Experiment-5

Student Name: Abhinav Paswan **UID:** 22BET10332
Branch: BE-IT **Section/Group:** 22BET-IOT-701(A)
Semester: 6th **Date of Performance:**21-02-2025
Subject Name: AP LAB-II **Subject Code:** 22ITP-351

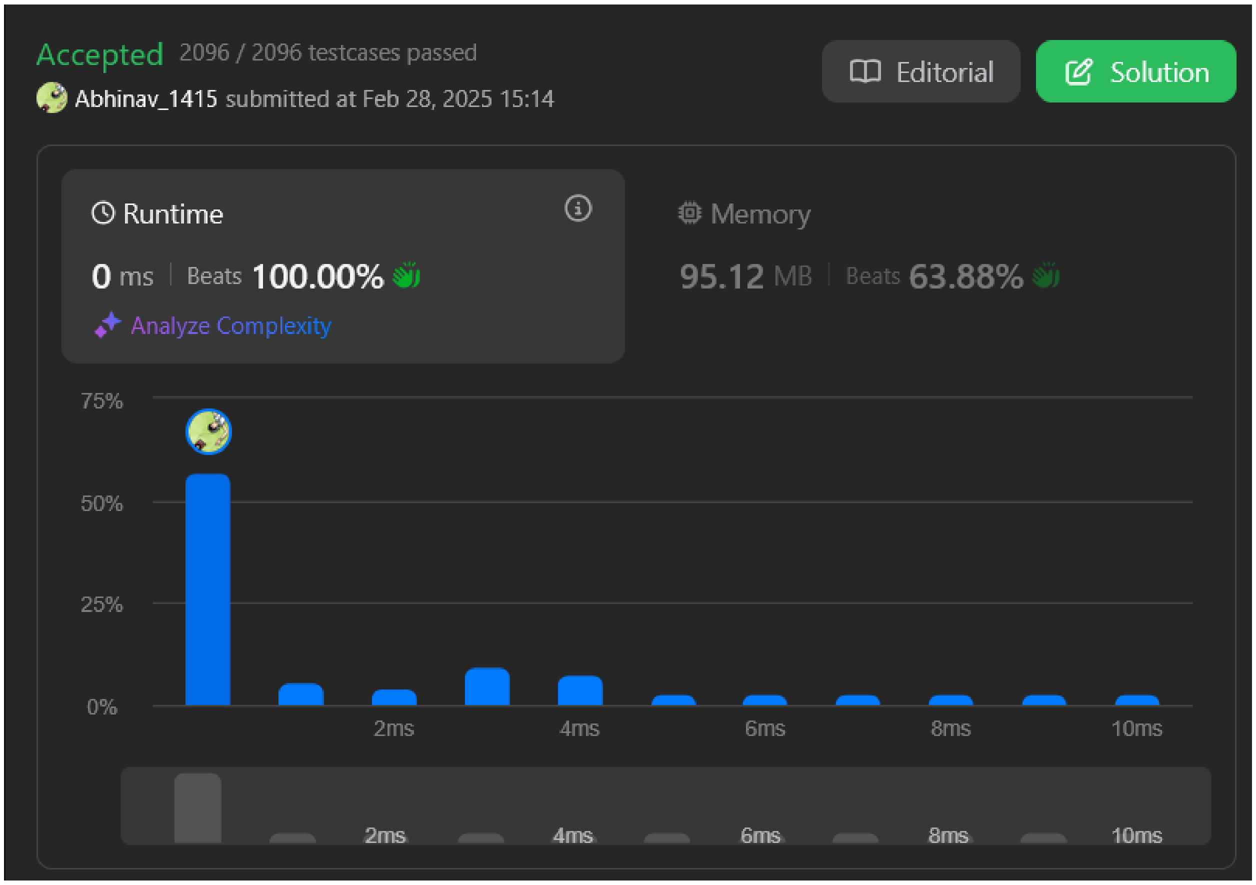
Problem 1:-

https://leetcode.com/problems/median-of-two-sorted-arrays/submissions/1558045850/

