

**CHANDIGARH
UNIVERSITY**
Discover. Learn. Empower.

**NAAC
GRADE A+**
Accredited University

UNIVERSITY INSTITUTE OF ENGINEERING

Department of Computer Science & Engineering

(BE-CSE/IT-6th Sem)



Subject Name: Advanced Programming Lab - 2

Subject Code: 22CSP-351

Submitted to:

Mr. Vishal

Submitted by:

Name: Sanya Singh

UID: 23BCS14374

Section: FL_IOT_604

Group: A

1763. Longest Nice String

Accepted 73 / 73 testcases passed
Sanya Singh submitted at Feb 05, 2025 20:46

Runtime
22 ms | Beats 26.88%
[Analyze Complexity](#)

Memory
14.52 MB | Beats 28.42%

Code

```
1 class Solution {
2 public:
3     bool check(string s){
4         for(int i=0;i<s.size();i++) {
5             char c = s[i];
6             if(c<=90) {
```

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

s =
"YazaAay"

Output

"aAa"

Expected

190. Reverse Bits

Accepted 600 / 600 testcases passed
San... submitted at Feb 05, 2025 20:55

Runtime
0 ms | Beats 100.00%
[Analyze Complexity](#)

Memory
7.79 MB | Beats 63.18%

Code

```
1 class Solution {
2 public:
3     uint32_t reverseBits(uint32_t n) {
4         uint32_t res = 0;
```

Testcase | Test Result

Accepted Runtime: 3 ms

Case 1 Case 2

Input

n =
00000010100101000001111010011100

Output

964176192 (0011100101111000001010101000000)

Expected

964176192 (0011100101111000001010101000000)

191. Number of 1 Bits

The screenshot shows a coding platform interface for the problem "191. Number of 1 Bits". The left sidebar displays the submission status as "Accepted" with 598 / 598 testcases passed, submitted on Feb 05, 2025 at 20:59. The runtime is 0 ms, beating 100.00% of other solutions. The memory usage is 8.19 MB, beating 80.68%. A bar chart shows the runtime distribution across different time intervals. The main editor shows the following C++ code:

```
1 class Solution {
2 public:
3     int hammingWeight(int n) {
4         int count = 0;
```

The right sidebar shows the "Test Result" section, indicating the solution is "Accepted" with a runtime of 0 ms. It lists three test cases: Case 1, Case 2, and Case 3. The input for Case 1 is n = 11, and the output is 3, which matches the expected result.

53. Maximum Subarray

The screenshot shows a coding platform interface for the problem "53. Maximum Subarray". The left sidebar displays the submission status as "Accepted" with 210 / 210 testcases passed, submitted on Nov 24, 2024 at 20:06. The runtime is 0 ms, beating 100.00% of other solutions. The memory usage is 70.65 MB, beating 99.93% of other solutions. A bar chart shows the runtime distribution across different time intervals. The main editor shows the following C++ code:

```
1 class Solution {
2 public:
3     int maxSubArray(vector<int>& nums) {
4         int sum=0;
5         int maxi=nums[0];
6         int n=nums.size();
7         for(int i=0;i<n;i++){
8             sum=sum+nums[i];
9             maxi=max(sum,maxi);
10            if(sum<0){
11                sum=0;
12            }
```

The right sidebar shows the "Test Result" section, indicating the solution is "Accepted" with a runtime of 0 ms. It lists three test cases: Case 1, Case 2, and Case 3. The input for Case 1 is nums = [-2,1,-3,4,-1,2,1,-5,4], and the output is 4, which matches the expected result.

