



链滴

旋转数组的最小数字

作者: [yudake](#)

原文链接: <https://ld246.com/article/1518763114440>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

题目描述

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。输入一个非递减排序的数的一个旋转，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。NOTE：给出的所有元素都大于0，若数组大小为0，请返回0。

解题思路

实际上就是找两个子数组的分界线，可以利用二分法查找。

$$\text{mid} = \text{low} + (\text{high} - \text{low}) / 2$$

需要考虑三种情况：

- $\text{array}[\text{mid}] > \text{array}[\text{high}]$ ，此时最小数字肯定在mid右边，令 $\text{low} = \text{mid} + 1$ ；
- $\text{array}[\text{mid}] == \text{array}[\text{high}]$ ，此时无法判断，一个一个试， $\text{high} = \text{high} - 1$ ；
- $\text{array}[\text{mid}] < \text{array}[\text{high}]$ ，此时最小数字就是mid或在mid左边， $\text{high} = \text{mid}$ 。

```
public class Solution {
    public int minNumberInRotateArray(int [] array) {
        int low = 0;
        int high = array.length - 1;
        while(low < high){
            int mid = low + (high - low) / 2;
            if(array[mid] > array[high]){
                low = mid + 1;
            }else if(array[mid] == array[high]){
                high = high - 1;
            }else{
                high = mid;
            }
        }
        return array[low];
    }
}
```