leetcode.com/problems/longest-nice-substring/

Problem List | Run | Submit | Premium

Description | Accepted | Editorial | Solutions | Submissions

All Submissions

Accepted 73 / 73 testcases passed

shashiranjan24 submitted at Mar 18, 2025 11:28

Solution

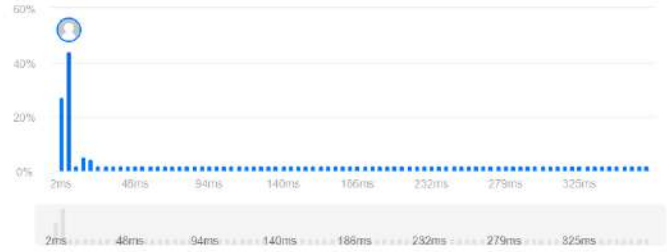
Runtime

7 ms | Beats 64.74%

Analyze Complexity

Memory

14.19 MB | Beats 63.11%



Code | C++

```cpp
class Solution {
public:
    string longestNiceSubstring(string s) {
        if (s.length() < 2) return "";

        unordered_set<char> charSet(s.begin(), s.end()); // Store all characters

        for (int i = 0; i < s.length(); i++) {
```

View more

Code

C++ | Auto

```cpp
class Solution {
public:
    string longestNiceSubstring(string s) {
        if (s.length() < 2) return "";

        unordered_set<char> charSet(s.begin(), s.end());

        for (int i = 0; i < s.length(); i++) {
            char ch = s[i];

            if (charSet.count(tolower(ch)) && charSet.count(toupper(ch))) {
                continue;
            }

            string left = longestNiceSubstring(s.substr(0, i));
            string right = longestNiceSubstring(s.substr(i + 1));
```

Saved

Testcase | Test Result

Case 1 | Case 2 | Case 3

s =

"YazaAay"

