**7 kyu**

**Simple Fun #87: Shuffled Array**

9293% of 12927 of405[myjinxin2015](https://www.codewars.com/users/myjinxin2015)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/simple-fun-number-87-shuffled-array/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/simple-fun-number-87-shuffled-array/solutions/csharp)

[Discourse (23)](https://www.codewars.com/kata/simple-fun-number-87-shuffled-array/discuss/csharp)

* Add to Collection
* |
* Share this kata:

**Task**

A noob programmer was given two simple tasks: sum and sort the elements of the given array arr = [a1, a2, ..., an].

He started with summing and did it easily, but decided to store the sum he found in some random position of the original array which was a bad idea. Now he needs to cope with the second task, sorting the original array arr, and it's giving him trouble since he modified it.

Given the array shuffled, consisting of elements a1, a2, ..., an, and their sumvalue in random order, return the sorted array of original elements a1, a2, ..., an.

**Example**

For shuffled = [1, 12, 3, 6, 2], the output should be [1, 2, 3, 6].

1 + 3 + 6 + 2 = 12, which means that 1, 3, 6 and 2 are original elements of the array.

For shuffled = [1, -3, -5, 7, 2], the output should be [-5, -3, 2, 7].

**Input/Output**

* [input] integer array shuffled

Array of at least two integers. It is guaranteed that there is an index i such that shuffled[i] = shuffled[0] + ... + shuffled[i - 1] + shuffled[i + 1] + ... + shuffled[n].

Constraints:

2 ≤ shuffled.length ≤ 30,

-300 ≤ shuffled[i] ≤ 300.

* [output] an integer array

A sorted array of shuffled.length - 1 elements.

<https://www.codewars.com/kata/simple-fun-number-87-shuffled-array/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

public static int[] ShuffledArray(int[] shuffled)

{

//coding and coding..

//int ind = -1;

int elem = 0;

int sum = shuffled.Sum();

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

{

if(sum - shuffled[i] == shuffled[i])

{

elem = shuffled[i];

break;

}

}

List<int> ans = shuffled.ToList();

ans.Remove(elem);

return ans.ToArray();

}

static void Main(string[] args)

{

// For shuffled = [1, -3, -5, 7, 2],

// the output should be[-5, -3, 2, 7].

//For shuffled = [1, 12, 3, 6, 2],

//the output should be[1, 2, 3, 6].

//int[] arr = { 1, -3, -5, 7, 2 };

int[] arr = {2,3,6,11 };

foreach (int item in ShuffledArray(arr) )

{

Console.Write(item + " ");

}

Console.ReadLine();

}

}

}