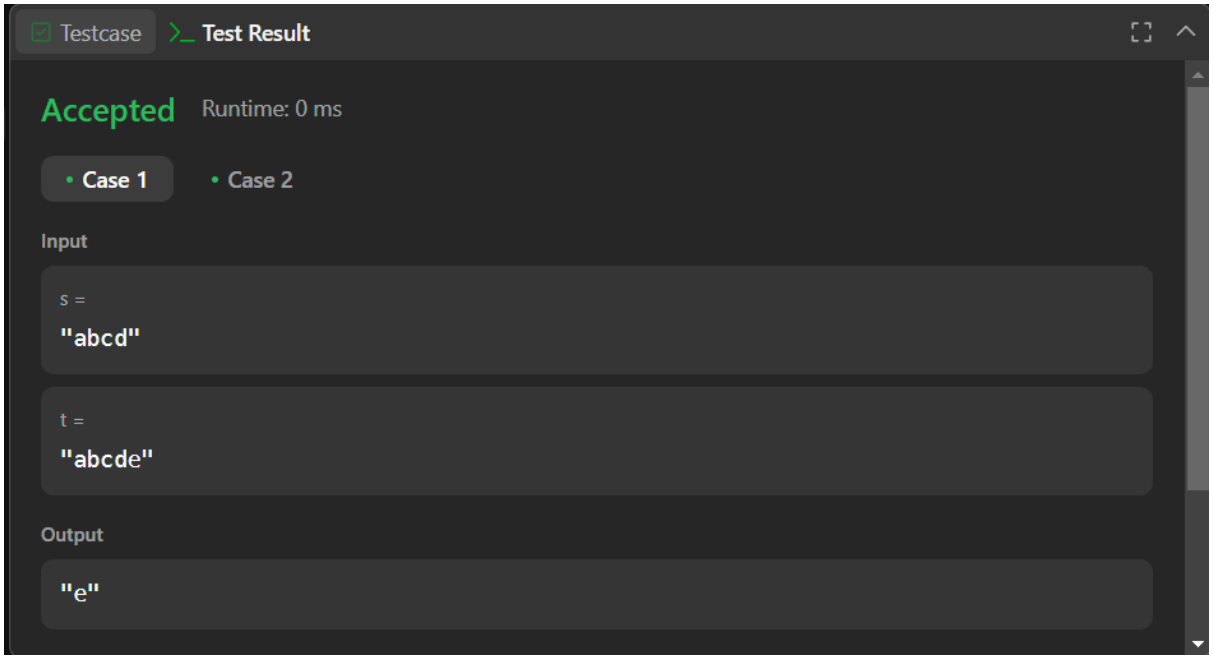
AP Assignment no. 5

Name : Aditya Kr. Ishwar UID : 22BCS14310 Section : 605-B

Ques 1. Find the Difference (389)

Solution :-

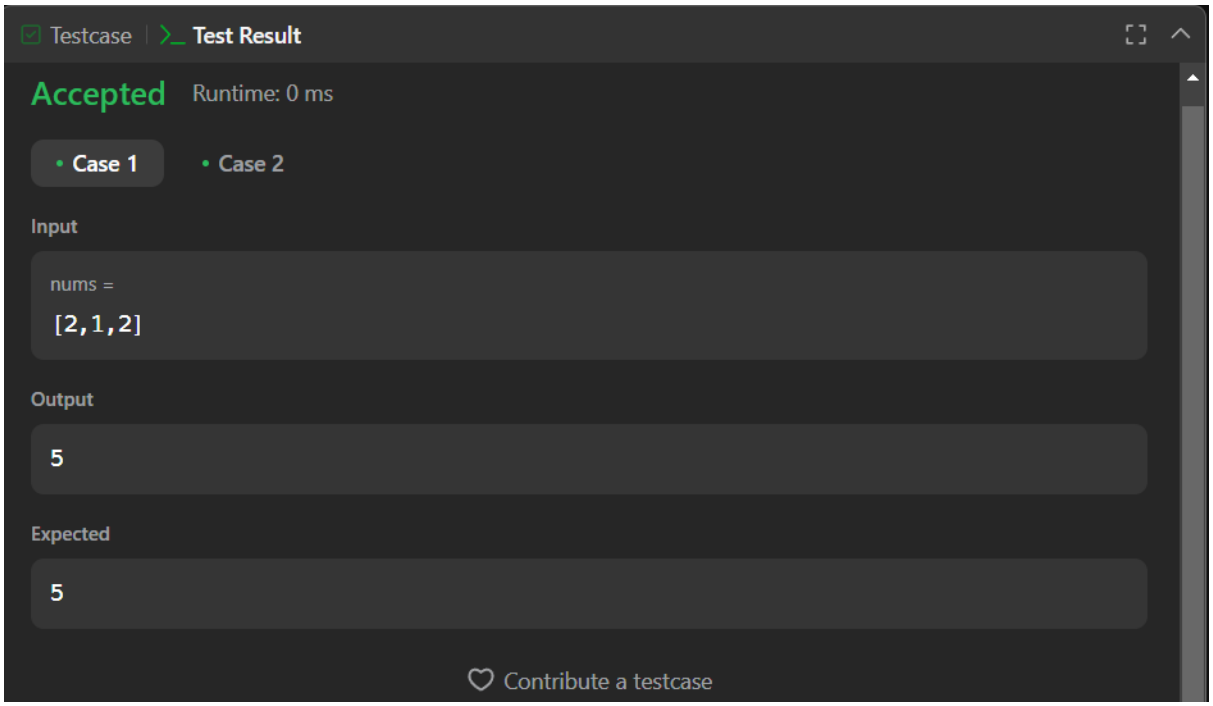
```
class Solution {
public:
   char findTheDifference(string s, string t)
   {
    for(int i=0;i<s.size();i++)
      t[i+1]+=t[i]-s[i];
    return t[t.size()-1];
   }
};
```