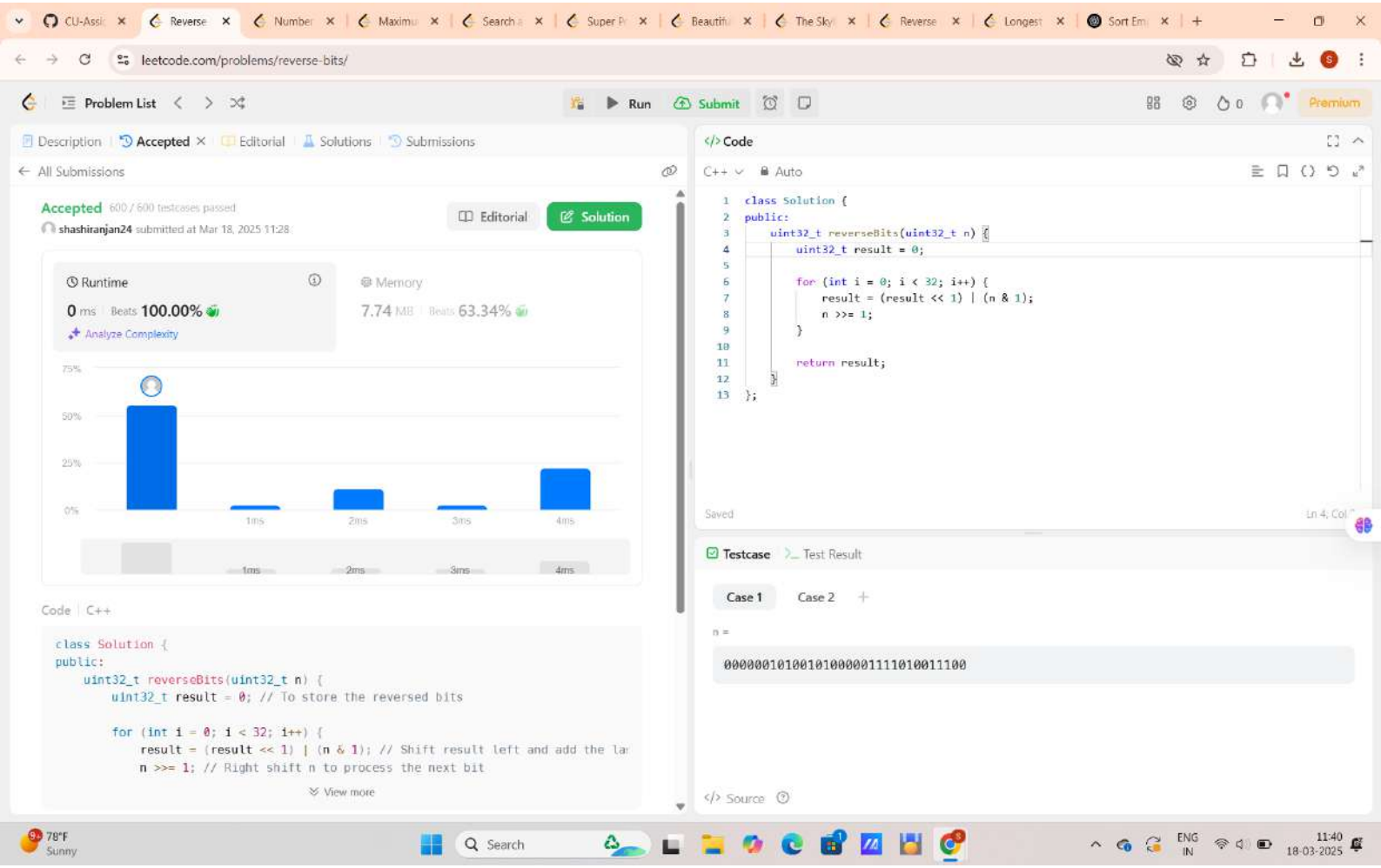
Source

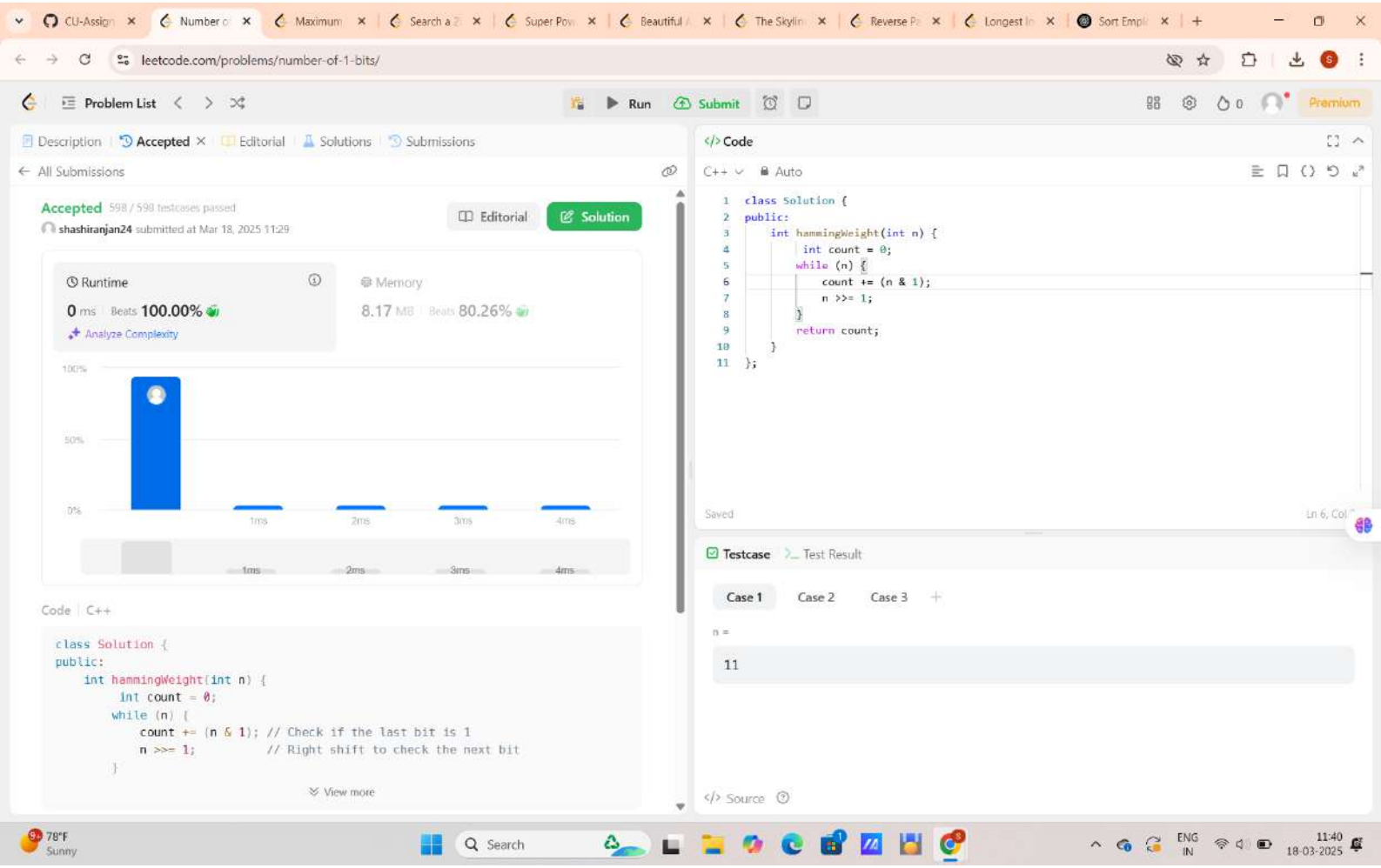
78°F Sunny

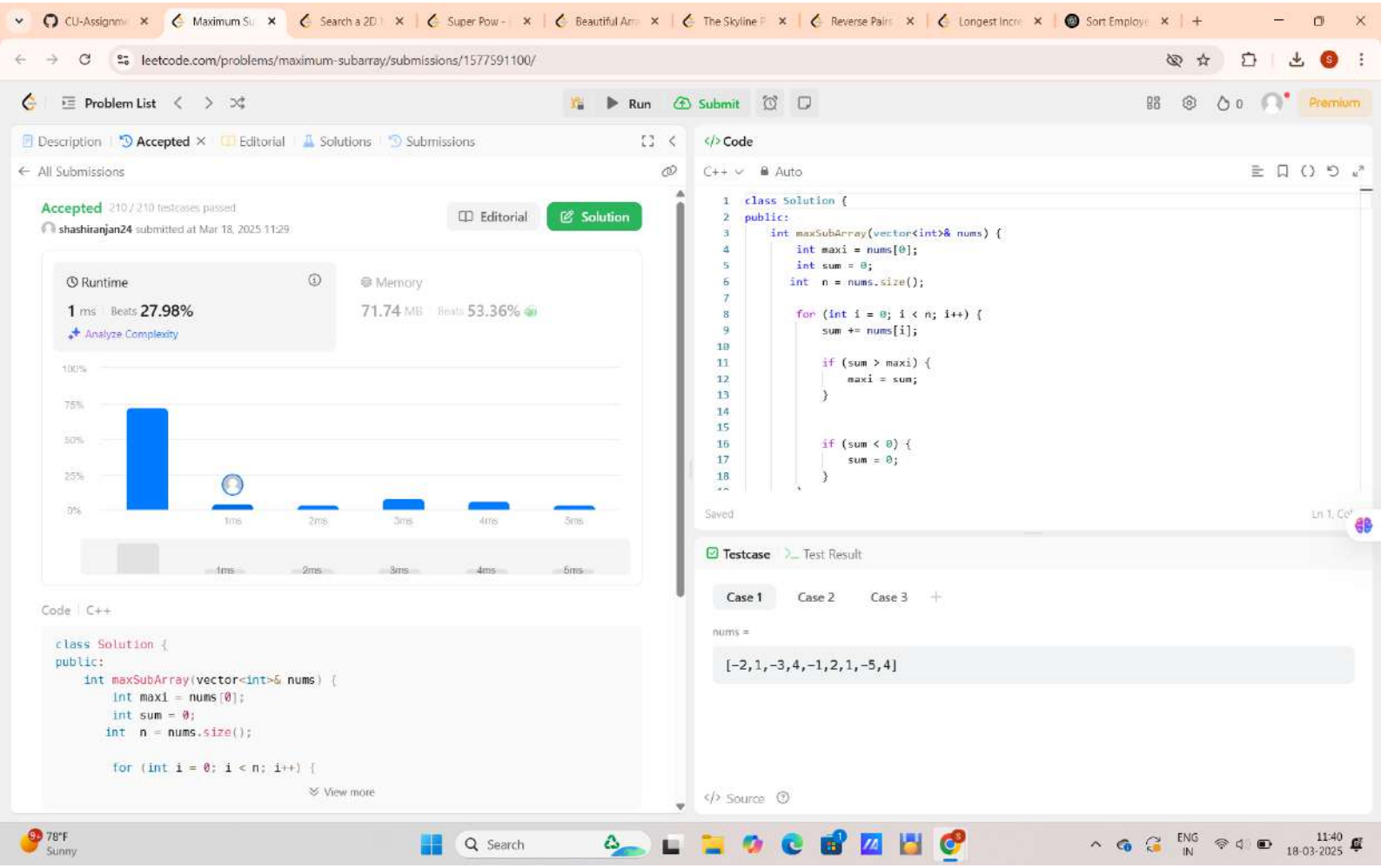
Search

ENG IN

11:40 18-03-2025


leetcode.com/problems/reverse-bits/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 600 / 600 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:28
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
7.74 MB | Beats 63.34%
75%
50%
25%
0%
1ms
2ms
3ms
4ms
Code | C++
class Solution {
public:
uint32_t reverseBits(uint32_t n) {
uint32_t result = 0; // To store the reversed bits
for (int i = 0; i < 32; i++) {
result = (result << 1) | (n & 1); // Shift result left and add the la
n >>= 1; // Right shift n to process the next bit
View more
Code
C++
Auto
class Solution {
public:
uint32_t reverseBits(uint32_t n) {
uint32_t result = 0;
for (int i = 0; i < 32; i++) {
result = (result << 1) | (n & 1);
n >>= 1;
}
return result;
}
};
Saved
Ln 4, Col
Testcase
Test Result
Case 1
Case 2
n =
00000010100101000001111010011100
Source
78°F
Sunny
Search
ENG
IN
11:40
18-03-2025


leetcode.com/problems/number-of-1-bits/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 598 / 598 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:29
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
8.17 MB | Beats 80.26%
100%
50%
0%
1ms
2ms
