CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

NAAC GRADE A+
Accredited University

Worksheet 3

Student Name- Akshat Chaudhary

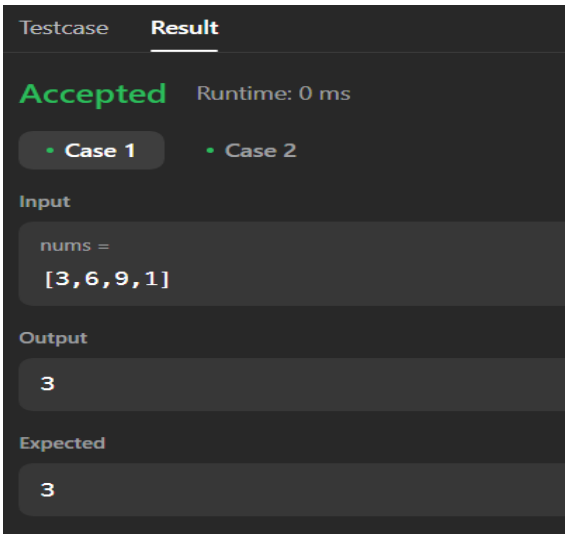
Student UID- 20BCS5751

Que-1: Maximum Gap

Code:

```
class Solution {
public:
    int maximumGap(vector<int>& nums) {
        sort(nums.begin(), nums.end());
        int ans=0;
        for(int i=0;i<nums.size()-1;i++){
            if(ans<(nums[i+1]-nums[i])){
                ans=nums[i+1]-nums[i];
            }
        }
        return ans;
    }
};
```

Output:



Que-2: Sort Colors

Code:

```
class Solution {
public:
    void sortColors(vector<int>& nums) {
        int start=0;
        int end=nums.size()-1;
        int i=0;
        while(i<=end){
            if(nums[i]==0){
                int temp=nums[i];
                nums[i]=nums[start];
                nums[start]=temp;
                start++;
                i++;
            }
            else if(nums[i]==2){
                int temp=nums[i];
                nums[i]=nums[end];
                nums[end]=temp;
                end--;
            }
            else{i++;}
        }
    }
};
```

Output:

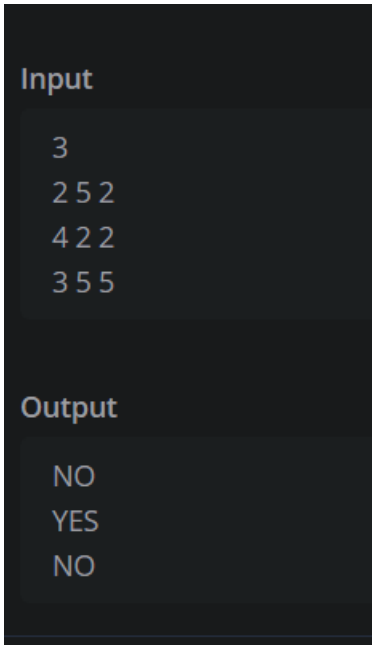
Que-3: Chef and Lockout Draws

Code:

```
#include <iostream>
using namespace std;

int main() {
        int t;
        cin>>t;
        while(t--){
           int a,b,c;
           cin>>a;
           cin>>b;
           cin>>c;
           if(a>b and a>c){
              if(a==b+c){
                 cout<<"YES"<<endl;
              }
              else{
                 cout<<"NO"<<endl;
              }
           }
           else if(b>a and b>c){
              if(b==a+c){
                 cout<<"YES"<<endl;
              }
              else{
                 cout<<"NO"<<endl;
              }
           }
           else{
              if(c==a+b){
                 cout<<"YES"<<endl;
              }
              else{
                 cout<<"NO"<<endl;
              }
           }
        }
```

Output:



Que-4: Turbo Sort

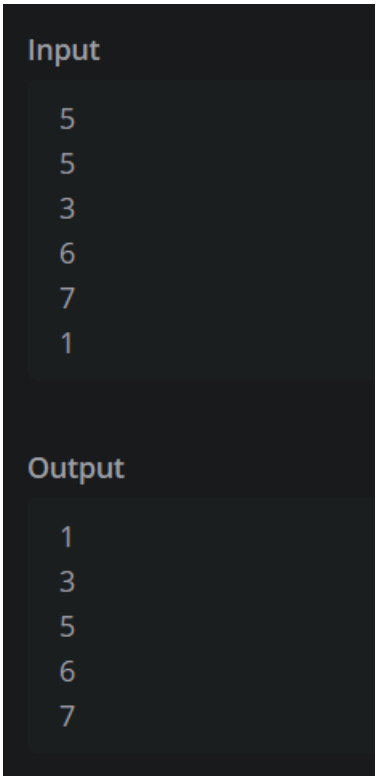
Code:

```
#include <bits/stdc++.h>

using namespace std;

int main() {
	// your code goes here
	int t;
	cin>>t;
	vector <int> a(t);
	for(int i = 0; i< t ; i++){
	    cin>>a[i];
	}
	sort(a.begin(),a.end());
	for(int x : a)
	    cout<<x<<endl;
	return 0;
}
```

