一、题目说明

题目448. Find All Numbers Disappeared in an Array，给定有n个数的数组，每个数`1<=a[i]<=n`,其中一些数字出现2次，一些出现一次。计算包含于1-n中未在数组中出现的数字。难度是Easy!

二、我的解答

这个题目，思考一下，一次循环，将数字i放到第i+1上面。然后不在位置上的就是了。

```
class Solution{
    public:
        vector<int> findDisappearedNumbers(vector<int>& nums){
            vector<int> res;
            int len = nums.size();
            for(int i=0;i<len;i++){
                while(nums[i]!= i+1 && nums[nums[i]-1] !=nums[i]){
                    swap(nums[i],nums[nums[i]-1]);
                }
            }

            for(int i=0;i<len;i++){
                if(nums[i] != i+1){
                    res.push_back(i+1);
                }
            }
            return res;
        }
};
```

性能如下:

```
Runtime: 112 ms, faster than 92.54% of C++ online submissions for Find All
Numbers Disappeared in an Array.
Memory Usage: 14.9 MB, less than 86.67% of C++ online submissions for Find All
Numbers Disappeared in an Array.
```

三、优化措施

无