3ms
4ms
Code | C++
class Solution {
public:
int hammingWeight(int n) {
int count = 0;
while (n) {
count += (n & 1); // Check if the last bit is 1
n >>= 1; // Right shift to check the next bit
}
View more
Code
C++
Auto
class Solution {
public:
int hammingWeight(int n) {
int count = 0;
while (n) {
count += (n & 1);
n >>= 1;
}
return count;
}
};
Saved
Ln 6, Col
Testcase
Test Result
Case 1
Case 2
Case 3
n =
11
Source
78°F
Sunny
Search
ENG
IN
11:40
18-03-2025


leetcode.com/problems/maximum-subarray/submissions/1577591100/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 210 / 210 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:29
Editorial
Solution
Runtime
1 ms | Beats 27.98%
Analyze Complexity
Memory
71.74 MB | Beats 53.36%
100%
75%
50%
25%
0%
1ms
2ms
3ms
4ms
5ms
Code | C++
class Solution {
public:
int maxSubArray(vector<int>& nums) {
int maxi = nums[0];
int sum = 0;
int n = nums.size();
for (int i = 0; i < n; i++) {
View more
Code
C++
Auto
class Solution {
public:
int maxSubArray(vector<int>& nums) {
int maxi = nums[0];
int sum = 0;
int n = nums.size();
for (int i = 0; i < n; i++) {
sum += nums[i];
if (sum > maxi) {
maxi = sum;
}
if (sum < 0) {
sum = 0;
}
Saved
Testcase
Test Result
Case 1
Case 2
Case 3
nums =
[-2,1,-3,4,-1,2,1,-5,4]
Source
78°F
Sunny
Search
ENG
IN
11:40
18-03-2025

