Assingment -6

Student Name: Subhojit Ghosh   UID: 22BCS15368

Branch: CSE   Section/Group:FL-602-A

Semester: 6   Date of Performance:18 march 25

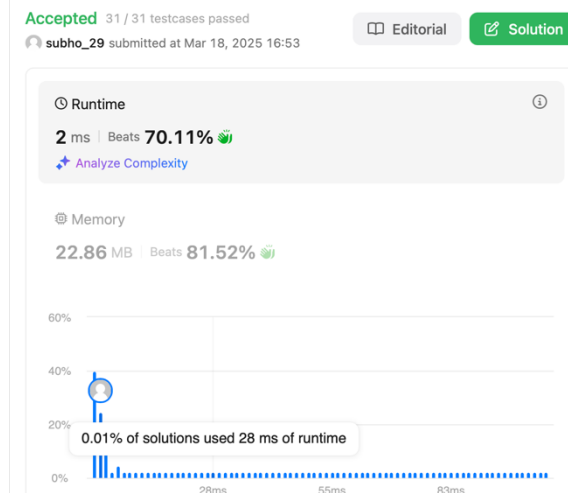**Subject Name: Advanced Programming**   Subject Code: 22CSH-359

1. **Convert Sorted Array to Binary Search Tree**

```cpp
#include <vector>
using namespace std;

class Solution {
public:
    TreeNode* sortedArrayToBST(vector<int>& nums) {
        return helper(nums, 0, nums.size() - 1);
    }

private:
    TreeNode* helper(vector<int>& nums, int left, int right) {
        if (left > right) return nullptr;
        int mid = left + (right - left) / 2;
        TreeNode* root = new TreeNode(nums[mid]);
        root->left = helper(nums, left, mid - 1);
        root->right = helper(nums, mid + 1, right);
        return root;
    }
};
```


Accepted 31 / 31 testcases passed
subho_29 submitted at Mar 18, 2025 16:53

Editorial   Solution

Runtime
2 ms | Beats 70.11%
Analyze Complexity

Memory
22.86 MB | Beats 81.52%

0.01% of solutions used 28 ms of runtime

2. Number of Bits

```cpp
class Solution {
public:
    int hammingWeight(uint32_t n) {
        int res = 0;
        for (int i = 0; i < 32; i++) {
            if ((n >> i) & 1) {
                res += 1;
            }
        }
        return res;
    }
};
```
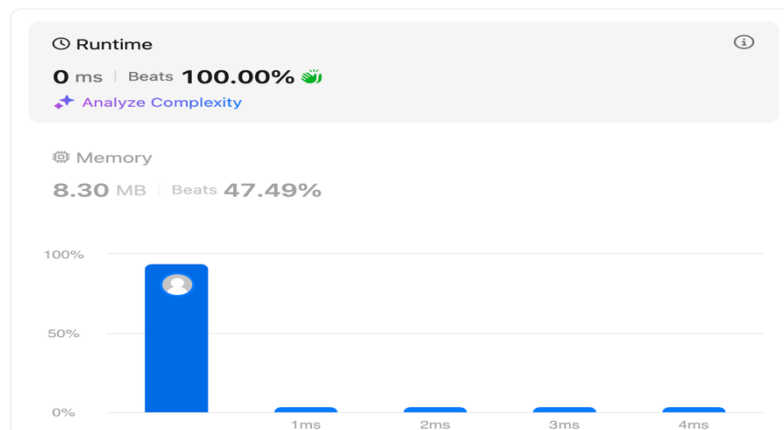
3. Sort an Array

```cpp
class Solution {
public:
int partition(vector <int>& arr,int l,int r){
    if (l >= r) return -1;
    int n=r-l;
    int pivot=l+rand()%n;
    swap(arr[l],arr[pivot]);
    int i=l+1;
    for(int j=l+1;j<r;j++){
        if(arr[j]<arr[l]){
            swap(arr[i],arr[j]);
            i++;
        }
    }
    swap(arr[l],arr[i-1]);
    return i-1;
}
void QuickSort(vector<int>& nums,int l,int r){
```

```
        if(l>=r){
            return;
        }
        int pivot=partition(nums,l,r);
        QuickSort(nums,l,pivot);
        QuickSort(nums,pivot+1,r);
    }
    vector<int> sortArray(vector<int>& nums) {
        int n=nums.size();
        QuickSort(nums,0,n);
        return nums;
    }
};
```

Accepted 21 / 21 testcases passed   📖 Editorial   ✅ Solution
🔵 subho_29 submitted at Mar 18, 2025 17:02

🕐 Runtime   ⓘ
399 ms | Beats **38.80%**
✦ Analyze Complexity

⊕ Memory
185.60 MB | Beats **31.29%**

```
30%
20%
10%
0%
     13ms         563ms         1112ms         1662ms
```
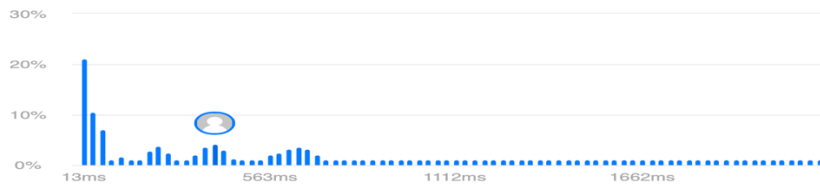
4. <u>Maximum Subarray</u>

```cpp
class Solution {
public:
    int maxSubArray(vector<int>& nums) {
        int res = nums[0];
        int total = 0;

        for (int n : nums) {
            if (total < 0) {
                total = 0;
            }

            total += n;
            res = max(res, total);
        }

        return res;
    }
};
```

