Problem: Sherlock and the valid String

Sherlock considers a string, , to be *valid* if either of the following conditions are satisfied:

1. All characters in  have the same exact frequency (i.e., occur the same number of times). For example,  is valid, but  is not valid.
2. Deleting exactly  character from  will result in all its characters having the same frequency. For example,  and  are valid because all their letters will have the same frequency if we remove  occurrence of c, but  is not valid because we'd need to remove  characters.

Given , can you determine if it's valid or not? If it's valid, print YES on a new line; otherwise, print NO instead.

**Input Format**

A single string denoting .

**Constraints**

* String  consists of lowercase letters only (i.e., [a-z]).

**Output Format**

Print YES if string  is valid; otherwise, print NO instead.

**Sample Input 0**

aabbcd

**Sample Output 0**

NO

**Explanation 0**

We would need to remove two characters from  to make it valid, because a and b both have a frequency of  and c and d both have a frequency of . This means  is *invalid* because we'd need to remove more than  character to make all its letters have the same frequency, so we print NO as our answer.

Solution

int main(){

string str;

int count=0, ping=0, length=0;

int hit[3]={0};

int score[26] = { 0 };

cin>>str;

length = str.length();

for(int i=0; i<length; i++) //creating a dictionary map

{

int temp = (int)str[i]-97;

score[temp]+=1;

}

for(int i=0; i<26; i++) //populating the frequency finder

{

if(!score[i]==0)

{

if(count<3)

{ hit[count]=score[i]; }

count+=1;

}

}

//finding the frequency

if(hit[0] == hit[1])

{ping = hit[0];}

else if(hit[1] == hit[2])

{ping = hit[1];}

else

{ping = hit[2];}

int chance = 1;//making the decision

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

{

if(!score[i]==0)

{

if(score[i]!=ping)

{if(chance==1)

{if(score[i]-1==ping || score[i]-1==0)

{chance-=1;}

else

{cout<<"NO"; return 0;}

}

else

{cout<<"NO"; return 0;}

}

}

}

cout<<"YES";

return 0;

}

`’Anshul AgGarwal