*Note: Write a solution with O(n) time complexity and O(1) additional space complexity, since this is what you would be asked to do during a real interview.*

Given an array array that contains only numbers in the range from 1 to array.length, find the first duplicate number for which the second occurrence has the minimal index. In other words, if there are more than 1 duplicated numbers, return the number for which the second occurrence has a smaller index than the second occurrence of the other number does. If there are no such elements, return -1.

**Example**

* For array = [2, 3, 3, 1, 5, 2], the output should be  
  firstDuplicate(array) = 3.  
  There are 2 duplicates: numbers 2 and 3. The second occurrence of 3 has a smaller index than than second occurrence of 2 does, so the answer is 3.
* For array = [2, 4, 3, 5, 1], the output should be  
  firstDuplicate(array) = -1.

**Input/Output**

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

*Guaranteed constraints:*  
1 ≤ array.length ≤ 10^4,  
1 ≤ a[i] ≤ a.length.

* **[output] integer**

The element in a that occurs in the array more than once and has the minimal index for its second occurrence. If there are no such elements, return -1.

**[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://codefights.com/challenge/yzcgNgQZHYkQrAfcR/solutions>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp104

{

class Program

{

static int firstDuplicateNumber(int[] array)

{

Dictionary<int, List<int>> dictionary = new Dictionary<int, List<int>>();

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

{

if (dictionary.ContainsKey(array[i]))

{

dictionary[array[i]].Add(i);

}

else

{

dictionary[array[i]] = new List<int>();

dictionary[array[i]].Add(i);

}

}

//foreach(KeyValuePair<int,List<int>> kvp in dictionary)

//{

// Console.Write(kvp.Key + " ");

//}

//Console.WriteLine();

int min\_seg\_indice = array.Length+1;

int min\_key = -1;

foreach (KeyValuePair<int, List<int>> kvp in dictionary)

{

if (kvp.Value.Count > 1 && kvp.Value[1] < min\_seg\_indice)

{

min\_key = kvp.Key;

min\_seg\_indice = kvp.Value[1];

}

}

return min\_key;

}

static void Main(string[] args)

{

int[] array = { 2, 3, 3, 1, 5, 2 };

Console.WriteLine(firstDuplicateNumber(array));

Console.ReadLine();

}

}

}

-------------SOLUCION POR marisa\_h --------------------

static int firstDuplicateNumber(int[] a)

{

int[] s = new int[10000];

foreach (int i in a)

{

if (s[i]==1)

return i;

s[i] = 1;

}

return -1;

}

------------SOLUCION POR c\_w -------------------------

static int firstDuplicateNumber(int[] A)

{

HashSet<int> S = new HashSet<int>();

foreach (int i in A)

if (!S.Add(i))

return i;

return -1;

}