Ques no. 2:- Largest Perimeter Triangle (976)

Solution :-

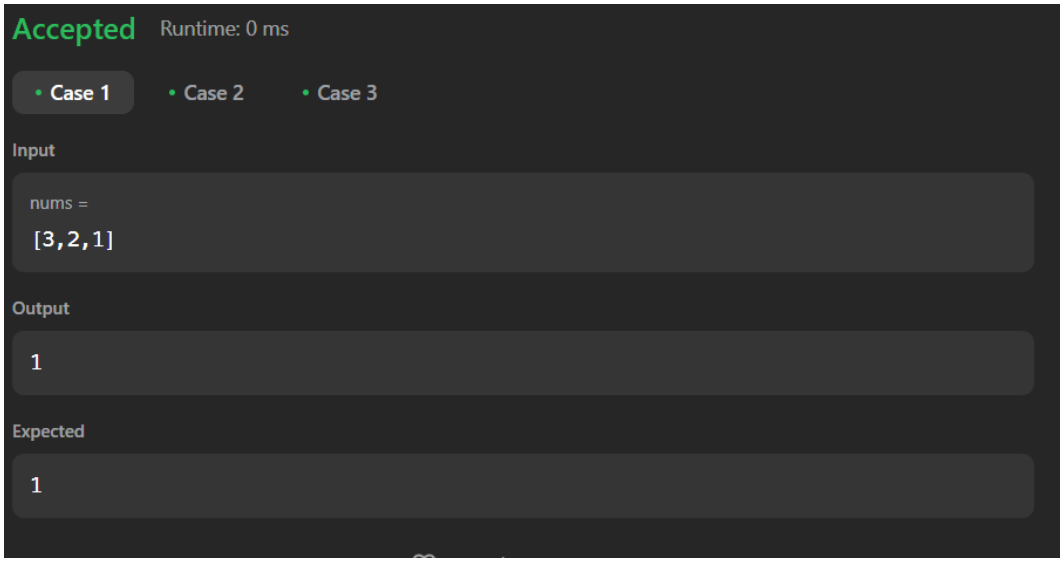
```
class Solution {
public:
    int largestPerimeter(vector<int>& nums) {
        sort(nums.begin(),nums.end());
        for(int i=nums.size()-1;i>1;i--){
            if(nums[i]<nums[i-1]+nums[i-2])
                return nums[i]+nums[i-1]+nums[i-2];
        }
        return 0;
    }
};
```



Ques no.3:- Third Maximum Number(414)