Code:

```
class Solution {
public:
    double findMedianSortedArrays(vector<int>& nums1, vector<int>& nums2) {
        if (nums1.size() > nums2.size()) {
        return findMedianSortedArrays(nums2, nums1);
    }

    int m = nums1.size(), n = nums2.size();
    int low = 0, high = m;

    while (low <= high) {
        int partition1 = (low + high) / 2;
        int partition2 = (m + n + 1) / 2 - partition1;

        int maxLeft1 = (partition1 == 0) ? INT_MIN : nums1[partition1 - 1];
        int minRight1 = (partition1 == m) ? INT_MAX : nums1[partition1];

        int maxLeft2 = (partition2 == 0) ? INT_MIN : nums2[partition2 - 1];
        int minRight2 = (partition2 == n) ? INT_MAX : nums2[partition2];

        if (maxLeft1 <= minRight2 && maxLeft2 <= minRight1) {
            if ((m + n) % 2 == 0) {
                return (max(maxLeft1, maxLeft2) + min(minRight1, minRight2)) / 2.0;
            } else {
                return max(maxLeft1, maxLeft2);
            }
        } else if (maxLeft1 > minRight2) {
            high = partition1 - 1;
        } else {
            low = partition1 + 1;
        }
    }
    return -1;
}

};
```

OUTPUT:



Accepted 2096 / 2096 testcases passed
Abhinav_1415 submitted at Feb 28, 2025 15:14

