1. 寻找旋转排序数组中的最小值

假设按照升序排序的数组在预先未知的某个点上进行了旋转。

( 例如，数组 [0,1,2,4,5,6,7] 可能变为 [4,5,6,7,0,1,2] )。

请找出其中最小的元素。

你可以假设数组中不存在重复元素。

示例 1:

输入: [3,4,5,1,2]

输出: 1

示例 2:

输入: [4,5,6,7,0,1,2]

输出: 0

解决方案1：

遍历数组，依次比较当前元素和后一个元素的值，如果当前元素大于下一个元素，返回下一个元素，如果遍历完整个数组后都没有找到对应元素，则返回第一个元素

解决方案2：

二分法，可以将时间复杂度从O(N)下降到O(logN)

先比较第一个元素和最后一个元素大小，如果第一个元素小于最后一个元素，则说明该数组没有进行过旋转，直接返回第一个元素

取出数组中间元素，如果该元素大于第一个元素，说明前半部分可以舍去，旋转节点一定的在后半部分，如果该元素大于最后一个元素，说明后半部分可以社区，旋转节点一定在前半部分，递归调用二分法，就能找到旋转点（即最小元素）