Solution :- class Solution {

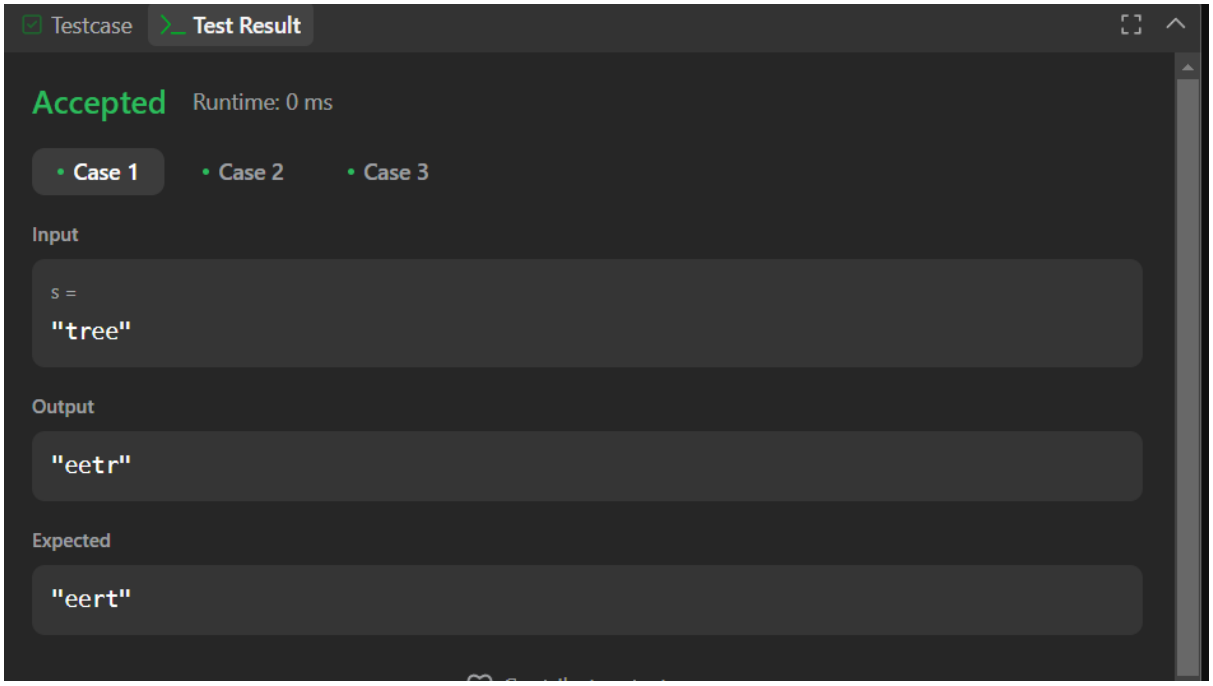
```
public:
    int thirdMax(vector<int>& nums) {
        long max = LONG_MIN, max2 = LONG_MIN, max3 = LONG_MIN;
        for (int i = 0; i < nums.size(); i++) {
            if (nums[i] == max || nums[i] == max2 || nums[i] == max3)
                continue;
            if (nums[i] > max) {
                max3 = max2;
                max2 = max;
                max = nums[i];
            } else if (nums[i] > max2 && nums[i] != max) {
                max3 = max2;
                max2 = nums[i];
            } else if (nums[i] > max3 && nums[i] != max && nums[i] != max2) {
                max3 = nums[i];   } }
        if (nums.size() < 3) {
            return max;    }
        return (max3 == LONG_MIN) ? max : max3;
    } };
```



Ques no. 4:- Sort Characters By Frequency(451)

Solution :-

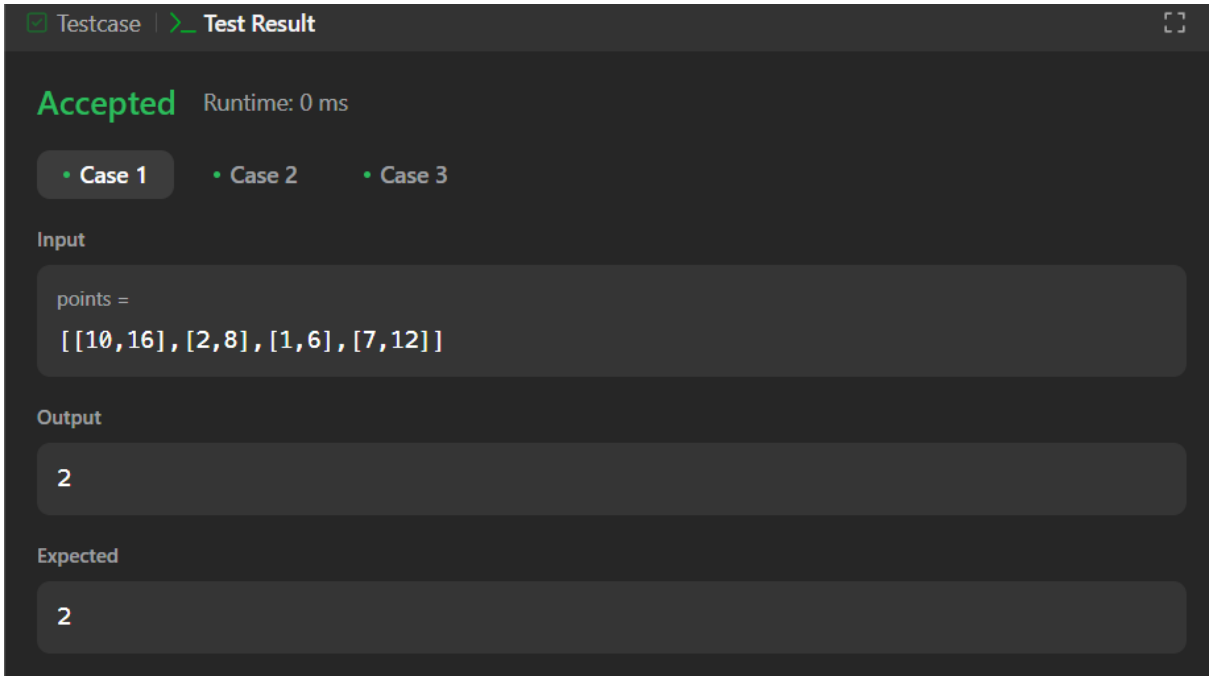
```
class Solution {
public:
    string frequencySort(string s) {
        unordered_map<char,int> mp;
        multimap<int,char> r;
        string ss="";
for(auto a : s)
  mp[a]++;
for(auto a : mp)
  r.insert({a.second, a.first});
for(auto it = r.rbegin(); it != r.rend(); ++it)
ss += string(it->first, it->second);
return ss;
    }
};
```