Runtime
0 ms | Beats 100.00%

Memory
95.12 MB | Beats 63.88%

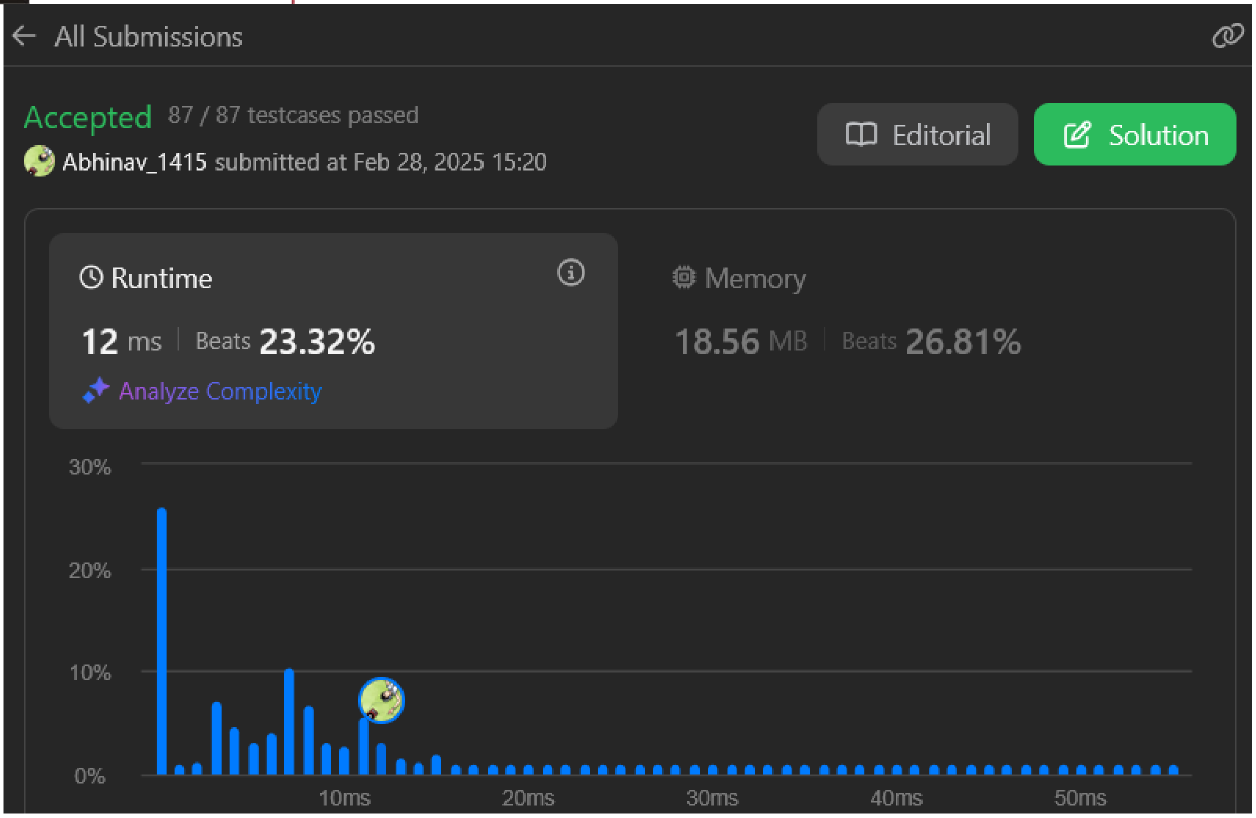
Problem 2 :-

https://leetcode.com/problems/kth-smallest-element-in-a-sorted-matrix/submissions/1558050429/

Code:

```
class Solution {
public:
    int kthSmallest(vector<vector<int>>& matrix, int k) {
    priority_queue<int, vector<int>, greater<int>> minHeap;

    for (int i = 0; i < matrix.size(); i++) {
        for (int j = 0; j < matrix[0].size(); j++) {
            minHeap.push(matrix[i][j]);
        }
    }

    while (k-- > 1) {
        minHeap.pop();
    }

    return minHeap.top();
}
};
```

OUTPUT:


All Submissions

Accepted 87 / 87 testcases passed

Abhinav_1415 submitted at Feb 28, 2025 15:20

Runtime

12 ms | Beats 23.32%

Memory

18.56 MB | Beats 26.81%

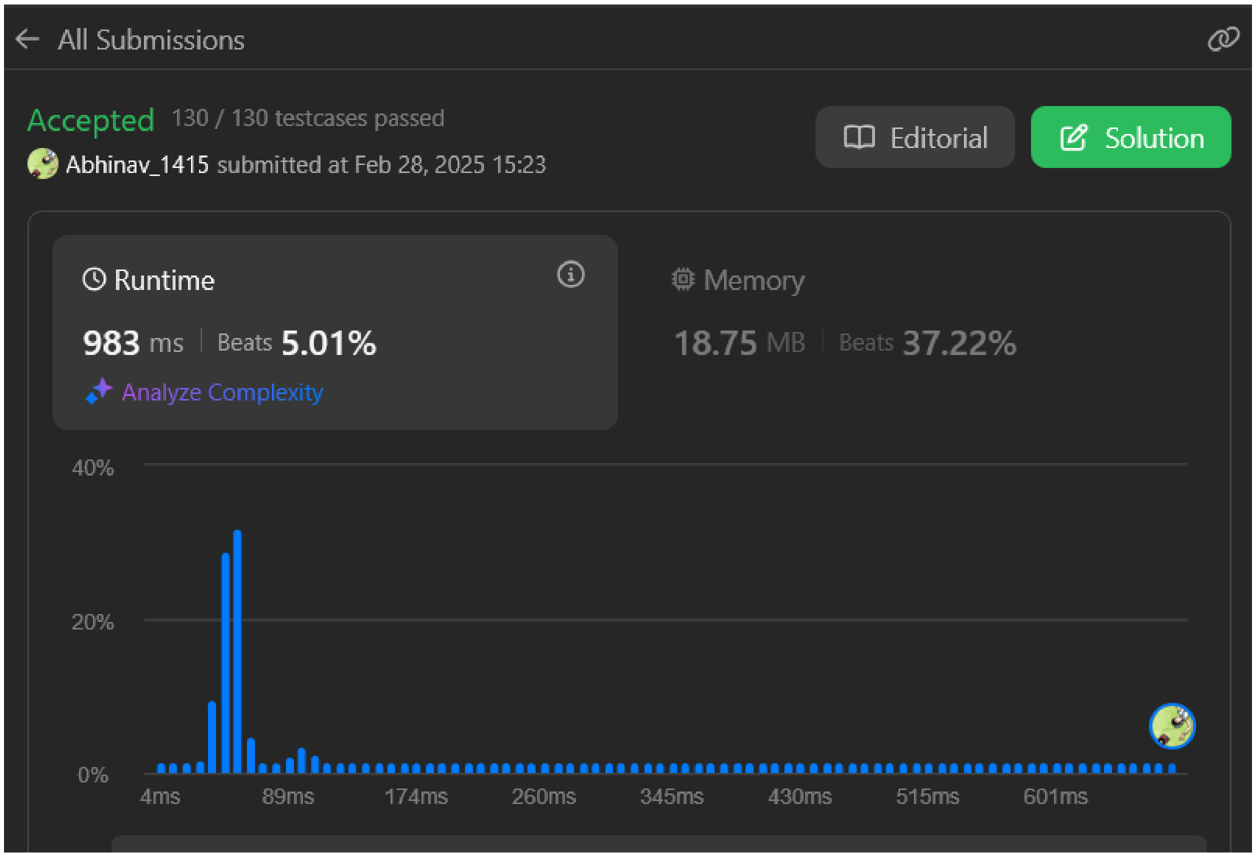
Problem 3:-

https://leetcode.com/problems/search-a-2d-matrix-ii/submissions/1558052452/

Code:

```
class Solution {
public:
    bool searchMatrix(vector<vector<int>>& matrix, int target) {
    for (int i = 0; i < matrix.size(); i++) {
        for (int j = 0; j < matrix[0].size(); j++) {
            if (matrix[i][j] == target) return true;
        }
    }
    return false;
}
};
```

OUTPUT:



All Submissions

Accepted 130 / 130 testcases passed

Abhinav_1415 submitted at Feb 28, 2025 15:23

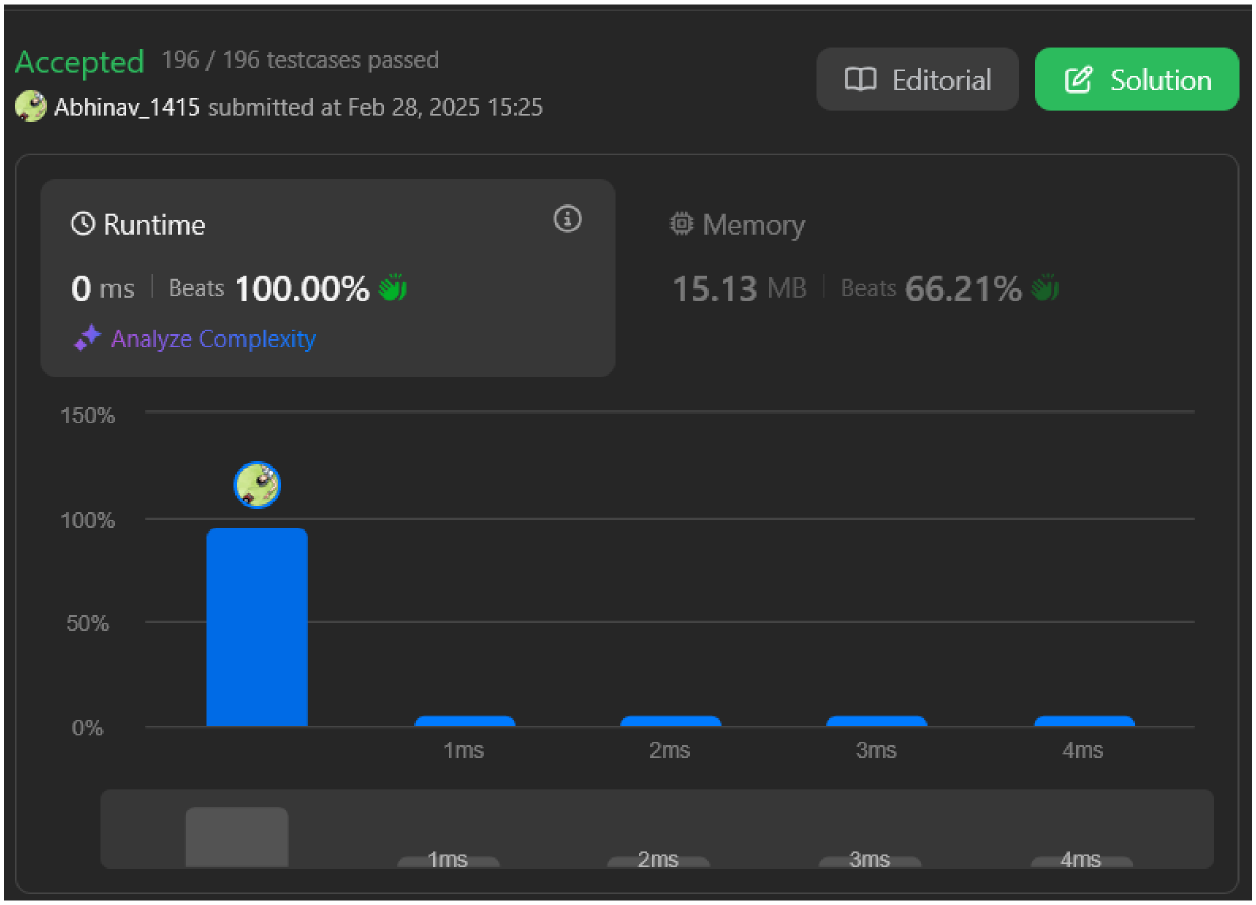
Problem 4:-

https://leetcode.com/problems/search-in-rotated-sorted-array/description/

Code:

```
#include <vector>
using namespace std;
class Solution {
public:
    int search(vector<int>& nums, int target) {
        int n = nums.size();
        int left = 0, right = n - 1;
        while (left < right) {
            int mid = left + (right - left) / 2;
            if (nums[mid] > nums[right])
                left = mid + 1;
            else
                right = mid;
        }
        int pivot = left;
        left = 0, right = n - 1;
        if (target >= nums[pivot] && target <= nums[right])
            left = pivot;
        else
            right = pivot;

        while (left <= right) {
            int mid = left + (right - left) / 2;
            if (nums[mid] == target)
                return mid;
            else if (nums[mid] < target)
                left = mid + 1;
            else
                right = mid - 1;
        }
        return -1;
    }
};
```

OUTPUT:



