

## 找到所有数组中消失的数字

给定一个范围在  $1 \leq a[i] \leq n$  ( $n =$  数组大小) 的 整型数组，数组中的元素一些出现了两次，另一些只出现一次。

找到所有在  $[1, n]$  范围之内没有出现在数组中的数字。

您能在不使用额外空间且时间复杂度为  $O(n)$  的情况下完成这个任务吗？你可以假定返回的数组不算在额外空间内。

示例：

输入：

[4,3,2,7,8,2,3,1]

输出：

[5,6]

```
class Solution {
public:
    vector<int> findDisappearedNumbers(vector<int>& nums) {
        /*vector<int> res;
        if(nums.size()==0)
        {
            return res;
        }
        sort(nums.begin(),nums.end()); //先对其进行从小到大排序
        //对排序好的进行遍历查找缺失项
        for(int i=0;i<nums.size()-1;i++)
        {
            if(nums[i+1]-nums[i]>1)
            {
                int k=nums[i+1]-nums[i];
                for(int j=1;j<k;j++)
                {
                    res.push_back(nums[i]+j);
                }
            }
        }
        return res;*/

        ///通过建立一种映射关系来完成查询
        //添加 0 号索引
    }
```

```
vector<int> res;
nums.push_back(0); //0 号索引尾插 然后桶排序
for(int i=0;i<nums.size();i++)
{
    while(nums[i]!=nums[nums[i]])
    {
        swap(nums[i],nums[nums[i]]);
    }
}
for(int j=1;j<nums.size();j++)
{
    if(j!=nums[j])
    {
        res.push_back(j);
    }
}
return res;
};
```