

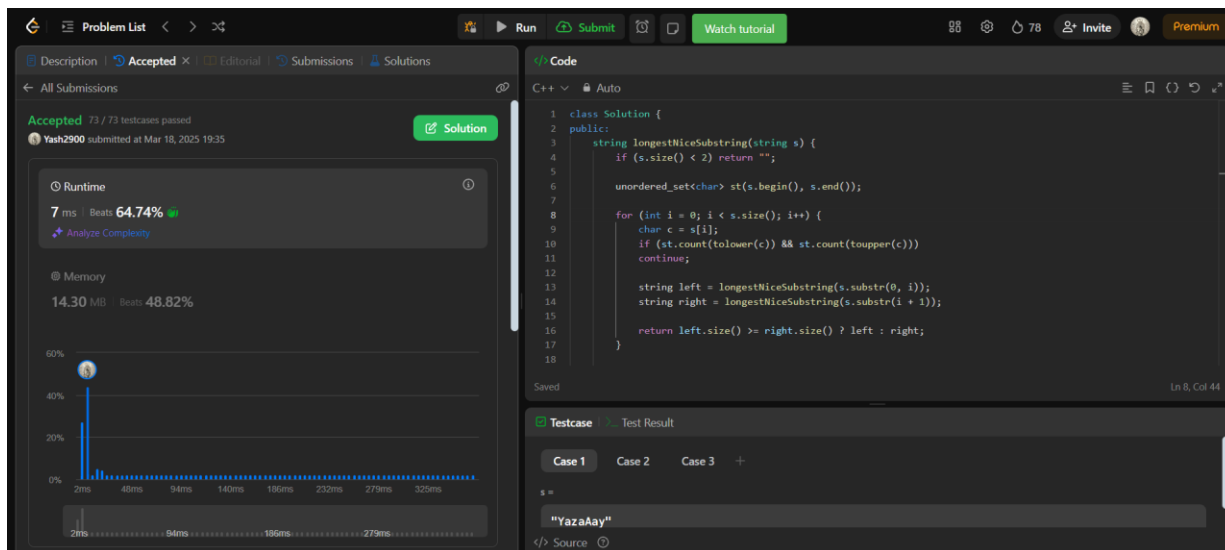
NAME – YASH GARG

UID – 22BCS13420

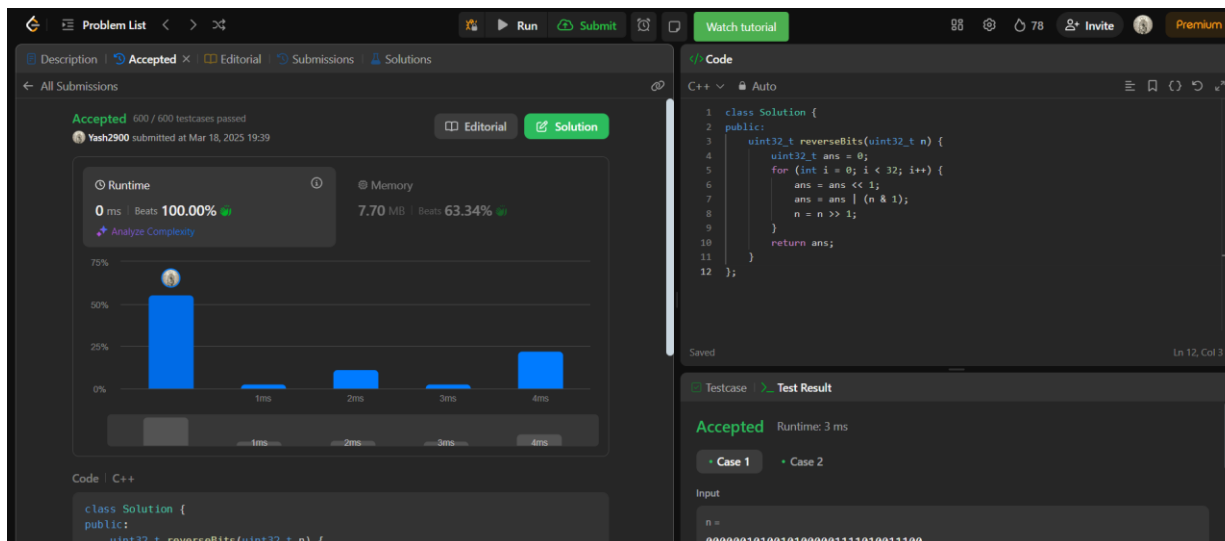
22BCS_IOT_608-B

LeetCode Id: [Yash2900](#)

