1. Find Peak Element

A peak element is an element that is greater than its neighbors.

Given an input array nums, where nums[i] ≠ nums[i+1], find a peak element and return its index.

The array may contain multiple peaks, in that case return the index to any one of the peaks is fine.

You may imagine that nums[-1] = nums[n] = -∞.

**Example 1:**

Input: nums = [1,2,3,1]  
Output: 2  
Explanation: 3 is a peak element and your function should return the index number 2.

**Example 2:**

Input: nums = [1,2,1,3,5,6,4]  
Output: 1 or 5   
Explanation: Your function can return either index number 1 where the peak element is 2,   
 or index number 5 where the peak element is 6.

**Follow up:** Your solution should be in logarithmic complexity.

**解法1** 线性扫描

**解法2** 二分查找

* nums[mid] > nums[mid+1]：nums[mid+1]一定不是峰值，[l, mid]搜索
* nums[mid] <= nums[mid+1]：nums[mid+1]可能是峰值，在[mid+1, r]搜索

class Solution {  
public:  
 int findPeakElement(vector<int>& nums) {  
 int l = 0, r = nums.size()-1;  
 while(l < r){  
 int mid = (l+r)/2;  
 if(nums[mid] > nums[mid+1])r=mid;  
 else l = mid+1;  
 }  
 return l;  
 }  
};