

```
public int firstBadVersion(int n) {  
    if (n == 0) {  
        return -1;  
    }  
  
    int start = 1;  
    int end = n;  
    while (start + 1 < end) {  
        int mid = start + (end - start) / 2;  
        if (isBadVersion(mid) == false) {  
            start = mid;  
        } else {  
            end = mid;  
        }  
    }  
  
    if (isBadVersion(start) == true) {  
        return start;  
    }  
  
    if (isBadVersion(end) == true) {  
        return end;  
    }  
  
    return -1;  
}
```

搜索插入位置

给定一个排序数组和一个目标值，如果在数组中找到目标值则返回索引。
如果没有，返回到它将会被按顺序插入的位置。
你可以假设在数组中无重复元素。

case 1:

输入: [1,3,5,6], 5 输出: 2

case 2:

输入: [1,3,5,6], 2 输出: 1

case 3:

输入: [1,3,5,6], 7 输出: 4

<https://leetcode.com/problems/search-insert-position/description/>