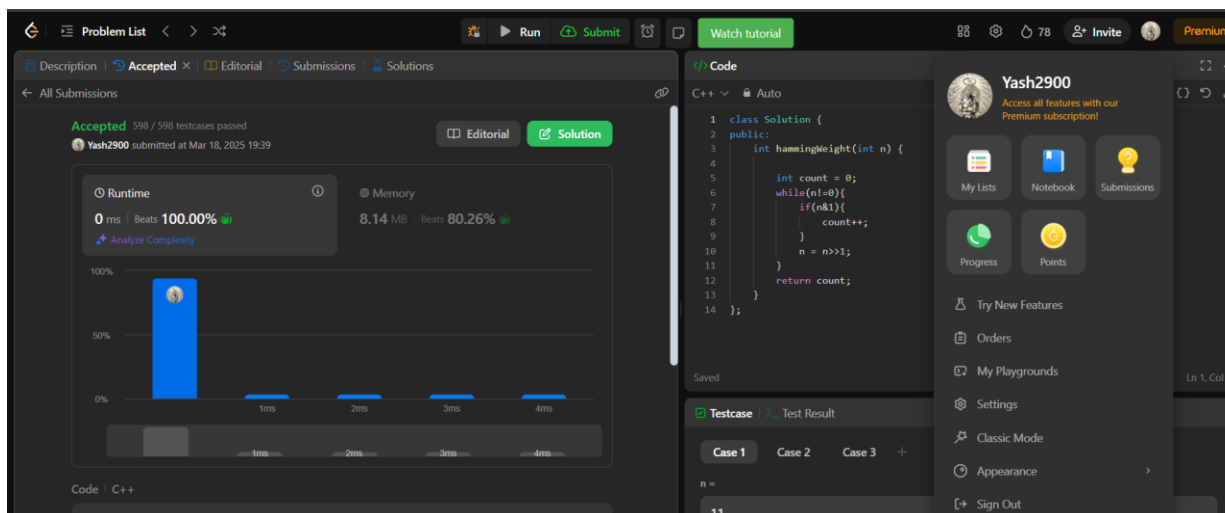
1. Longest Nice Substring: <https://leetcode.com/problems/longest-nice-substring/description/>



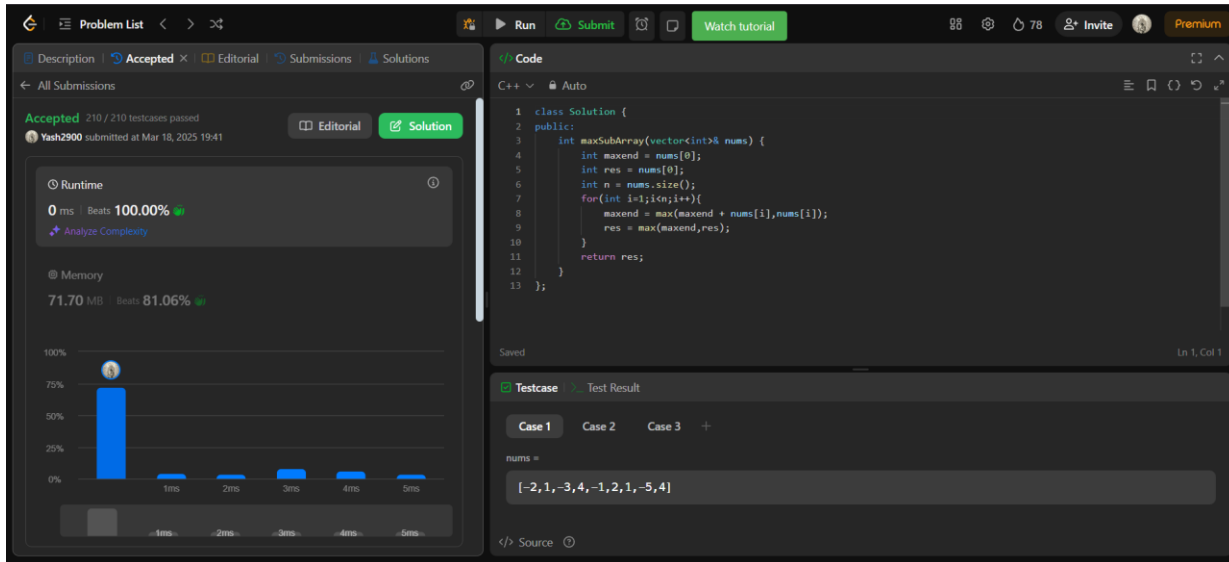
2. Reverse Bits: <https://leetcode.com/problems/reverse-bits/description/>



