**难度: 简单**

把一个数组最开始的若干个元素搬到数组的末尾，我们称之为数组的旋转。输入一个递增排序的数组的一个旋转，输出旋转数组的最小元素。例如，数组 [3,4,5,1,2] 为 [1,2,3,4,5] 的一个旋转，该数组的最小值为1。

示例 1：

输入：[3,4,5,1,2]

输出：1

示例 2：

输入：[2,2,2,0,1]

输出：0

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/xuan-zhuan-shu-zu-de-zui-xiao-shu-zi-lcof

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). 先把数组进行排序

2). 然后返回数组下标的第一个

|  |
| --- |
| public int minArray(int[] numbers) {  Arrays.sort(numbers);  return numbers[0];  } |

**思路(2):**

1). 使用二分查找

2). 先比较中间节点的值是否大于最后节点的值,如果大于所有最小值在最右边

那么就左边节点 = 中间节点+1

3). 比较中间节点的值是否小于最后节点的值,如果小于所有最小值在最左边

那么就右边节点 = 中间节点

4). 如果相等,则把右节点减1

|  |
| --- |
| public int minArray2(int[] numbers) {  //左  int i = 0;  //右  int j = numbers.length - 1;  //循环条件  while(i < j) {  //获取中间节点  int mid = (i + j) /2;  //当中间节点大于 最右的节点  if (numbers[mid] > numbers[j]) {  //中间节点+1给左  i = mid +1;  }else if (numbers[mid] < numbers[j]) {  //中间节点给到右  j = mid;  }else {  //相等,右点1  j--;  }  }  //返回  return numbers[i];  } |