100%

75%

50%

25%

0%

1ms   2ms   3ms   4ms   5ms

5. Beautiful Array

```cpp
class Solution {
public:
    int partition(vector<int> &v, int start, int end, int mask)
    {
        int j = start;
        for(int i = start; i <= end; i++)
        {
            if((v[i] & mask) != 0)
            {
                swap(v[i], v[j]);
                j++;
            }
        }
        return j;
    }

    void sort(vector<int> & v, int start, int end, int mask)
    {
        if(start >= end) return;
        int mid = partition(v, start, end, mask);
        sort(v, start, mid - 1, mask << 1);
        sort(v, mid, end, mask << 1);
    }
```

```cpp
    vector<int> beautifulArray(int N) {
        vector<int> ans;
        for(int i = 0; i < N; i++) ans.push_back(i + 1);
        sort(ans, 0, N - 1, 1);
        return ans;
    }
};
```

Accepted 38 / 38 testcases passed

📖 Editorial   ✏️ Solution

subho_29 submitted at Mar 18, 2025 17:10

🕐 Runtime   ⓘ
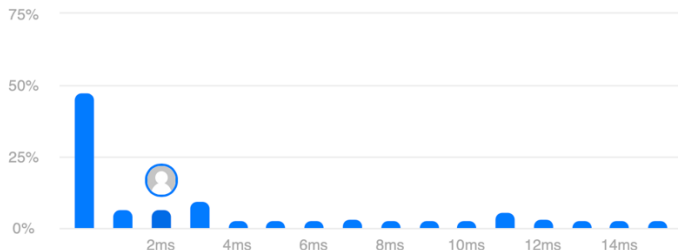
2 ms | Beats **45.73%**

✦ Analyze Complexity

⚙ Memory

9.35 MB | Beats **97.29%** 🤚

✦ Analyze Complexity

6.   Super pow

```cpp
class Solution {
    const int base = 1337;
    int powmod(int a, int k) //a^k mod 1337 where 0 <= k <= 10
    {
        a %= base;
        int result = 1;
        for (int i = 0; i < k; ++i)
            result = (result * a) % base;
        return result;
    }
public:
    int superPow(int a, vector<int>& b) {
        if (b.empty()) return 1;
        int last_digit = b.back();
        b.pop_back();
        return powmod(superPow(a, b), 10) * powmod(a, last_digit) % base;
```

```
        }
    };
```
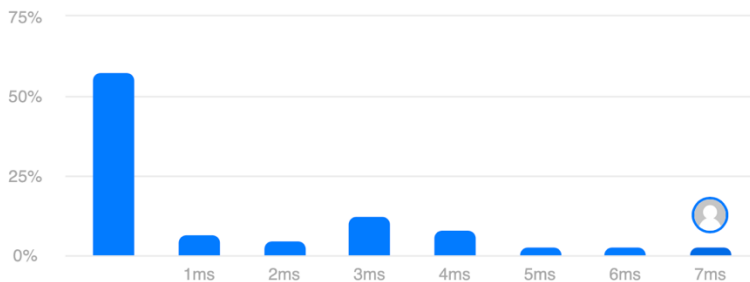
7. [The Skyline Problem](#)

```cpp
class Solution {
public:
    vector<vector<int>> getSkyline(vector<vector<int>>& buildings) {
        vector<vector<int>> ans;
        multiset<int> pq{0};

        vector<pair<int, int>> points;

        for(auto b: buildings){
            points.push_back({b[0], -b[2]});
            points.push_back({b[1], b[2]});
        }

        sort(points.begin(), points.end());

        int ongoingHeight = 0;

        // points.first = x coordinate, points.second = height
        for(int i = 0; i < points.size(); i++){
```

```cpp
            int currentPoint = points[i].first;
            int heightAtCurrentPoint = points[i].second;

            if(heightAtCurrentPoint < 0){
                pq.insert(-heightAtCurrentPoint);
            } else {
                pq.erase(pq.find(heightAtCurrentPoint));
            }

            // after inserting/removing heightAtI, if there's a change
            auto pqTop = *pq.rbegin();
            if(ongoingHeight != pqTop){
                ongoingHeight = pqTop;
                ans.push_back({currentPoint, ongoingHeight});
            }
        }

        return ans;
    }
};
```

Accepted 44 / 44 testcases passed   Editorial   Solution

subho_29 submitted at Mar 18, 2025 17:14

🕐 Runtime

18 ms | Beats **55.11%** 🙌

✦ Analyze Complexity

⚙ Memory

28.95 MB | Beats **42.21%**

Analyze Complexity