

给定一个二进制数组， 计算其中最大连续1的个数。

示例 1:

输入: [1,1,0,1,1,1]

输出: 3

解释: 开头的两位和最后的三位都是连续1， 所以最大连续1的个数是 3.

注意:

- 输入的数组只包含 0 和1。
- 输入数组的长度是正整数， 且不超过 10,000。

思路：直接遍历数组找最大值即可。

```
class Solution {
public:
    int findMaxConsecutiveOnes(vector<int>& nums) {
        int n=nums.size();
        int num=0;
        int max=0;
        for(int i=0;i<n;++i) {
            while(nums[i]!=0&&i<n) {
                ++num;
                ++i;
            }
            max=num>max?num:max;
            num=0;
        }
        return max;
    }
};
```