Output:



Que-5: [Reorder Data in Log Files]

Code:

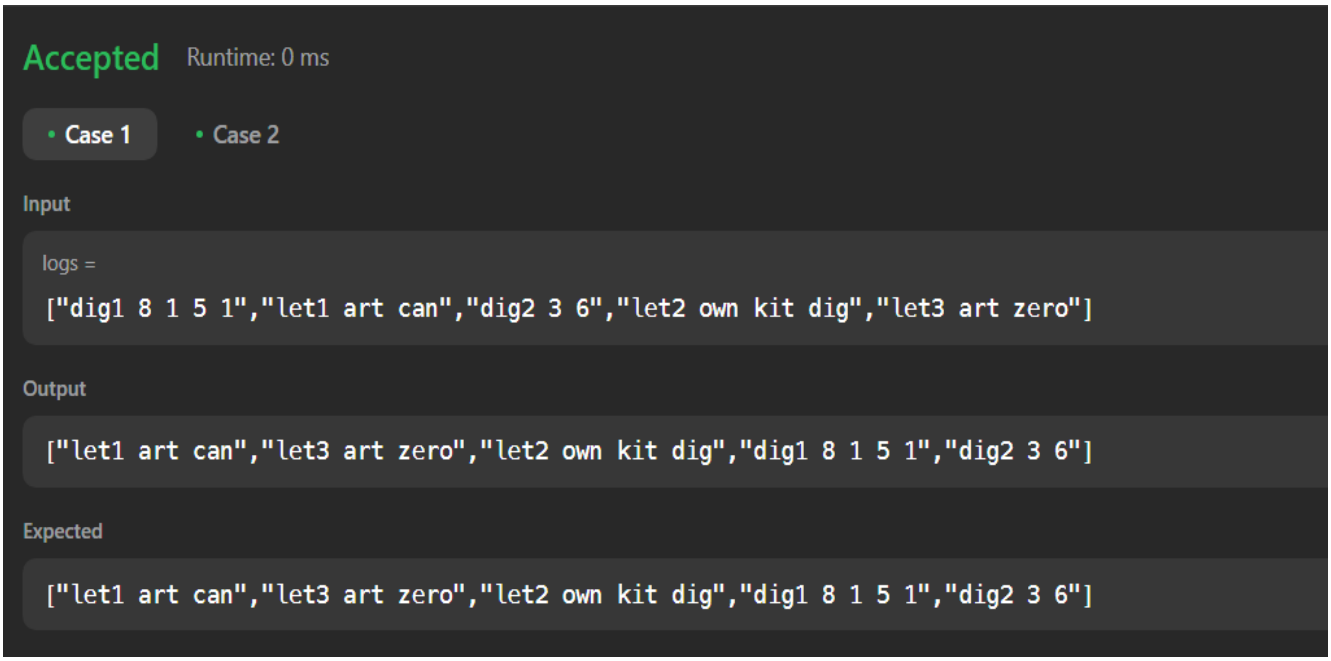
```
class Solution {
public:
    vector<string> reorderLogFiles(vector<string>& logs) {
        auto it = stable_partition(logs.begin(), logs.end(), [](const string& str) {
            return isalpha(str[str.find(' ') + 1]);
        });

        sort(logs.begin(), it, [](const string& str1, const string& str2) {
            auto substr1 = string(str1.begin() + str1.find(' '), str1.end());
            auto substr2 = string(str2.begin() + str2.find(' '), str2.end());
            return (substr1 == substr2) ? str1 < str2 : substr1 < substr2;
        });

        return logs;
    }
```

};

Output:


Accepted
Runtime: 0 ms
Case 1
Case 2
Input
logs =
["dig1 8 1 5 1","let1 art can","dig2 3 6","let2 own kit dig","let3 art zero"]
Output
["let1 art can","let3 art zero","let2 own kit dig","dig1 8 1 5 1","dig2 3 6"]
Expected
["let1 art can","let3 art zero","let2 own kit dig","dig1 8 1 5 1","dig2 3 6"]