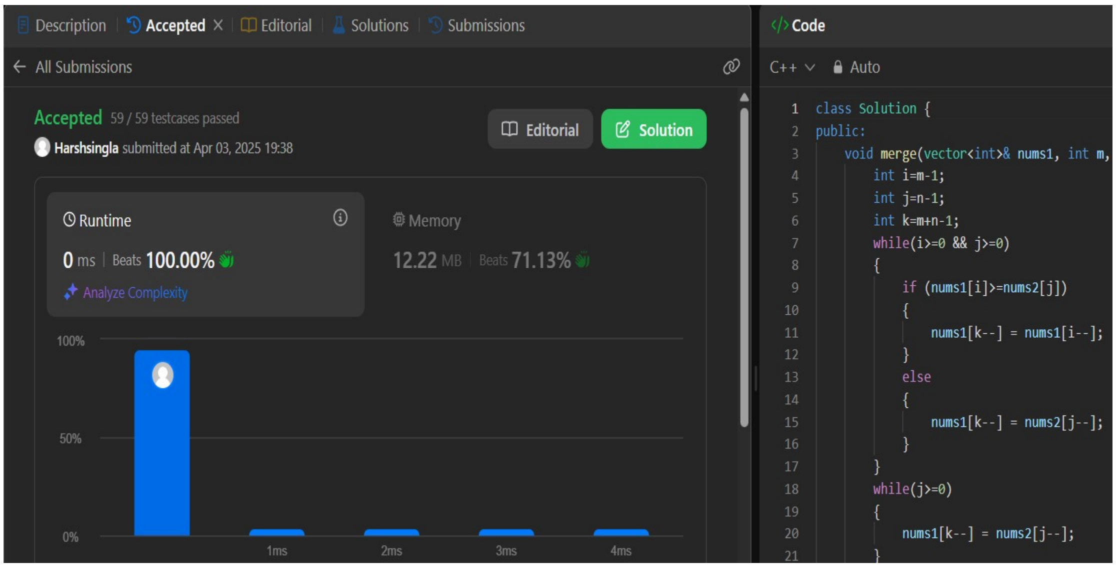
Assignment - 6

Name – Harsh Singla

UID – 22BCS15757

Section – 608-B

1. Merge Sorted Array



Accepted 59 / 59 testcases passed

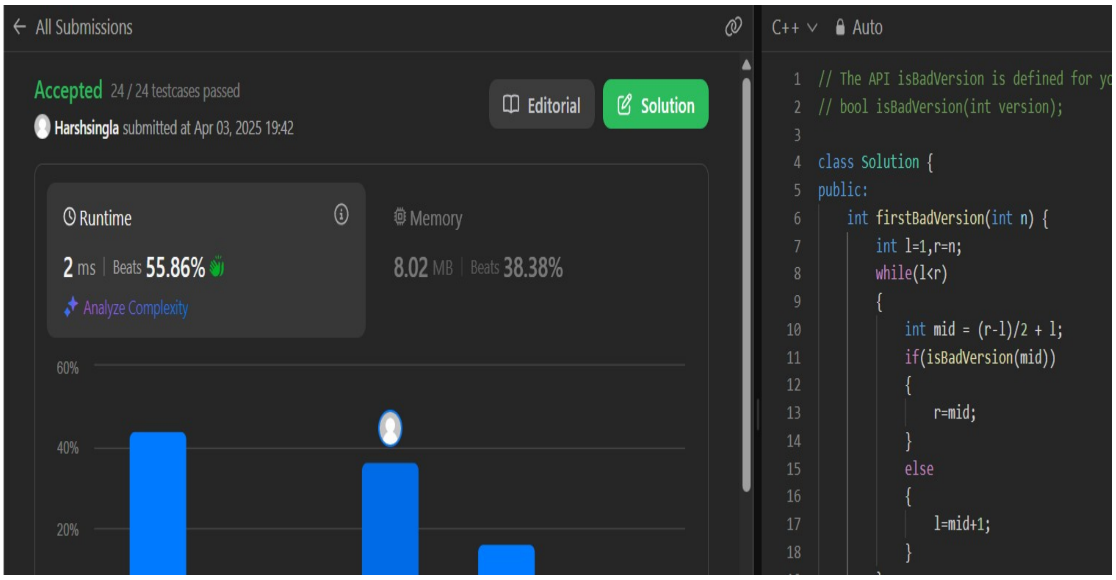
Harshsingla submitted at Apr 03, 2025 19:38

Runtime
0 ms | Beats 100.00%

Memory
12.22 MB | Beats 71.13%

```
class Solution {
public:
    void merge(vector<int>& nums1, int m,
        int i=m-1;
        int j=n-1;
        int k=m+n-1;
        while(i>=0 && j>=0)
        {
            if (nums1[i]>=nums2[j])
            {
                nums1[k--] = nums1[i--];
            }
            else
            {
                nums1[k--] = nums2[j--];
            }
        }
        while(j>=0)
        {
            nums1[k--] = nums2[j--];
        }
```

