A noob programmer was given two simple tasks: sum and sort the elements of the given array  
a = [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 a, and it's giving him trouble since he modified it.

Given the array shuffled, consisting of elements a1, a2, ..., an, a1 + a2 + ... + an 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  
  shuffledArray(shuffled) = [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  
  shuffledArray(shuffled) = [-5, -3, 2, 7].

Input/Output

* **[execution time limit] 3 seconds (cs)**
* **[input] array.integer 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].

*Guaranteed constraints:*  
2 ≤ shuffled.length ≤ 15,  
-300 ≤ shuffled[i] ≤ 300.

* **[output] array.integer**
  + A sorted array of shuffled.length - 1 elements.

**[C#] Syntax Tips**

// Prints help message to the console

// Returns a string

**string** **helloWorld**(**string** name) {

Console.Write("This prints to the console when you Run Tests");

**return** "Hello, " + name;

}

<https://app.codesignal.com/arcade/code-arcade/sorting-outpost/s4BEFMcpLdGbjX9KX/description>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

class Principal

{

//static int[][] starRotation(int[][] matrix, int width, int[] center, int t)

//{

//}

static int[] shuffledArray(int[] shuffled)

{

int sum = shuffled.Sum();

//bool flag = false;

List<int> lista = new List<int>();

lista.AddRange(shuffled);

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

{

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

{

lista.RemoveAt(i);

break;

}

}

lista.Sort();

return lista.ToArray();

}

public static void Main(string[] args)

{

int[] shuffled = { 1, 12, 3, 6, 2 };

//int[] shuffled= { 2, -1, 2, 2, -1}; //[-1, -1, 2, 2]

int[] ans = shuffledArray(shuffled);

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

{

Console.Write(ans[i] + " ");

}

Console.ReadLine();

}

}