

给定一个未经排序的整数数组，找到最长且**连续**的的递增序列。

**示例 1:**

**输入:** [1,3,5,4,7]

**输出:** 3

**解释:** 最长连续递增序列是 [1,3,5]，长度为3。

尽管 [1,3,5,7] 也是升序的子序列，但它不是连续的，因为5和7在原数组里被4隔开。

**示例 2:**

**输入:** [2,2,2,2,2]

**输出:** 1

**解释:** 最长连续递增序列是 [2]，长度为1。

**注意:** 数组长度不会超过10000。

思路，直接进行遍历，当前数和前一个数比较即可进行排序判断，动态的统计最长的一个连续序列。

```
class Solution {
public:
    int findLengthOfLCIS(vector<int>& nums) {
        int max=1;
        int n=nums.size();
        if(n==0){
            return 0;
        }
        int num=1;
        for(int i=1;i<n;++i){
            if(nums[i]>nums[i-1]){
                num++;
                max=max>num?max:num;
            }else{
                num=1;
            }
        }
        return max;
    }
};
```