Accepted 196 / 196 testcases passed

Abhinav_1415 submitted at Feb 28, 2025 15:25

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

15.13 MB | Beats 66.21%

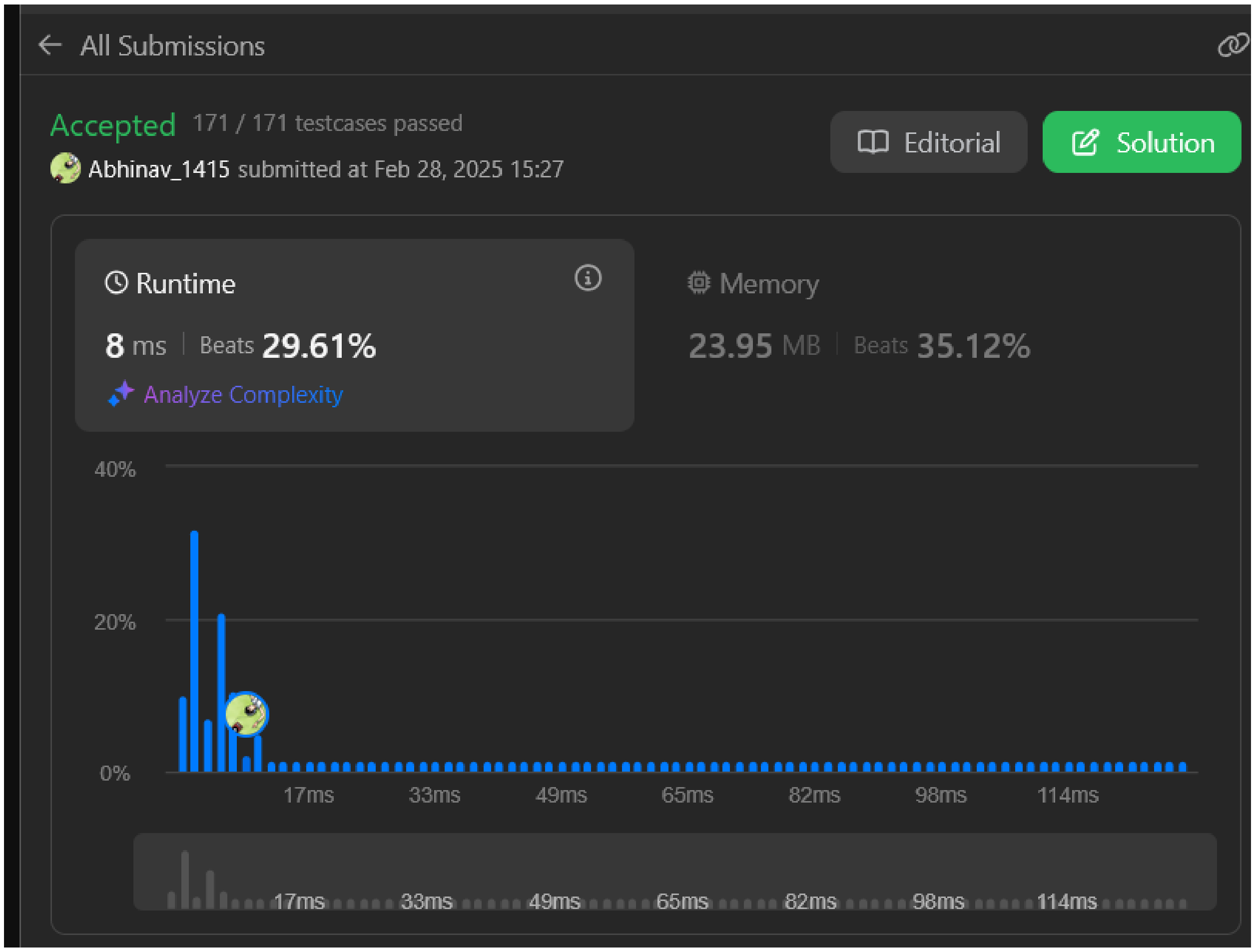
Problem 5:-

https://leetcode.com/problems/merge-intervals/submissions/1558055228/

Code:-

```
class Solution {
public:
    vector<vector<int>
>
merge(vector<vector
<int>>& intervals) {
        if
(intervals.empty())
return {};

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

    vector<vector<int>
> merged;

    for (auto& interval
: intervals) {
        if
(merged.empty() ||
merged.back()[1] <
interval[0]) {
            merged.push_
back(interval);
        }
        else {
            merged.back()
[1] =
max(merged.back()[1
], interval[1]);
        }
    }

    return merged;
    }
};
```

OUTPUT:



All Submissions

Accepted 171 / 171 testcases passed

Abhinav_1415 submitted at Feb 28, 2025 15:27

Runtime

8 ms | Beats 29.61%

Memory

23.95 MB | Beats 35.12%