2. First Bad Version



Accepted 24 / 24 testcases passed

Harshsingla submitted at Apr 03, 2025 19:42

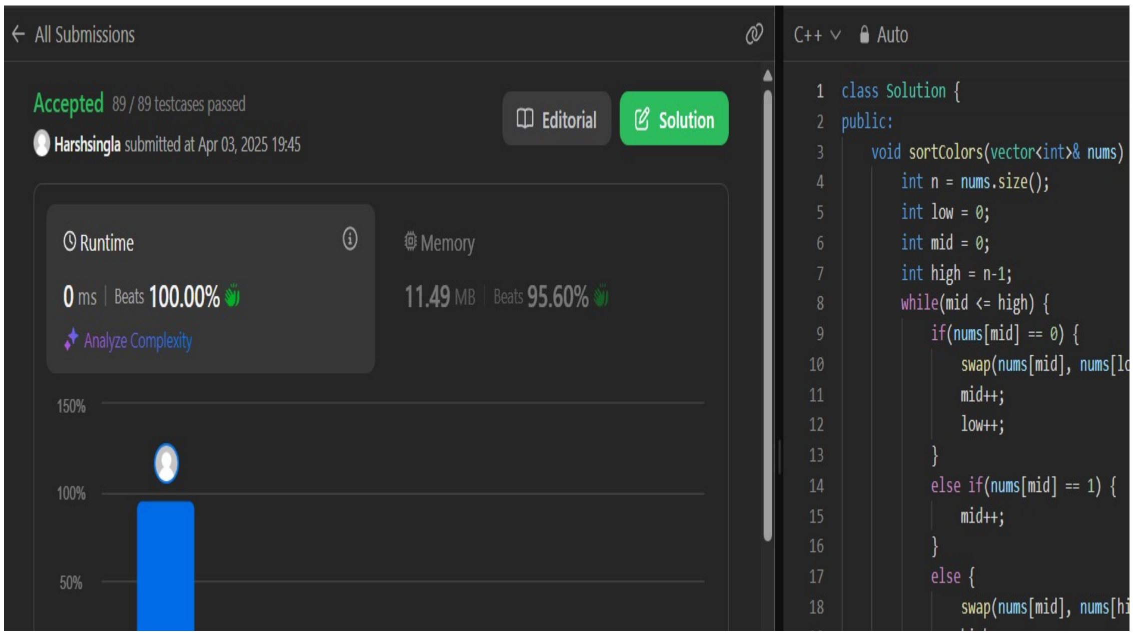
Runtime
2 ms | Beats 55.86%

Memory
8.02 MB | Beats 38.38%

```
// The API isBadVersion is defined for yo
// bool isBadVersion(int version);

class Solution {
public:
    int firstBadVersion(int n) {
        int l=1,r=n;
        while(l<r)
        {
            int mid = (r-l)/2 + l;
            if(isBadVersion(mid))
            {
                r=mid;
            }
            else
            {
                l=mid+1;
            }
```

3. Sort Colors



Accepted 89 / 89 testcases passed

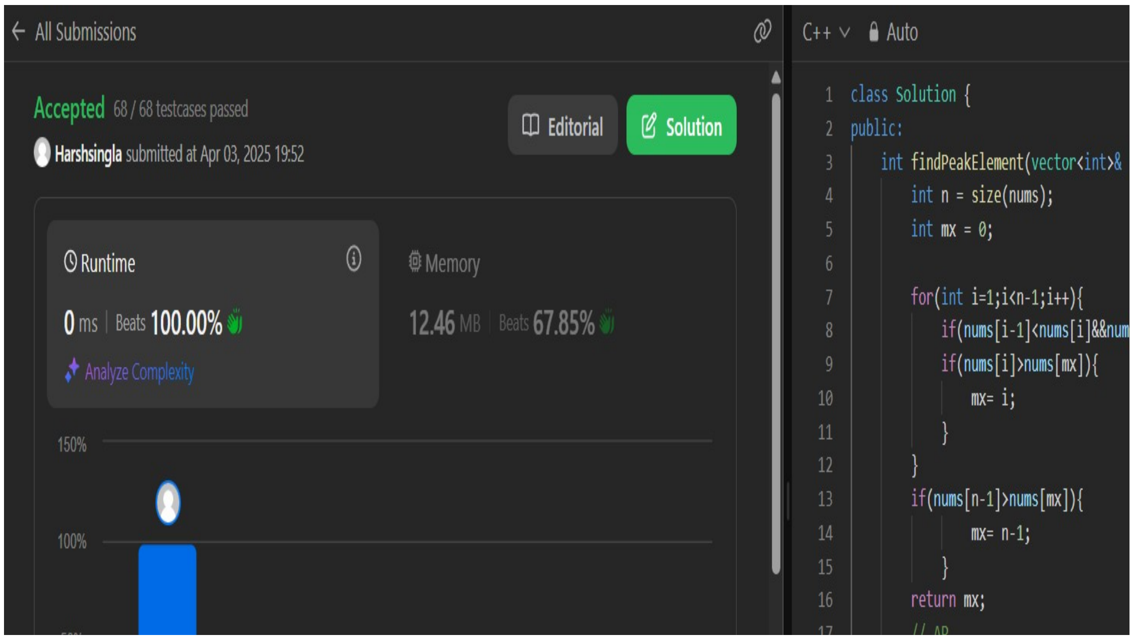
Harshsingla submitted at Apr 03, 2025 19:45