240. Search A 2D matrix II

Accepted 130 / 130 testcases passed
San... submitted at Jun 08, 2024 01:38

Runtime
57 ms | Beats 39.50%
[Analyze Complexity](#)

Memory
17.37 MB | Beats 99.96%

Code

```
1 class Solution {
2 public:
3     bool searchMatrix(vector<vector<int>>& matrix, int target) {
4         int row=matrix.size();
5         int col=matrix[0].size();
6
7         int rowindex=0;
8         int colindex=col-1;
9
10        while(rowindex<row && colindex>=0){
11            int element=matrix[rowindex][colindex];
12        }
```

Testcase | **Test Result**

Accepted Runtime: 6 ms

Case 1 Case 2

Input

matrix =
[[1, 4, 7, 11, 15], [2, 5, 8, 12, 19], [3, 6, 9, 16, 22], [10, 13, 14, 17, 24], [18, 21, 23, 26, 30]]

372. Super Pow

Accepted 57 / 57 testcases passed
Sanya Singh submitted at Feb 05, 2025 21:02

Runtime
3 ms | Beats 32.16%
[Analyze Complexity](#)

Memory
15.25 MB | Beats 52.19%

Code

```
12
13     int superPow(int a, vector<int>& b) {
14         if (b.empty()) return 1;
15         int last_digit = b.back();
16         b.pop_back();
17         return powmod(superPow(a, b), 10) * powmod(a, last_digit) % base;
18     }
19 ;
```

Testcase | **Test Result**

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

a =
2

b =
[3]

Output

932. [Beautiful Array](#)

Accepted 38 / 38 testcases passed
San... submitted at Feb 05, 2025 21:04

Runtime
0 ms | Beats 100.00%
[Analyze Complexity](#)

Memory
9.33 MB | Beats 98.74%

Code

```
1 class Solution {
2 public:
3     int partition(vector<int> &v, int start, int end, int mask)
4     {
5         int j = start;
6         for(int i = start; i <= end; i++)
7         {
8             if((v[i] & mask) == 0)
9             {
10                 j = i;
11                 break;
12             }
13         }
14         if(start > end) return 1;
15         if(start == end) return 1;
16         return partition(v, start, j-1, mask) + partition(v, j, end, mask);
17     }
18 }
```

Testcase | **Test Result**

Accepted Runtime: 0 ms

Case 1 Case 2

Input

n =
4

Output

[3,1,2,4]

218. [The Skyline Problem](#)

Accepted 44 / 44 testcases passed
San... submitted at Feb 05, 2025 21:05

Runtime
24 ms | Beats 37.83%
[Analyze Complexity](#)

Memory
28.82 MB | Beats 45.75%

Code

```
6 vector<pair<int, int>> points;
7
8
9 for(auto b: buildings){
10     points.push_back({b[0], -b[2]});
11     points.push_back({b[1], b[2]});
12 }
13
```

Testcase | **Test Result**

Accepted Runtime: 0 ms

Case 1 Case 2

Input

buildings =
[[2,9,10], [3,7,15], [5,12,12], [15,20,10], [19,24,8]]

Output

[[2,10], [3,15], [7,12], [12,0], [15,10], [20,8], [24,0]]

493. [Reverse Pairs](#)

Problem List < > ✕

Description Editorial Solutions Accepted ✕

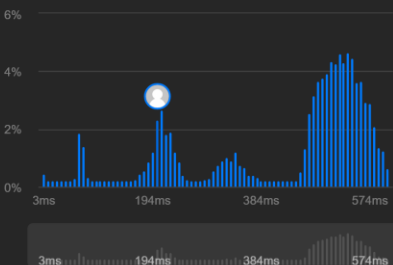
All Submissions

Accepted 140 / 140 testcases passed

San... submitted at Feb 05, 2025 21:06

Runtime 203 ms | Beats 89.78%

Memory 112.76 MB | Beats 88.26%



3ms 194ms 384ms 574ms

Code

```
C++
42 }
43 public:
44     int reversePairs(vector<int>& nums) {
45         int reversePairsCount = 0;
46         mergeSort(nums, 0, nums.size()-1, reversePairsCount);
47         return reversePairsCount;
48     }
49 };
```

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

nums =

[1,3,2,3,1]

