**6 kyu**

**Sort the odd**

51512291% of2,2471,267 of 14,210[fyvfyv](https://www.codewars.com/users/fyvfyv)

C#

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

Details

[Solutions](https://www.codewars.com/kata/sort-the-odd/solutions/csharp)

[Forks (9)](https://www.codewars.com/kata/sort-the-odd/forks/csharp)

[Discourse (98)](https://www.codewars.com/kata/sort-the-odd/discuss/csharp)

* Add to Collection
* |
* Share this kata:

You have an array of numbers.  
Your task is to sort ascending odd numbers but even numbers must be on their places.

Zero isn't an odd number and you don't need to move it. If you have an empty array, you need to return it.

*Example*

sortArray([5, 3, 2, 8, 1, 4]) == [1, 3, 2, 8, 5, 4]

<https://www.codewars.com/kata/sort-the-odd/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

public static int[] SortArray(int[] ar)

{

for (int i = 0; i < ar.Length - 1; i++)

{

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

{

if (ar[i] > ar[j] && ar[i] %2 !=0 && ar[j] %2 !=0)

{

int temp = ar[i];

ar[i] = ar[j];

ar[j] = temp;

}

}

}

return ar;

}

static void Main(string[] args)

{

int[] arr = new int[] { 5, 3, 2, 8, 1, 4 };

//1, 3, 2, 8, 5, 4

foreach (int item in SortArray(arr))

{

Console.Write(item + " ");

}

Console.ReadLine();

}

}

}