Lapindrome is defined as a string which when split in the middle, gives two halves having the same characters and same frequency of each character. If there are odd number of characters in the string, we ignore the middle character and check for lapindrome. For example gaga is a lapindrome, since the two halves ga and ga have the same characters with same frequency. Also, abccab, rotor and xyzxy are a few examples of lapindromes. Note that abbaab is NOT a lapindrome. The two halves contain the same characters but their frequencies do not match.
Your task is simple. Given a string, you need to tell if it is a lapindrome.

Input:
First line of input contains a single integer T, the number of test cases.
Each test is a single line containing a string S composed of only lowercase English alphabet.

Output:
For each test case, output on a separate line: "YES" if the string is a lapindrome and "NO" if it is not.

Constraints:
$1 \leq T \leq 100$
$2 \leq |S| \leq 1000$, where $|S|$ denotes the length of S

Example:

Input:

```
6
gaga
abcde
rotor
xyzxy
abbaab
ababc
```

Output:

```
YES
NO
YES
```

```
YES
NO
NO
```

Solution:

```
#include <iostream>
#include <bits/stdc++.h>
using namespace std;

int main() {

        char str[1002];
        int count[26];
        int t;
        cin>>t;

        for(;t>0;t--){
          cin>>str;
          int len = strlen(str);
          memset(count,0,sizeof(count));
          for(int i=0, j=len-1 ; i < j ; i++ , j-- ){
            count[str[i] - 'a']++;
            count[str[j] - 'a']--;
          }

         bool ok = true;

         for(int i =0 ; i < 26; i++){
              if(count[i] != 0 ){
              ok = false;
              break;
             }
         }

       puts(ok ? "YES" : "NO");
      }

     return 0;
```

}