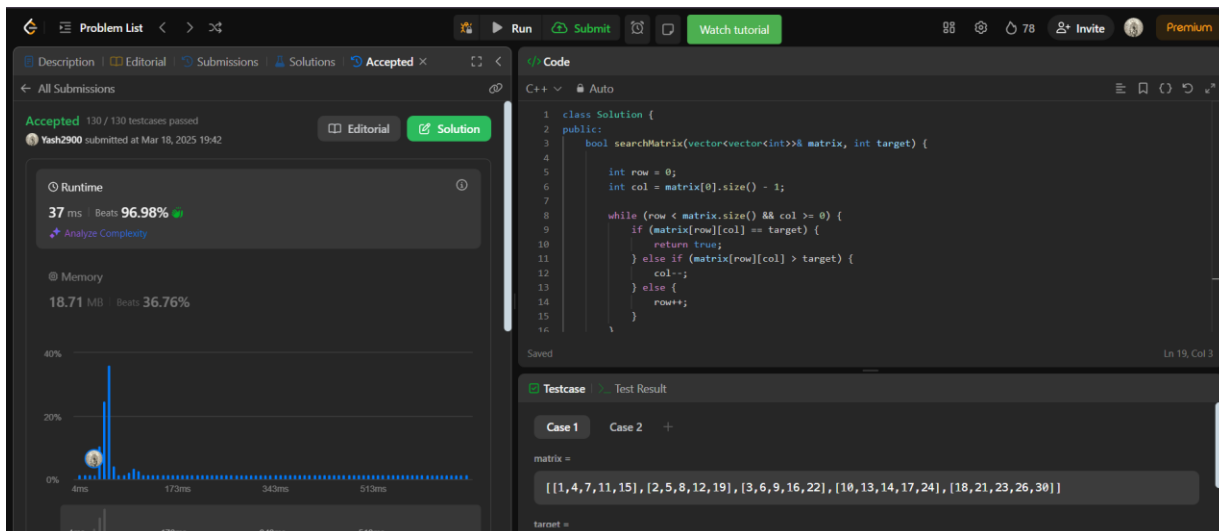
3. Number of 1 Bits: <https://leetcode.com/problems/number-of-1-bits/description/>