Runtime 0 ms | Beats 100.00%

Memory 11.49 MB | Beats 95.60%

```
class Solution {
public:
    void sortColors(vector<int>& nums)
        int n = nums.size();
        int low = 0;
        int mid = 0;
        int high = n-1;
        while(mid <= high) {
            if(nums[mid] == 0) {
                swap(nums[mid], nums[lo
                mid++;
                low++;
            }
            else if(nums[mid] == 1) {
                mid++;
            }
            else {
                swap(nums[mid], nums[hi
```

4. Find Peak Element



Accepted 68 / 68 testcases passed

Harshsingla submitted at Apr 03, 2025 19:52

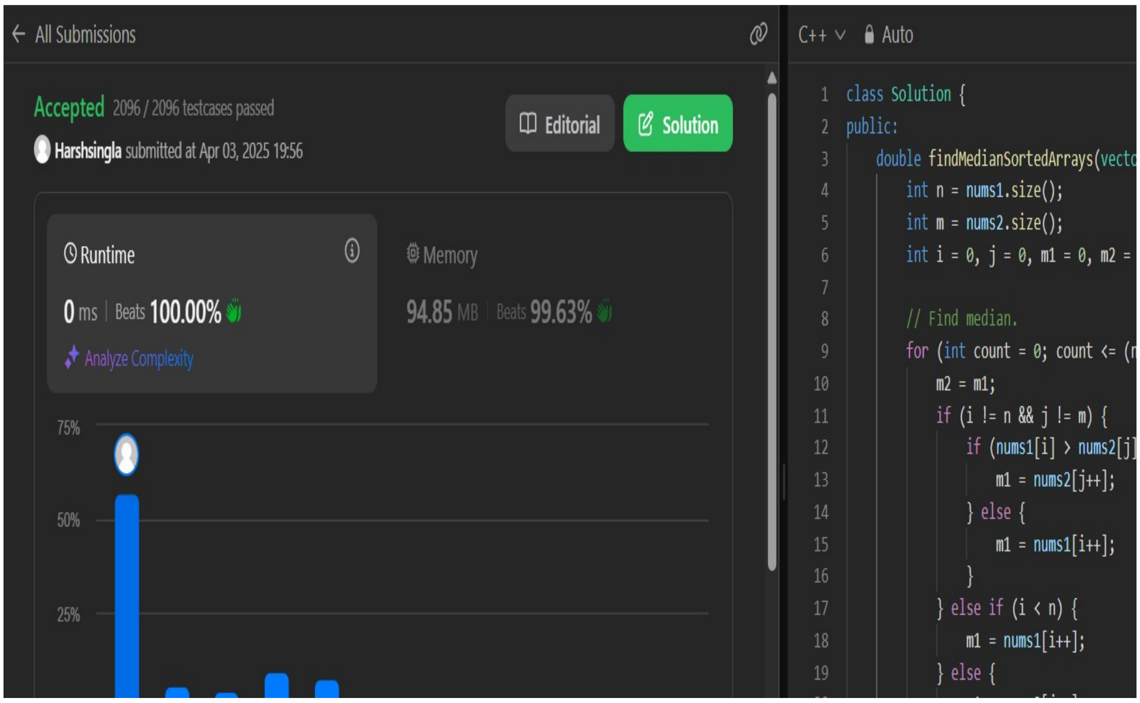
Runtime 0 ms | Beats 100.00%

Memory 12.46 MB | Beats 67.85%

```
class Solution {
public:
    int findPeakElement(vector<int>&
        int n = size(nums);
        int mx = 0;

        for(int i=1;i<n-1;i++){
            if(nums[i-1]<nums[i]&&num
            if(nums[i]>nums[mx]){
                mx= i;
            }
        }
        if(nums[n-1]>nums[mx]){
                mx= n-1;
            }
        return mx;
```

5. Median Of Two Sorted Arrays



All Submissions

Accepted 2096 / 2096 testcases passed

Harshsingla submitted at Apr 03, 2025 19:56

Runtime

0 ms | Beats 100.00%

Memory

94.85 MB | Beats 99.63%

```
class Solution {
public:
    double findMedianSortedArrays(vecto
        int n = nums1.size();
        int m = nums2.size();
        int i = 0, j = 0, m1 = 0, m2 =

        // Find median.
        for (int count = 0; count <= (n
            m2 = m1;
            if (i != n && j != m) {
                if (nums1[i] > nums2[j]
                    m1 = nums2[j++];
                } else {
                    m1 = nums1[i++];
                }
            } else if (i < n) {
                m1 = nums1[i++];
            } else {
```