**难度: 简单**

魔术索引。 在数组A[0...n-1]中，有所谓的魔术索引，满足条件A[i] = i。给定一个有序整数数组，编写一种方法找出魔术索引，若有的话，在数组A中找出一个魔术索引，如果没有，则返回-1。若有多个魔术索引，返回索引值最小的一个。

示例1:

输入：nums = [0, 2, 3, 4, 5]

输出：0

说明: 0下标的元素为0

示例2:

输入：nums = [1, 1, 1]

输出：1

说明:

nums长度在[1, 1000000]之间

此题为原书中的 Follow-up，即数组中可能包含重复元素的版本

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/magic-index-lcci

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

**思路1:**

1). 使用循环, 判断是否数组的数值是否等于下标的值, 如果相等直接放回该下标,

2). 循环结束表示没有相等的下标值, 直接返回 -1

|  |
| --- |
| public int findMagicIndex(int[] nums) {  for (int i = 0; i < nums.length; i++) {  if (nums[i] == i) {  return i;  }  }  return -1;  } |

**思路2(优化):**

1). 因为这是一个有序的int类型数组,所有当nums[i]的数大于 i的话

2). 我们可以直接把i的下标替换成nums[i]的值,但是必须把i--

3). 因为我们在for循环里面已经++了,如果不减1的话i会变成nums[i]的值加1了

|  |
| --- |
| public int findMagicIndex2(int[] nums) {  for (int i = 0; i < nums.length; i++) {  if (nums[i] == i) {  return i;  }else if (i > 0 && nums[i] > i) {  i = nums[i];  i--;  }  }  return -1;  } |