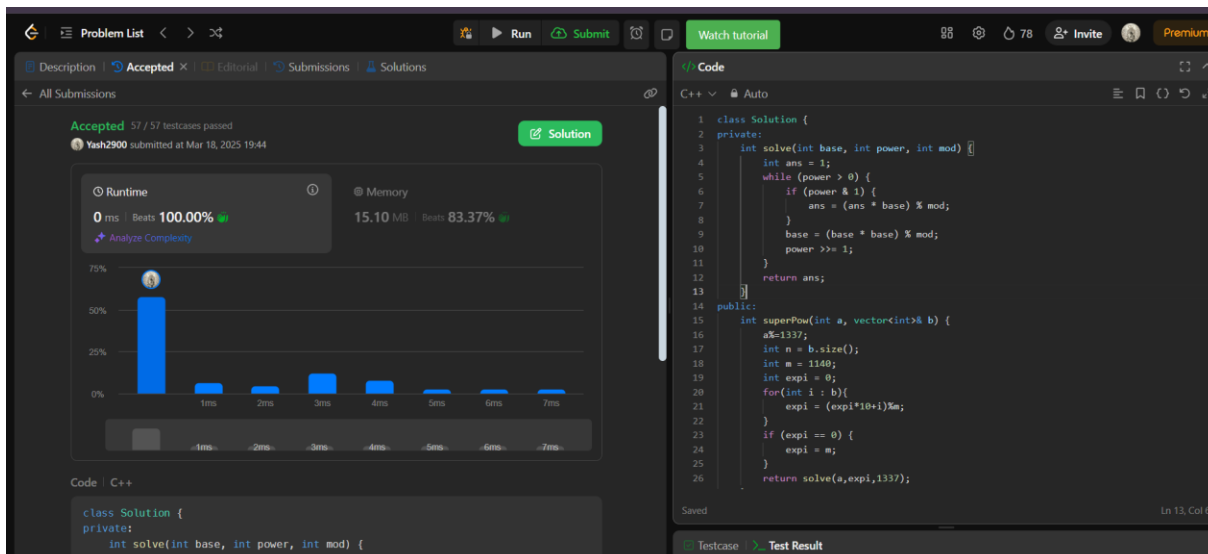
4. Maximum Subarray: <https://leetcode.com/problems/maximum-subarray/description/>



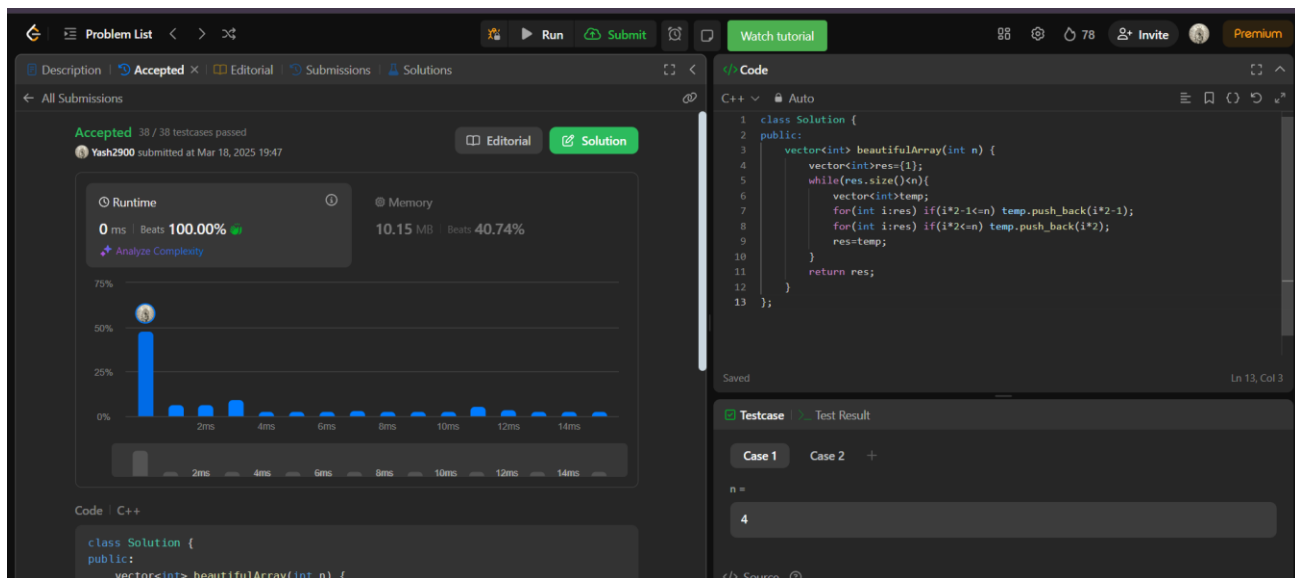
5. Search a 2D Matrix II: <https://leetcode.com/problems/search-a-2d-matrix-ii/description/>



