**Two Strings**

Given two strings, determine if they share a common substring. A substring may be as small as one character.

For example, the words "a", "and", "art" share the common substring a. The words "be" and "cat" do not share a substring.

**Function Description**

Complete the function *twoStrings* in the editor below. It should return a string, either YES or NO based on whether the strings share a common substring.

twoStrings has the following parameter(s):

* *s1, s2*: two strings to analyze.

**Input Format**

The first line contains a single integer p, the number of test cases.

The following p pairs of lines are as follows:

* The first line contains string s1.
* The second line contains string s2.

**Constraints**

* S1 and s2 consist of characters in the range ascii[a-z].
* 1<= p <=5
* 1<= |s1|,|s2| <=10^5

**Output Format**

For each pair of strings, return YES or NO.

**Sample Input**

2

hello

world

hi

world

**Sample Output**

YES

NO

**Explanation**

We have  pairs to check:

1. A=”hello”, b-“world”. The substrings “o” and “l” are common to both strings.
2. A=”hi”, b=”world”. s1 and s2 share no common substrings.

using System.CodeDom.Compiler;

using System.Collections.Generic;

using System.Collections;

using System.ComponentModel;

using System.Diagnostics.CodeAnalysis;

using System.Globalization;

using System.IO;

using System.Linq;

using System.Reflection;

using System.Runtime.Serialization;

using System.Text.RegularExpressions;

using System.Text;

using System;

class Solution {

// Complete the twoStrings function below.

static string twoStrings(string s1, string s2) {

string temp;

if(s1.Length<s2.Length){

temp=s1;

s1=s2;

s2=temp;

}

Hashtable ht = new Hashtable();

for(int j=0;j<s1.Length;j++)

{

if(!ht.ContainsKey(s1[j]))

ht.Add(s1[j],j);

}

for(int i=0;i<s2.Length;i++){

if(ht.ContainsKey(s2[i]))

return "YES";

}

return "NO";

}

static void Main(string[] args) {

TextWriter textWriter = new StreamWriter(@System.Environment.GetEnvironmentVariable("OUTPUT\_PATH"), true);

int q = Convert.ToInt32(Console.ReadLine());

for (int qItr = 0; qItr < q; qItr++) {

string s1 = Console.ReadLine();

string s2 = Console.ReadLine();

string result = twoStrings(s1, s2);

textWriter.WriteLine(result);

}

textWriter.Flush();

textWriter.Close();

}

}

**Congratulations**

You solved this challenge. Would you like to challenge your friends?

[Next Challenge](https://www.hackerrank.com/challenges/sherlock-and-anagrams?h_l=interview&playlist_slugs%5B%5D=interview-preparation-kit&playlist_slugs%5B%5D=dictionaries-hashmaps&h_r=next-challenge&h_v=zen)

* **Test case 0**
* **Test case 1**
* **Test case 2**
* **Test case 3**
* **Test case 4**
* **Test case 5**
* **Test case 6**
* **Test case 7**

Compiler Message