Ques no.5:- Minimum Number of Arrows to Burst Balloons(452)

Solution :-

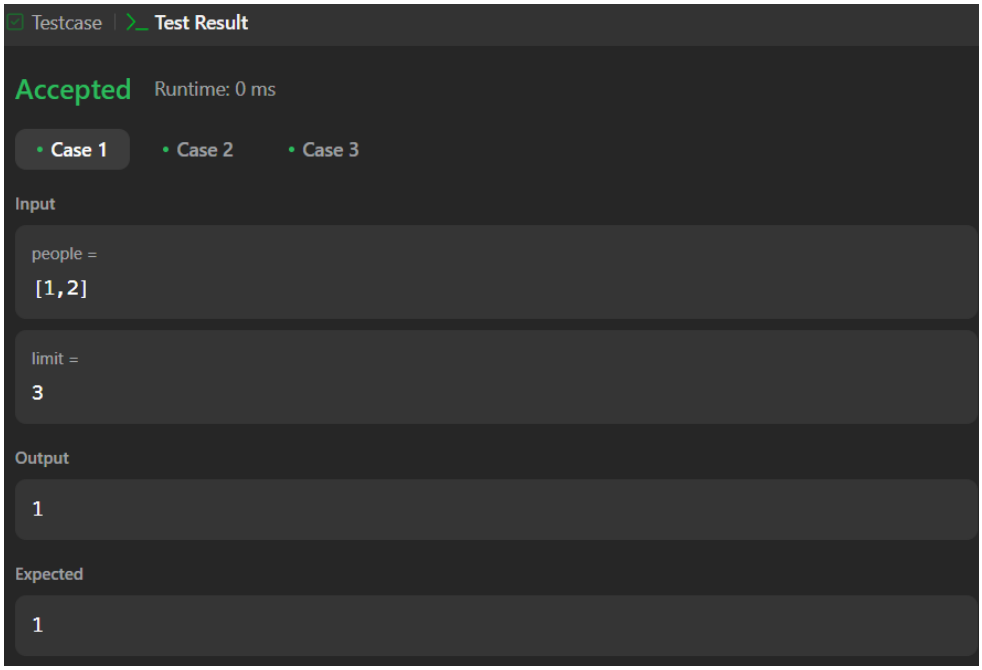
```
class Solution {
public:
    int findMinArrowShots(vector<vector<int>>& p) {
        sort(p.begin(), p.end());
        int lastpoint = p[0][1];
        int ans = 1;
        for(auto point : p) {
            if(point[0] > lastpoint) {
                ans++;
                lastpoint = point[1];
            }
            lastpoint = min(point[1],lastpoint);
        }
        return ans;
    }
};
```



Ques no. 6 :- Boats to Save People (881)

