



# Strings: Making Anagrams ☆

[Problem](#)[Submissions](#)[Leaderboard](#)[Editorial](#)

Check out the resources on the page's right side to learn more about strings. The video tutorial is by Gayle Laakmann McDowell, author of the best-selling interview book [Cracking the Coding Interview](#).

Alice is taking a cryptography class and finding anagrams to be very useful. We consider two strings to be anagrams of each other if the first string's letters can be rearranged to form the second string. In other words, both strings must contain the same exact letters in the same exact frequency. For example, bacdc and dcbac are anagrams, but bacdc and dcbad are not.

Alice decides on an encryption scheme involving two large strings where encryption is dependent on the minimum number of character deletions required to make the two strings anagrams. Can you help her find this number?

Given two strings, **a** and **b**, that may or may not be of the same length, determine the minimum number of character deletions required to make **a** and **b** anagrams. Any characters can be deleted from either of the strings.

For example, if **a** = **cde** and **b** = **dcf**, we can delete **e** from string **a** and **f** from string **b** so that both remaining strings are **cd** and **dc** which are anagrams.

## Function Description

Complete the `makeAnagram` function in the editor below. It must return an integer representing the minimum total characters that must be deleted to make the strings anagrams.

`makeAnagram` has the following parameter(s):

- **a**: a string
- **b**: a string

## Input Format

The first line contains a single string, **a**.

The second line contains a single string, **b**.

## Constraints

- $1 \leq |a|, |b| \leq 10^4$
- The strings **a** and **b** consist of lowercase English alphabetic letters `ascii[a-z]`.

## Output Format

Print a single integer denoting the number of characters you must delete to make the two strings anagrams of each other.

## Sample Input

```
cde
abc
```

## Sample Output



4

### Explanation

We delete the following characters from our two strings to turn them into anagrams of each other:

1. Remove d and e from cde to get c.
2. Remove a and b from abc to get c.

We must delete **4** characters to make both strings anagrams, so we print **4** on a new line.

C#



```
19
20     var count = 0;
21     var aux = 0;
22
23     var temp = a.GroupBy(x => x)
24         .Select(y => new{ Id = y.Key, Cont = y.Count()})
25         .OrderBy(z => z.Cont);
26     var temp2 = b.GroupBy(x => x)
27         .Select(y => new{ Id = y.Key, Cont = y.Count()})
28         .OrderBy(z => z.Cont);
29
30     foreach(var a_item in temp)
31     {
32         foreach(var b_item in temp2)
33         {
34             if(a_item.Id == b_item.Id)
35             {
36                 count += Math.Abs(a_item.Cont - b_item.Cont);
37                 aux += a_item.Cont + b_item.Cont;
38             }
39         }
40     }
41     return a.Length + b.Length + count - aux ;
42 }
43
44 static void Main(string[] args) {
45     TextWriter textWriter = new StreamWriter(@System.Environment.GetEnvironmentVariable
46     ("OUTPUT_PATH"), true);
47     string a = Console.ReadLine();
```

Line: 41 Col: 44

[Upload Code as File](#) ☐ [Test against custom input](#)[Run Code](#)[Submit Code](#)

## Congratulations!

[Facing any Issues? Let us know!](#)

You have passed the sample test cases. Click the submit button to run your code against all the test cases.