CU-Assignments | Search a 2D Mat | Super Pow - Lee | Beautiful Array - | The Skyline Prob | Reverse Pairs - L | Longest Increasi | Sort Employee L

leetcode.com/problems/search-a-2d-matrix-ii/

Problem List | Run | Submit | Premium

Description | Accepted | Editorial | Solutions | Submissions

All Submissions

Accepted 130 / 130 testcases passed

shashiranjan24 submitted at Mar 18, 2025 11:31

Editorial | Solution

Runtime

57 ms | Beats 38.68%

Analyze Complexity

Memory

18.79 MB | Beats 36.76%

Code | C++

```cpp
class Solution {
public:
    bool searchMatrix(vector<vector<int>>& matrix, int target) {
        if (matrix.empty() || matrix[0].empty()) return false;

        int rows = matrix.size();
        int cols = matrix[0].size();
        int row = 0, col = cols - 1; // Start from the top-right corner
```

View more

Code

C++ | Auto

```cpp
        while (row < rows && col >= 0) {
            if (matrix[row][col] == target) {
                return true;
            } else if (matrix[row][col] > target) {
                col--;
            } else {
                row++;
            }
        }

        return false;
    }
};
```

Saving...

Testcase | Test Result

Case 1 | Case 2

matrix =

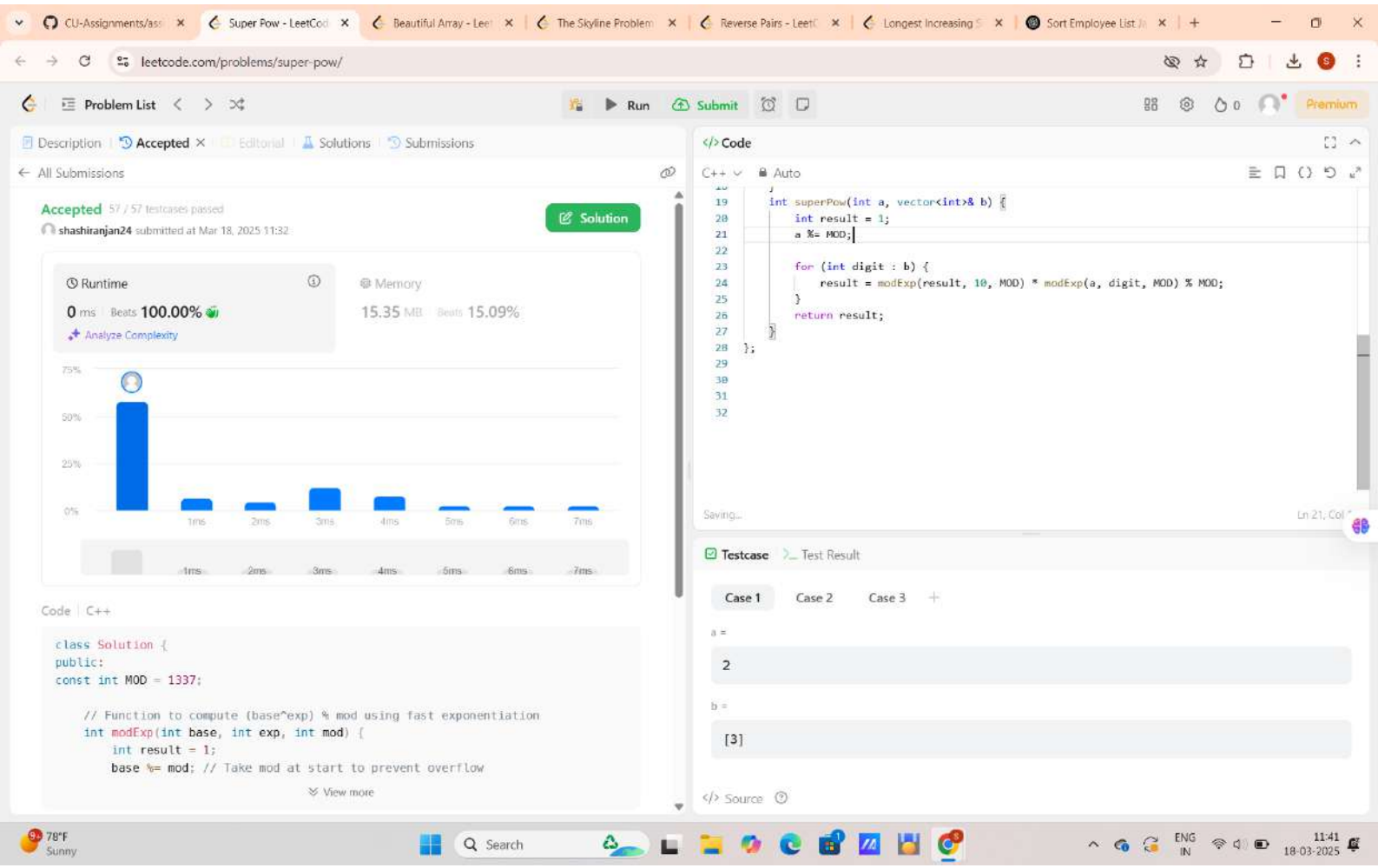
```
[[1,4,7,11,15],[2,5,8,12,19],[3,6,9,16,22],[10,13,14,17,24],[18,21,23,26,30]]
```

