You need to figure out a way to find all duplicates in a list of numbers numberListprovided to you, and return them in an array sorted in the order they appear innumberList.

Additional constraints for those who really like challenges:

* you're not allowed to create more variables;
* all operations should be done on the original list;
* the run time should be linear.

Of course, you can choose to ignore these constraints and submit a solution but where is fun in that?

**Example**

For numberList = [1,2,1,1,3,2,4,6,5], the output should be  
GetDuplicates(numberList) = [1,2].

* **[time limit] 3000ms (cs)**
* **[input] array.integer numberList**

List of numbers to check for duplicates.

*Constraints:*  
0 ≤ numberList.length ≤ 150,  
1 ≤ numberList[i] ≤ 4 · 105.

* **[output] array.integer**

Array of duplicates in the order they appear.

<https://codefights.com/challenge/x4fLDh2ZYHbaycvFX/main>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static int[] GetDuplicates(int[] numberList)

{

var ans = new List<int>();

foreach (int elem in numberList)

{

var numQuery = numberList.Where(num => num == elem);

if (numQuery.Count() > 1 && !ans.Contains(elem))

{

ans.Add(elem);

}

}

return ans.ToArray();

}

static void Main(string[] args)

{

int[] arr= {1,1,2,2,2,3};

foreach (int elem in GetDuplicates(arr))

{

Console.Write(elem + " ");

}

Console.ReadLine();

}

}

}