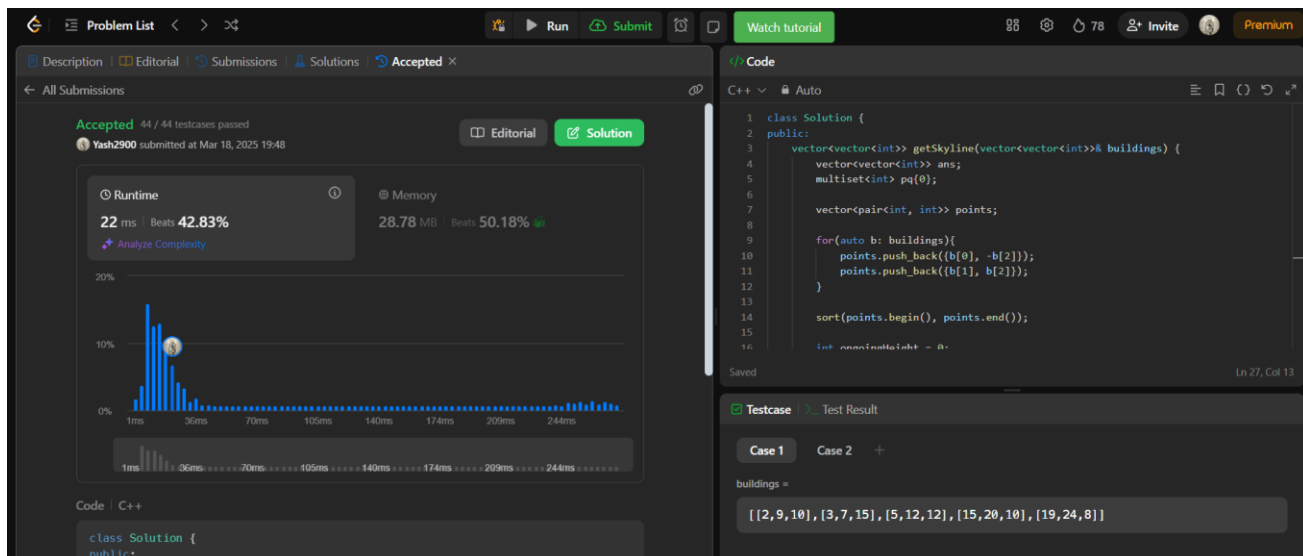
6. Super Pow: <https://leetcode.com/problems/super-pow/description/>



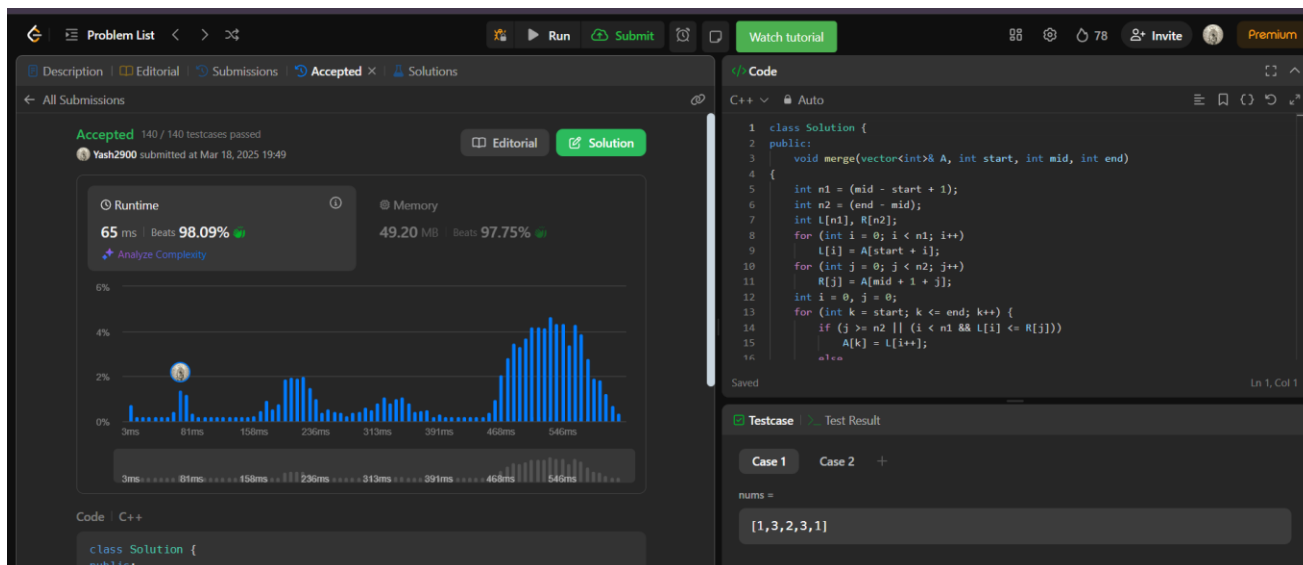
7. Beautiful Array: <https://leetcode.com/problems/beautiful-array/description/>