Output

2

Expected

2407. Longest Increasing Subsequence II

Description Editorial Solutions Accepted ✕

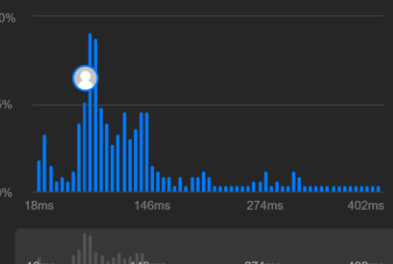
All Submissions

Accepted 84 / 84 testcases passed

Sanya Singh submitted at Feb 05, 2025 21:07

Runtime 70 ms | Beats 85.41%

Memory 59.89 MB | Beats 80.24%



18ms 146ms 274ms 402ms

Code

```
C++
35     int x=query(1,0,m,l,r);
36     if(x!=-1e9) x=0;
37     update(1,0,m,nums[i],x+1);
38 }
39 return tree[1];
40 }
41 };
```

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

nums =

[4,2,1,4,3,4,5,8,15]

k =

3

Output

88. Merge Sorted Array

Problem List < >

Editorial Solutions Submissions Accepted

All Submissions

Accepted 59 / 59 testcases passed

San... submitted at Jun 05, 2024 23:05

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

10.86 MB | Beats 99.94%

100% 50% 0%

1ms 2ms 3ms 4ms

Code

C++ Auto

```

1 class Solution {
2 public:
3     void mergeSort(int* nums1, int n, int* nums2, int m) {

```

Saved

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

nums1 =

[1,2,3,0,0,0]

m =

3

nums2 =

[2,5,6]

n =

278.[First Bad Version](#)

Description Editorial Solutions Accepted

All Submissions

Accepted 24 / 24 testcases passed

San... submitted at Feb 05, 2025 21:09

Runtime

2 ms | Beats 54.93%

Analyze Complexity

Memory

7.92 MB | Beats 38.30%

60% 40% 20% 0%

1ms 2ms 3ms 4ms

Code

C++ Auto

```

> class Solution {
6 public:
7     int firstBadVersion(int n) {
8         int low=1;
9         int high=n;
10        while(low<=high)
11        {
12            int mid=low+(high-low)/2;

```

Saved

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

n =

5

bad =

4

Output

75.[Sort Colors](#)

Editorial

Solutions

Submissions

Accepted

All Submissions

Accepted

88 / 88 testcases passed

San... submitted at Oct 19, 2024 23:09

Solution

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

10.78 MB | Beats 99.98%

Code

```

1 class Solution {
2 public:
3     void sortColors(vector<int>& nums) {
4         map<int,int> mp;
5         for(int i=0;i<nums.size();i++){
6             mp[nums[i]]++;
7         }
8     }
9 }

```

Testcase

Test Result

Case 1 Case 2 +

nums =

[2,0,2,1,1,0]

347. [Top K Frequent Elements](#)

Problem List

<

>

↔

Description

Editorial

Solutions

Accepted

All Submissions

Accepted

21 / 21 testcases passed

San... submitted at Feb 05, 2025 21:13

Solution

Runtime

7 ms | Beats 19.66%

Analyze Complexity

Memory

19.93 MB | Beats 6.42%

Code

```

1 class Solution {
2 public:
3     vector<int> topKFrequent(vector<int>& nums, int k) {
4         int n = nums.size();
5         unordered_map<int, int> map;
6         vector<int> ans;
7         for (int &x : nums) map[x]++;
8         vector<vector<int>> ans2(n+1);
9     }

```

Testcase

Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

nums =

[1,1,1,2,2,3]

k =

2

Output

215. [Kth Largest Element in an Array](#)

Problem List < > > >

Description | Editorial | Solutions | Accepted x

All Submissions

Accepted 42 / 42 testcases passed

San... submitted at Feb 05, 2025 21:11

Runtime

40 ms | Beats 26.37%

Analyze Complexity

Memory

64.75 MB | Beats 9.85%

15% 10% 5% 0%

1ms 78ms 155ms 232ms

Code

```
C++
6      pq.push(num);
7      }
8
9      for(int i=0;i<k-1;i++){
10         pq.pop();
11     }
12     return pq.top();
13 }
```

