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。 NOTE：给出的所有元素都大于0，若数组大小为0，返回0。

分析：非递减数组旋转之后最小值，也就是寻找分界点，分界点前后都是非递减数组，分界点后面的非递减数组比分界点前面的数组都要小

考虑3中情况，注意array[mid] == array[high]:

出现这种情况的array类似 [1,0,1,1,1] 或者[1,1,1,0,1]，此时最小数字不好判断在mid左边

还是右边,这时只好一个一个试

high = high - 1

非递减就是a[i]<=a[i+1]

递减就是a[i]>a[i+1]

非递增就是a[i]>=a[i+1]

递增就是a[i]<a[i+1]

class Solution {

public:

int minNumberInRotateArray(vector<int> rotateArray) {

int len=rotateArray.size();

if(len==0)

return 0;

int left=0,right=len-1;

while(left<right){

int mid=left+(right-left)/2;

if(rotateArray[mid]>rotateArray[right])

left=mid+1;

else if(rotateArray[mid]==rotateArray[right])

right=right-1;

else

right=mid;

}

return rotateArray[left];

}

};