**Anagram**

[string](http://www.practice.geeksforgeeks.org/tag-page.php?tag=string&isCmp=0)

Given two strings, check whether two given strings are anagram of each other or not. An anagram of a string is another string that contains same characters, only the order of characters can be different. For example, “act” and “tac” are anagram of each other.

**Input:**

The first line of input contains an integer T denoting the number of test cases. Each test case consist of two strings in 'lowercase' only, in a separate line.

**Output:**

Print "YES" without quotes if the two strings are anagram else print "NO".

**Constraints:**

1 ≤ T ≤ 30

1 ≤ |s| ≤ 100

**Example:**

Input:  
2  
geeksforgeeks  
forgeeksgeeks  
allergy  
allergic

Output:  
YES  
NO

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=88>

#include <iostream>

#include <stdio.h>

#include <math.h>

#include <map>

//#include <conio.h>

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

std::string P;

cin >> P;

std::string Q;

cin >> Q;

std::string answer = "YES";

if(P.size() != Q.size()) {

answer = "NO";

}

std::map<char, int> mp;

for(int i =0; i < P.size(); i++) {

mp[P[i]]++;

}

std::map<char, int> mq;

for(int i =0; i < Q.size(); i++) {

mq[Q[i]]++;

}

for(std::map<char, int>::iterator it = mp.begin(); it != mp.end(); it++) {

if(mp[it->first] != mq[it->first]) {

answer = "NO";

}

}

cout << answer << endl;

}

//getch();

return 0;

}