**Alphabet symmetry**

20894% of 16481 of 799[KenKamau](http://www.codewars.com/users/KenKamau)

* C#
* Mono 4.2.3
  + VIM
  + EMACS

Instructions

Output

* Consider the word "abode". We can see that the letter a is in position 1 and b is in position 2. In the alphabet, a and b are also in positions 1 and 2. Notice also that d and e in abode occupy the positions they would occupy in the alphabet, which are positions 4 and 5.

Given an array of words, return an array of the number of letters that occupy their positions in the alphabet for each word. For example, solve(["abode","ABc","xyzD"]) = [4,3,1]. See test cases for more examples.

Input will consist of alphabet characters, both uppercase and lowercase. No spaces.

Good luck!

If you like this Kata, please try:

[Last digit symmetry](https://www.codewars.com/kata/59a9466f589d2af4c50001d8)

[Alternate capitalization](https://www.codewars.com/kata/59cfc000aeb2844d16000075)

<http://www.codewars.com/kata/alphabet-symmetry/train/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp88

{

class Program

{

public static List<int> Solve(List<string> arr)

{

//throw new NotImplementedException();

string alfab = "abcdefghijklmnopqrstuvwxyz";

List<int> ans = new List<int>();

foreach(string elem in arr)

{

int cont = 0;

string tolow = elem.ToLower();

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

{

if(i<alfab.Length && tolow[i] == alfab[i])

{

cont++;

}

}

ans.Add(cont);

}

return ans;

}

static void Main(string[] args)

{

Console.ReadLine();

}

}

}