Line: 12 Col: 12
```

<u>**1**</u> <u>Upload Code as File</u> □ Test against custom input

Run Code

Submit Code

```
Congrats, you solved this challenge!

✓ Test Case #0
✓ Test Case #1
✓ Test Case #2
✓ Test Case #4
```

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature