**6 kyu**

**Consonant value**

221193% of 26992 of1,202[KenKamau](https://www.codewars.com/users/KenKamau" \o "This kata's Sensei)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/consonant-value/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/consonant-value/solutions/csharp)

[Discourse (45)](https://www.codewars.com/kata/consonant-value/discuss/csharp)

* Add to Collection
* |
* Share this kata:

A consonant is any letter of the alphabet except a, e, i ,o, u. The consonant substrings in the word "zodiacs" are z, d, cs. Assuming a = 1, b = 2 ... z = 26, the values of these substrings are 26 ,4, 22 because z = 26,d = 4,cs=3+19=22. The maximum value of these substrings is 26. Therefore, solve("zodiacs") = 26.

Given a lowercase string that has alphabetic characters only and no spaces, return the highest value of consonant substrings.

Good luck!

If you like this Kata, please try:

[Word values](https://www.codewars.com/kata/598d91785d4ce3ec4f000018)

[Vowel-consonant lexicon](https://www.codewars.com/kata/59cf8bed1a68b75ffb000026)

<https://www.codewars.com/kata/consonant-value/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

//static int maxSubArraySum(int[] a, int size)

//{

// int max\_so\_far = a[0];

// int curr\_max = a[0];

// for (int i = 1; i < size; i++)

// {

// curr\_max = Math.Max(a[i], curr\_max + a[i]);

// max\_so\_far = Math.Max(max\_so\_far, curr\_max);

// }

// return max\_so\_far;

//}

public static int Solve(string a)

{

int max\_so\_far = 0;

int curr\_max = 0 ;

if(!"aeiou".Contains(a[0]))

{

max\_so\_far = a[0] - 'a' + 1;

curr\_max = max\_so\_far;

}

for (int i = 1; i < a.Length; i++)

{

//curr\_max = Math.Max(a[i], curr\_max + a[i]);

//max\_so\_far = Math.Max(max\_so\_far, curr\_max);

if (!"aeiuo".Contains(a[i]))

{

curr\_max = Math.Max(a[i]-'a'+1, curr\_max + (a[i] - 'a'+1));

}

else

{

curr\_max = 0;

}

max\_so\_far = Math.Max(max\_so\_far, curr\_max);

}

return max\_so\_far;

}

static void Main(string[] args)

{

string s = "twelfthstreet"; //103

Console.WriteLine(Solve(s));

Console.ReadLine();

}

}

}