8. The Skyline Problem: <https://leetcode.com/problems/the-skyline-problem/description/>



9. Reverse Pairs: <https://leetcode.com/problems/reverse-pairs/description/>



10. Longest Increasing Subsequence II: <https://leetcode.com/problems/longest-increasing-subsequence-ii/description/>

Problem List < >

Run Submit Watch tutorial 78 Invite Premium

Description Editorial Submissions Solutions Accepted x

< All Submissions

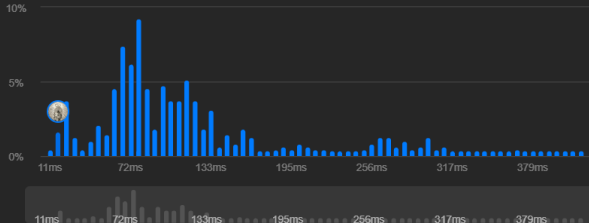
Accepted 84 / 84 testcases passed

Yash2900 submitted at Mar 18, 2025 19:50

Solution

Runtime 17 ms | Beats: 99.59% Memory 46.38 MB | Beats: 98.15%

Analyze Complexity



11ms 72ms 133ms 195ms 256ms 317ms 379ms

Code | C++

```
const int N = 1e5+1;

class Solution {
public:
    int lengthOfLIS(vector<int>& nums, int k) {
        int n = nums.size();
        int t[2*N + 2]; memset(t, 0, sizeof(t));

        auto upd = [&](int p, int v) {
            p += N;
            t[p] = v;
            for(; p > 1; p >>= 1) t[p>>1] = max(t[p], t[p^1]);
        };

        auto qry = [&](int l, int r) {
            l += N, r += N;
            int res = 0;
            while(l <= r) {
                if(l%2 == 1) res = max(res, t[l]);
                if(r%2 == 0) res = max(res, t[r]);
                l = l/2, r = r/2;
            }
            return res;
        };

        int ans = 0;
        for(int i = 0; i < n; i++) {
            int p = qry(0, i);
            upd(i, p+1);
            ans = max(ans, p+1);
        }
        return ans;
    }
};
```

Code C++ Auto

```
1 const int N = 1e5+1;
2
3 class Solution {
4 public:
5     int lengthOfLIS(vector<int>& nums, int k) {
6         int n = nums.size();
7         int t[2*N + 2]; memset(t, 0, sizeof(t));
8
9         auto upd = [&](int p, int v) {
10             p += N;
11             t[p] = v;
12             for(; p > 1; p >>= 1) t[p>>1] = max(t[p], t[p^1]);
13         };
14
15         auto qry = [&](int l, int r) {
16             l += N, r += N;
17             int res = 0;
18             while(l <= r) {
19                 if(l%2 == 1) res = max(res, t[l]);
20                 if(r%2 == 0) res = max(res, t[r]);
21                 l = l/2, r = r/2;
22             }
23             return res;
24         };
25
26         int ans = 0;
27         for(int i = 0; i < n; i++) {
28             int p = qry(0, i);
29             upd(i, p+1);
30             ans = max(ans, p+1);
31         }
32         return ans;
33     }
34 };
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Saved Ln 24, Col 1

Testcase Test Result

Case 1 Case 2 Case 3 +

nums =

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

k =

</> Source