Saved

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

nums =

[3,2,1,5,6,4]

k =

2

162. [Find Peak Element](#)

Problem List < > > >

Description | Editorial | Solutions | Accepted x

All Submissions

Accepted 68 / 68 testcases passed

San... submitted at Feb 05, 2025 21:14

Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

12.45 MB | Beats 65.57%

150% 100% 50% 0%

1ms 2ms 3ms 4ms

Code

```
C++
1 class Solution {
2 public:
3     int findPeakElement(vector<int>& nums) {
4         int left = 0;
5         int right = nums.size() - 1;
6
7         while (left < right) {
8             int mid = (left + (right - left) / 2);
9         }
10    }
```

Saved

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

nums =

[1,2,3,1]

Output

2

56. [Merge Intervals](#)

Accepted 171 / 171 testcases passed
 Submitted at Feb 05, 2025 21:16

Runtime
 8 ms | Beats 29.41%

Memory
 24.90 MB | Beats 15.27%

```

class Solution {
public:
    vector<vector<int>> merge(vector<vector<int>>& intervals) {
        sort(intervals.begin(), intervals.end(), [](const vector<int>& a, const vector<int>& b) {
            return a[0] < b[0];
        });
    }
};
  
```

Testcase | **Test Result**
Accepted Runtime: 0 ms

Case 1 | Case 2

Input
 intervals =
 [[1,3],[2,6],[8,10],[15,18]]

Output
 [[1,6],[8,10],[15,18]]

33. [Search in Rotated Sorted Array](#)

Accepted 195 / 195 testcases passed
 Submitted at May 30, 2024 23:11

Runtime
 0 ms | Beats 100.00%

Memory
 13.36 MB | Beats 100.00%

```

class Solution {
public:
    int pivot(vector<int>&nums){
        int n=nums.size();
        int f=0;
        int l=n-1;
    }
};
  
```

Testcase | **Test Result**
Case 1 | Case 2 | Case 3 | +

nums =
 [4,5,6,7,0,1,2]

target =
 0

240. [Search a 2D Matrix II](#)

Problem List < >

Editorial Solutions Submissions Accepted

All Submissions

Accepted 130 / 130 testcases passed

San... submitted at Jun 08, 2024 01:38

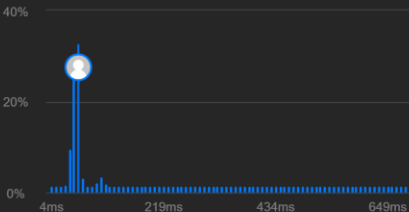
Runtime

57 ms | Beats 39.50%

Analyze Complexity

Memory

17.37 MB | Beats 99.96%



4ms 219ms 434ms 649ms

Code

```

1 class Solution {
2 public:
3     bool searchMatrix(vector<vector<int>>& matrix, int target) {
4         int row=matrix.size();
5         int col=matrix[0].size();
6
7         int rowindex=0;
8         int colindex=col-1;

```

Saved

Testcase Test Result

Case 1 Case 2 +

matrix =

```
[[1,4,7,11,15],[2,5,8,12,19],[3,6,9,16,22],[10,13,14,17,24],[18,21,23,26,30]]
```

target =

5

324. [Wiggle Sort II](#)

Problem List < >

Description Editorial Solutions Accepted

All Submissions

Accepted 52 / 52 testcases passed

Sanya Singh submitted at Feb 05, 2025 21:17

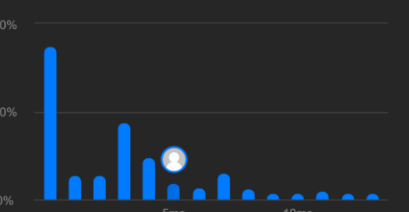
Runtime

5 ms | Beats 26.78%

Analyze Complexity

Memory

21.73 MB | Beats 83.17%



5ms 10ms

Code

```

1 class Solution {
2 public:
3     void wiggleSort(vector<int>& nums) {
4         int n = nums.size();
5         vector<int> nums1(nums);
6         sort(nums1.begin(), nums1.end());
7         int i = n-1;
8         int j = 0;

