

数组的度

给定一个非空且只包含非负数的整数数组 `nums`，数组的度的定义是指数组里任一元素出现频数的最大值。

你的任务是在 `nums` 中找到与 `nums` 拥有相同大小的度的最短连续子数组，返回其长度。

示例 1：

输入：[1, 2, 2, 3, 1]

输出：2

解释：

输入数组的度是 2，因为元素 1 和 2 的出现频数最大，均为 2。

连续子数组里面拥有相同度的有如下所示：

[1, 2, 2, 3, 1], [1, 2, 2, 3], [2, 2, 3, 1], [1, 2, 2], [2, 2, 3], [2, 2]

最短连续子数组 [2, 2] 的长度为 2，所以返回 2。

示例 2：

输入：[1,2,2,3,1,4,2]

输出：6

```
class Solution {
public:
    int findShortestSubArray(vector<int>& nums) {
        unordered_map<int,int> map;
        int maxcout=0;
        //通过 map 映射 找到最大度
        for(auto& num:nums)
        {
            map[num]++;
            maxcout=max(maxcout,map[num]);
        }
        //寻找相同度的最短子数组
        int minlen=INT_MAX;
        for(int i=0;i<nums.size();i++)
        {
            if(map[nums[i]]==maxcout)//从当前最大度的对应数字开始向后遍历
            {
                for(int j=i;j<nums.size();j++)
                {
```

if(nums[j]==nums[i])//遍历过程中每出现相同数字 对应 map 中
减一，直至为 0，遍历结束获取长度 做最短长度比较

```
        {
            map[nums[i]]--;
        }
        if(map[nums[i]]==0)
        {
            minlen=min(minlen,j-i+1);//获取当前度下 最短子数组长
            break;
        }
    }
}
return minlen;
}
};
```