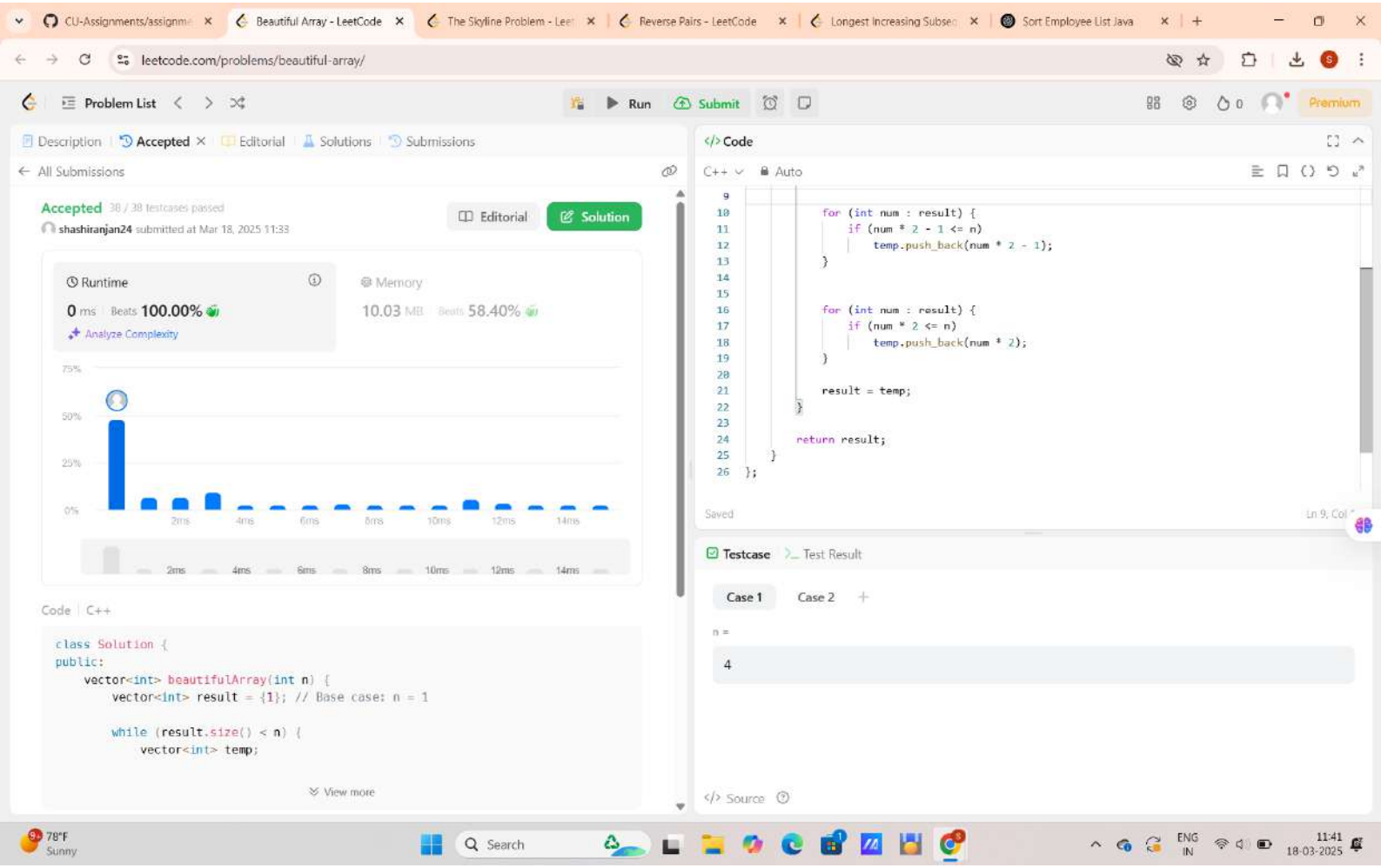
target =

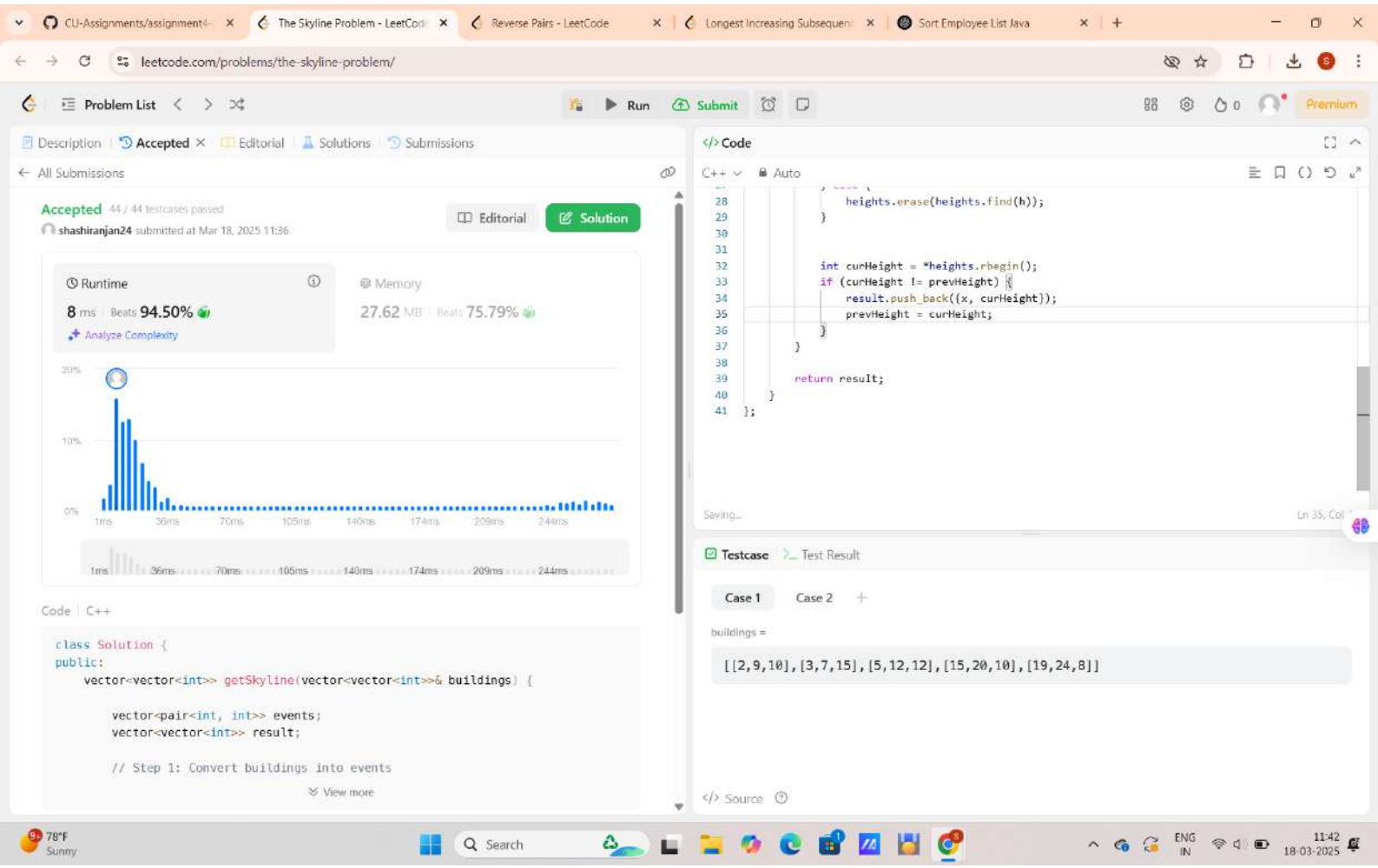
```
5
```

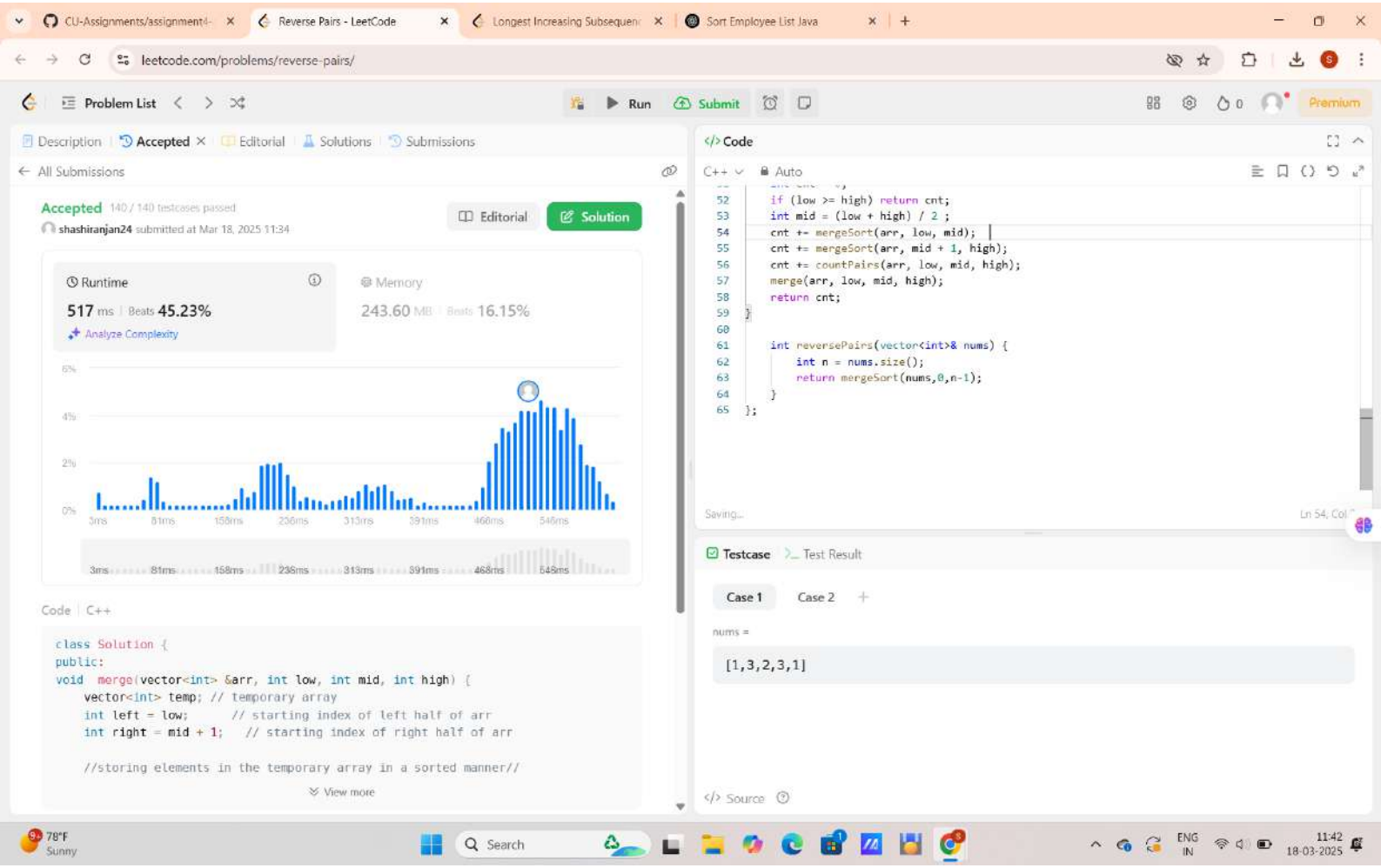
Source

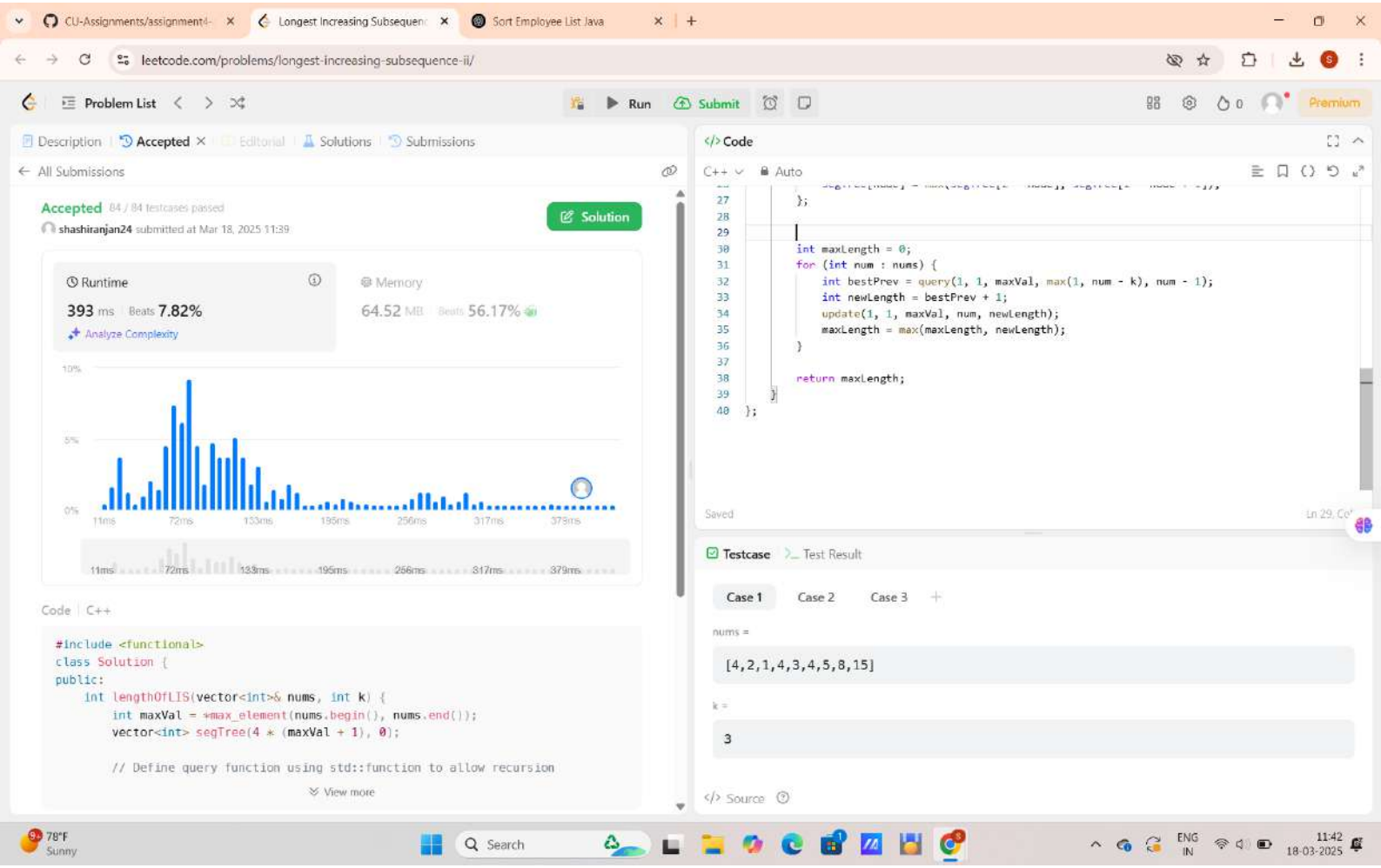
78°F Sunny | Search | ENG IN | 11:41 18-03-2025


CU-Assignments/ass
Super Pow - LeetCod
Beautiful Array - Lee
The Skyline Problem
Reverse Pairs - Leet
Longest Increasing S
Sort Employee List
leetcode.com/problems/super-pow/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 57 / 57 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:32
Solution
Runtime
0 ms Beats 100.00%
Analyze Complexity
Memory
15.35 MB Beats 15.09%
75%
50%
25%
0%
1ms
2ms
3ms
4ms
5ms
6ms
7ms
Code | C++
class Solution {
public:
const int MOD = 1337;
// Function to compute (base^exp) % mod using fast exponentiation
int modExp(int base, int exp, int mod) {
int result = 1;
base %= mod; // Take mod at start to prevent overflow
View more
Code
C++
Auto
int superPow(int a, vector<int>& b) {
int result = 1;
a %= MOD;
for (int digit : b) {
result = modExp(result, 10, MOD) * modExp(a, digit, MOD) % MOD;
}
return result;
}
};
Saving...
Ln 21, Col
Testcase
Test Result
Case 1
Case 2
Case 3
a =
2
b =
[3]
Source
78°F
Sunny
Search
ENG
IN
11:41
18-03-2025