Solution :- class Solution {

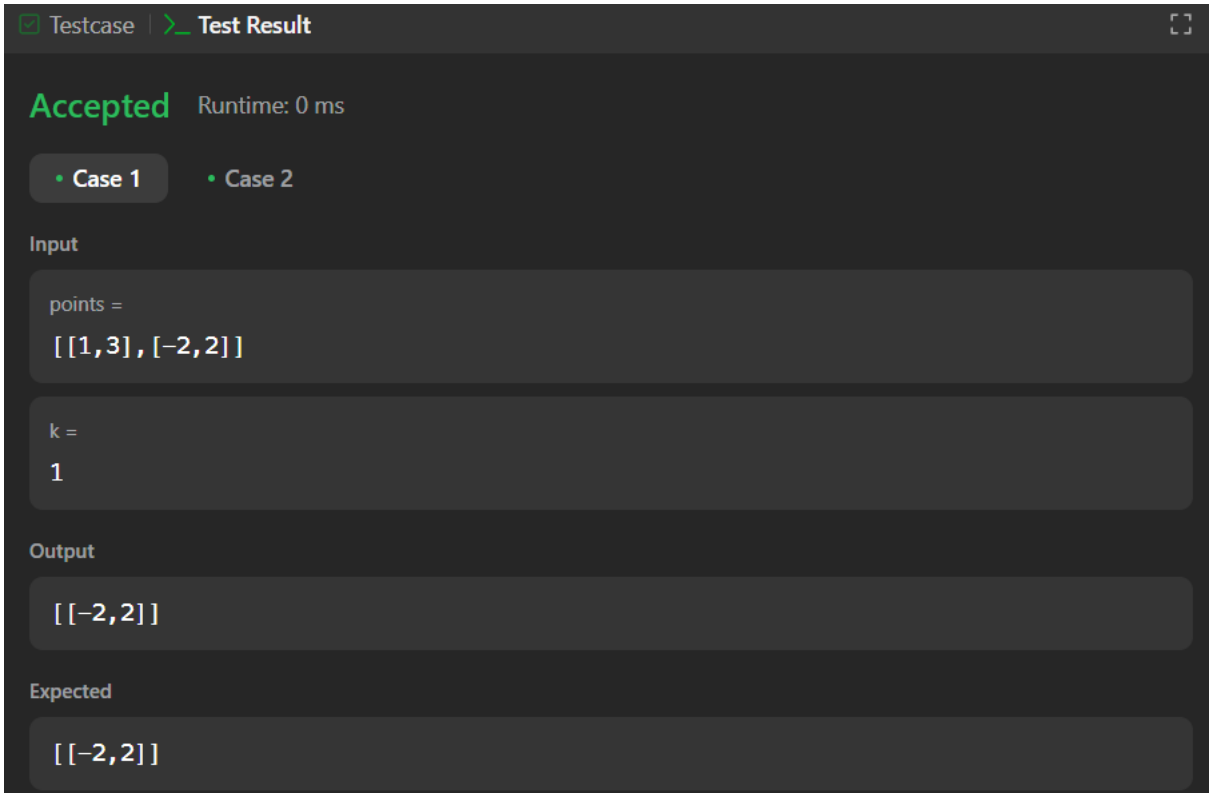
```
public:
    int numRescueBoats(vector<int>& people, int limit) {
        sort(people.begin(),people.end());
        int i = 0, j = people.size() - 1,cnt = 0;
        while(i <= j)
        {
            if(people[i] + people[j] <= limit)
            {
                ++i;
                --j;
            }
            else
                --j;
            ++cnt;
        }
        return cnt;
    }
};
```



Ques no. 7 :- K Closest Points to Origin (973)

Solution :-

```
class Solution {
public:
    vector<vector<int>> kClosest(vector<vector<int>>& points, int k) {
        vector<pair<int,int>> a;
        vector<vector<int>> ans;
        for(int i=0;i<points.size();i++)
a.push_back(make_pair(i,((points[i][0]*points[i][0])+(points[i][1]*points[i][1]))));
        sort(a.begin(),a.end() ,[] (pair<int,int> x,pair<int,int> y){return x.second<y.second;});
        for(auto i=a.begin();i<a.begin()+k;i++) ans.push_back(points[i->first]);
        return ans;}};
```



Ques no. 8 :- Reduce Array Size to The Half(1338)

Solution :- class Solution {

```
public:
    int minSetSize(vector<int>& arr) {
        int n = arr.size();
        unordered_map<int, int> cnt;
        for (int x : arr) ++cnt[x];
vector<int> counting(n + 1);
        for (auto [_, freq] : cnt) ++counting[freq];
int ans = 0, removed = 0, half = n / 2, freq = n;
        while (removed < half) {
            ans += 1;
            while (counting[freq] == 0) --freq;
            removed += freq;
            --counting[freq];
        }return ans;
    } };
```