**7 kyu**

**Reversing Fun**

531194% of 190198of 1,686[Shivo](https://www.codewars.com/users/Shivo)

C#

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

Details

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

[Forks (2)](https://www.codewars.com/kata/reversing-fun/forks/csharp)

[Discourse (69)](https://www.codewars.com/kata/reversing-fun/discuss/csharp)

* Add to Collection
* |
* Share this kata:

You are going to be given a string. Your job is to return that string in a certain order that I will explain below:

Let's say you start with this: 012345

The first thing you do is reverse it:543210  
Then you will take the string from the 1st position and reverse it again:501234  
Then you will take the string from the 2nd position and reverse it again:504321  
Then you will take the string from the 3rd position and reverse it again:504123

Continue this pattern until you have done every single position, and then you will return the string you have created. For this particular number, you would return:504132

#Input: A string of length 1 - 1000

#Output: A correctly reordered string.

<https://www.codewars.com/kata/reversing-fun/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp2

{

class Program

{

public static string FlipNumber(string n)

{

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

{

string sub = n.Substring(i, n.Length - i);

Console.WriteLine(sub);

char[] ch = sub.ToCharArray();

Array.Reverse(ch);

n = n.Substring(0, i) + new string(ch);

}

return n;

}

static void Main(string[] args)

{

//Console.WriteLine( FlipNumber("1234567"));

Console.WriteLine(FlipNumber("012345"));

Console.ReadLine();

}

}

}

[dcieslak](https://www.codewars.com/users/dcieslak)

**using System.Text;**

**public class Kata**

**{**

**public static string FlipNumber(string n) {**

**StringBuilder sb = new StringBuilder();**

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

**sb.Append( i %2 == 0 ? n[n.Length- (int)( i/2)-1] : n[(int) (i/2)]);**

**return sb.ToString();**

**}**

**}**