```

Saved

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

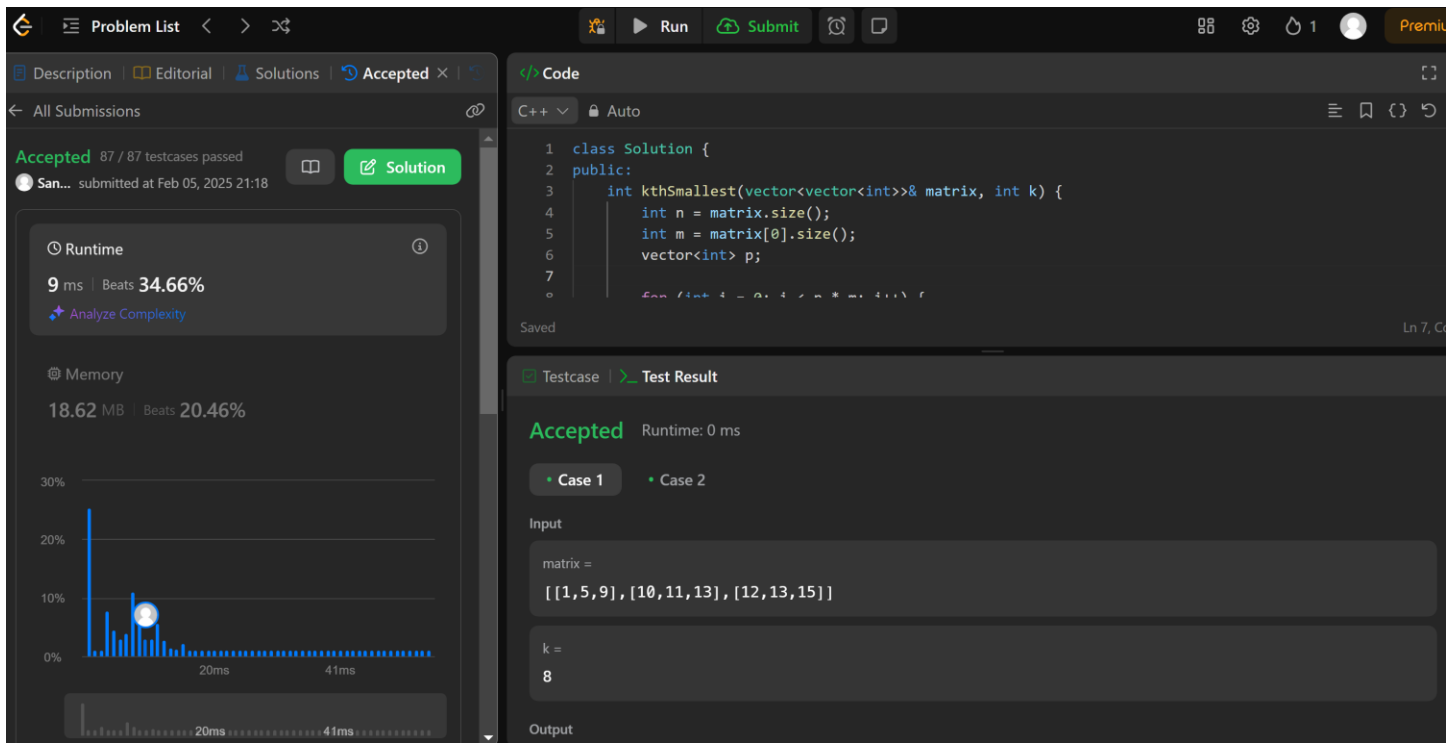
nums =

```
[1,5,1,1,6,4]
```

Output

```
[1,6,1,5,1,4]
```

378. [Kth Smallest Element in a Sorted Matrix](#)



4. [Median of Two Sorted Arrays](#)

