**4 kyu**

**Next bigger number with the same digits**

115016195% of1,306754 of7,259[GiacomoSorbi](https://www.codewars.com/users/GiacomoSorbi)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/next-bigger-number-with-the-same-digits/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/next-bigger-number-with-the-same-digits/solutions/csharp)

[Forks (18)](https://www.codewars.com/kata/next-bigger-number-with-the-same-digits/forks/csharp)

[Discourse (425)](https://www.codewars.com/kata/next-bigger-number-with-the-same-digits/discuss/csharp)

* Add to Collection
* |
* Share this kata:

You have to create a function that takes a positive integer number and returns the next bigger number formed by the same digits:

12 ==> 21

513 ==> 531

2017 ==> 2071

If no bigger number can be composed using those digits, return -1:

9 ==> -1

111 ==> -1

531 ==> -1

<https://www.codewars.com/kata/next-bigger-number-with-the-same-digits/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

//static int find\_x(int n)

//{

// int x = 0;

// for(int i =0; i<n; i++)

// {

// for(int j =0; j<2\*n; j++)

// {

// x += j + i;

// }

// }

// return x;

//}

public static bool nextPermutation(char[] array)

{

// Find non-increasing suffix

int i = array.Length - 1;

while (i > 0 && array[i - 1] >= array[i])

i--;

if (i <= 0)

return false;

// Find successor to pivot

int j = array.Length - 1;

while (array[j] <= array[i - 1])

j--;

char temp = array[i - 1];

array[i - 1] = array[j];

array[j] = temp;

// Reverse suffix

j = array.Length - 1;

while (i < j)

{

temp = array[i];

array[i] = array[j];

array[j] = temp;

i++;

j--;

}

return true;

}

public static long NextBiggerNumber(long n)

{

//code me

char[] ch = n.ToString().ToCharArray();

nextPermutation(ch);

//Console.WriteLine(new string(ch));

long res = long.Parse(new string(ch));

if (res == n) return -1;

return res;

}

static void Main(string[] args)

{

//for (int i = 1; i < 20; i++)

//{

// Console.WriteLine(find\_x(i));

//}

// Console.WriteLine(NextSmallerPronic(22));

Console.WriteLine( NextBiggerNumber(9876543201));

Console.ReadLine();

}

}

}