CU-Assignments/assignme
Beautiful Array - LeetCode
The Skyline Problem - Leet
Reverse Pairs - LeetCode
Longest Increasing Subseq
Sort Employee List Java
leetcode.com/problems/beautiful-array/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 38 / 38 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:33
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
10.03 MB | Beats 58.40%
75%
50%
25%
0%
2ms
4ms
6ms
8ms
10ms
12ms
14ms
Code | C++
class Solution {
public:
vector<int> beautifulArray(int n) {
vector<int> result = {1}; // Base case: n = 1
while (result.size() < n) {
vector<int> temp;
View more
Code
C++
Auto
for (int num : result) {
if (num * 2 - 1 <= n)
temp.push_back(num * 2 - 1);
}
for (int num : result) {
if (num * 2 <= n)
temp.push_back(num * 2);
}
result = temp;
}
return result;
}
};
Saved
Ln 9, Col
Testcase
Test Result
Case 1
Case 2
n =
4
Source
78°F
Sunny
Search
ENG
IN
11:41
18-03-2025


CU-Assignments/assignment4-
The Skyline Problem - LeetCod
Reverse Pairs - LeetCode
Longest Increasing Subsequen
Sort Employee List Java
leetcode.com/problems/the-skyline-problem/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 44 / 44 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:36
Editorial
Solution
Runtime
8 ms | Beats 94.50%
Analyze Complexity
Memory
27.62 MB | Beats 75.79%
20%
10%
0%
1ms
36ms
70ms
105ms
140ms
174ms
209ms
244ms
Code | C++
class Solution {
public:
vector<vector<int>> getSkyline(vector<vector<int>>& buildings) {
vector<pair<int, int>> events;
vector<vector<int>> result;
// Step 1: Convert buildings into events
View more
Code
C++
Auto
heights.erase(heights.find(h));
int curHeight = *heights.rbegin();
if (curHeight != prevHeight) {
result.push_back({x, curHeight});
prevHeight = curHeight;
return result;
};
Saving...
Ln 35, Col
Testcase
Test Result
Case 1
Case 2
buildings =
[[2,9,10],[3,7,15],[5,12,12],[15,20,10],[19,24,8]]
Source
78°F
Sunny
Search
ENG
IN
11:42
18-03-2025


CU-Assignments/assignment4-
Reverse Pairs - LeetCode
Longest Increasing Subsequen
Sort Employee List Java
leetcode.com/problems/reverse-pairs/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 140 / 140 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:34
Editorial
Solution
Runtime
517 ms | Beats 45.23%
Analyze Complexity
Memory
243.60 MB | Beats 16.15%
6%
4%
2%
0%
3ms
81ms
158ms
236ms
313ms
391ms
468ms
546ms
Code | C++
class Solution {
public:
void merge(vector<int> &arr, int low, int mid, int high) {
vector<int> temp; // temporary array
int left = low; // starting index of left half of arr
int right = mid + 1; // starting index of right half of arr
//storing elements in the temporary array in a sorted manner//
View more
Code
C++
Auto
52 if (low >= high) return cnt;
53 int mid = (low + high) / 2 ;
54 cnt += mergeSort(arr, low, mid);
55 cnt += mergeSort(arr, mid + 1, high);
56 cnt += countPairs(arr, low, mid, high);
57 merge(arr, low, mid, high);
58 return cnt;
59 }
60
61 int reversePairs(vector<int>& nums) {
62 int n = nums.size();
63 return mergeSort(nums,0,n-1);
64 }
65 };
Saving...
Ln 54, Col
Testcase
Test Result
Case 1
Case 2
nums =
[1,3,2,3,1]
Source
78°F
Sunny
Search
ENG
IN
11:42
18-03-2025


CU-Assignments/assignment4-
Longest Increasing Subsequen
Sort Employee List Java
leetcode.com/problems/longest-increasing-subsequence-ii/
Problem List
Run
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 84 / 84 testcases passed
shashiranjan24 submitted at Mar 18, 2025 11:39
Solution
Runtime
393 ms Beats 7.82%
Analyze Complexity
Memory
64.52 MB Beats 56.17%
10%
5%
0%
11ms
72ms
133ms
195ms
256ms
317ms
379ms
Code | C++
#include <functional>
class Solution {
public:
int lengthOfLIS(vector<int>& nums, int k) {
int maxVal = *max_element(nums.begin(), nums.end());
vector<int> segTree(4 * (maxVal + 1), 0);
// Define query function using std::function to allow recursion
View more
Code
C++
Auto
};
int maxLength = 0;
for (int num : nums) {
int bestPrev = query(1, 1, maxVal, max(1, num - k), num - 1);
int newLength = bestPrev + 1;
update(1, 1, maxVal, num, newLength);
maxLength = max(maxLength, newLength);
}
return maxLength;
}
};
Saved
Testcase
Test Result
Case 1
Case 2
Case 3
nums =
[4,2,1,4,3,4,5,8,15]
k =
3
Source
78°F
Sunny
Search
ENG
IN
11:42
18-03-2025