DSA

Array

1. Majority element n/2

Link: https://leetcode.com/problems/majority-element/

Code:

class Solution {

public:

    int majorityElement(vector<int>& nums) {

       if(nums.size() == 1) return nums[0];

        int count = 1;

        int majorityElement = nums[0];

        for(int i = 1; i<nums.size(); i++){

            if(nums[i] == majorityElement) count++;

            else count--;

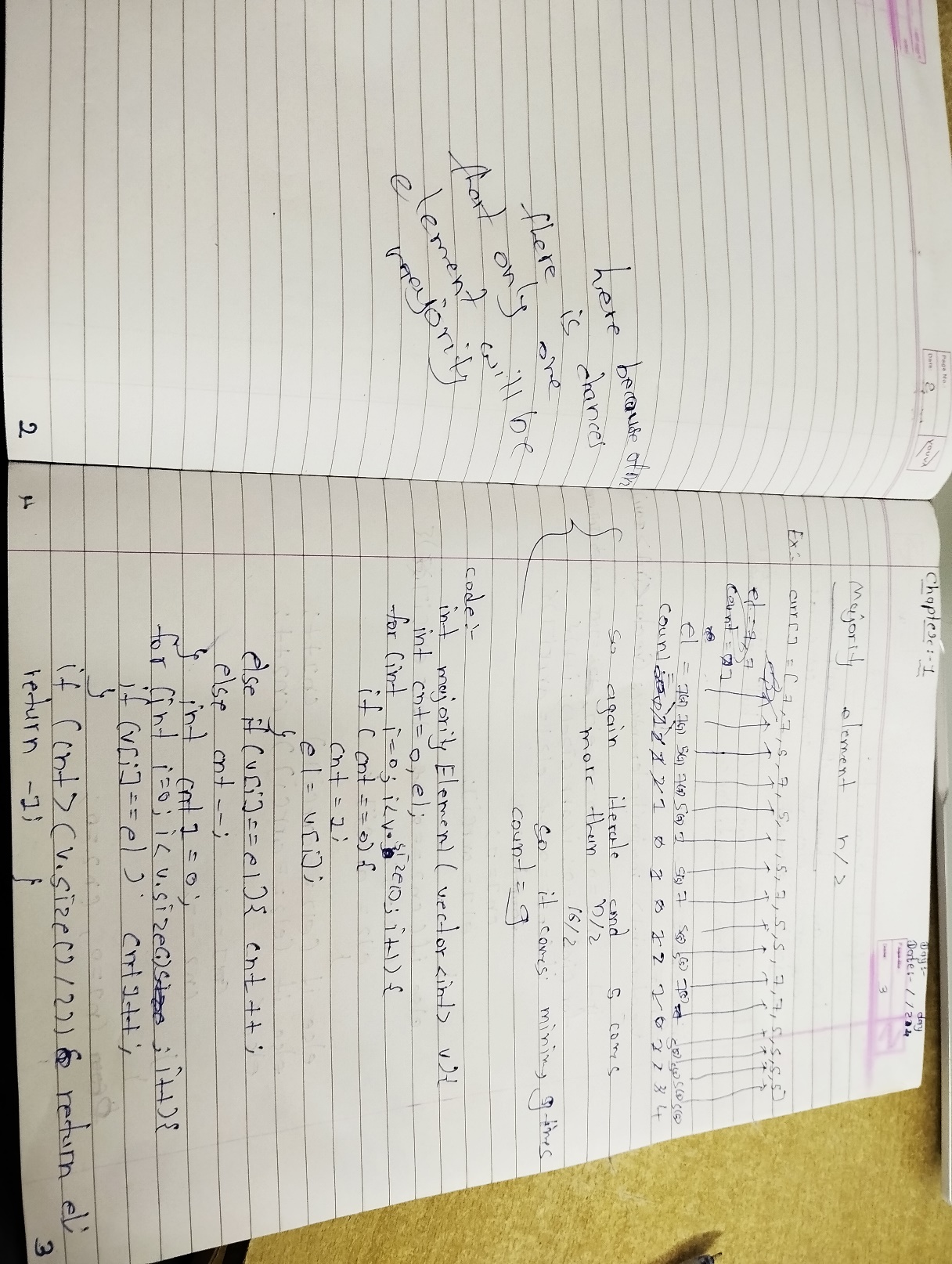
            if(!count) majorityElement = nums[i], count = 1;

        }

        return majorityElement;

    }

};



1. Majority element n/3

Link: https://leetcode.com/problems/majority-element-ii/

Code:

class Solution {

public:

    vector<int> majorityElement(vector<int>& arr) {

    // Write your code here.

    int cont1 = 0,cont2 = 0,el1=INT\_MIN,el2=INT\_MIN;

    for(int i = 0;i<arr.size();i++){

        if(cont1==0 && el2!=arr[i]){

            cont1=1;

            el1=arr[i];

        }

        else if(cont2==0 && el1!=arr[i]){

            cont2=1;

            el2=arr[i];

        }

        else if(el1==arr[i]) cont1++;

        else if(el2==arr[i]) cont2++;

        else {

            cont1--;

            cont2--;

        }

    }

    cont1 = 0,cont2 = 0;

    for(int i = 0;i<arr.size();i++){

        if(el1==arr[i]) cont1++;

        if(el2==arr[i]) cont2++;

    }

    vector<int> ans;

    if(cont1>(arr.size()/3)) ans.push\_back(el1);

    if(cont2>(arr.size()/3)) ans.push\_back(el2);

    // sort(ans.begin(),ans.end());

    return ans;

    }

};

