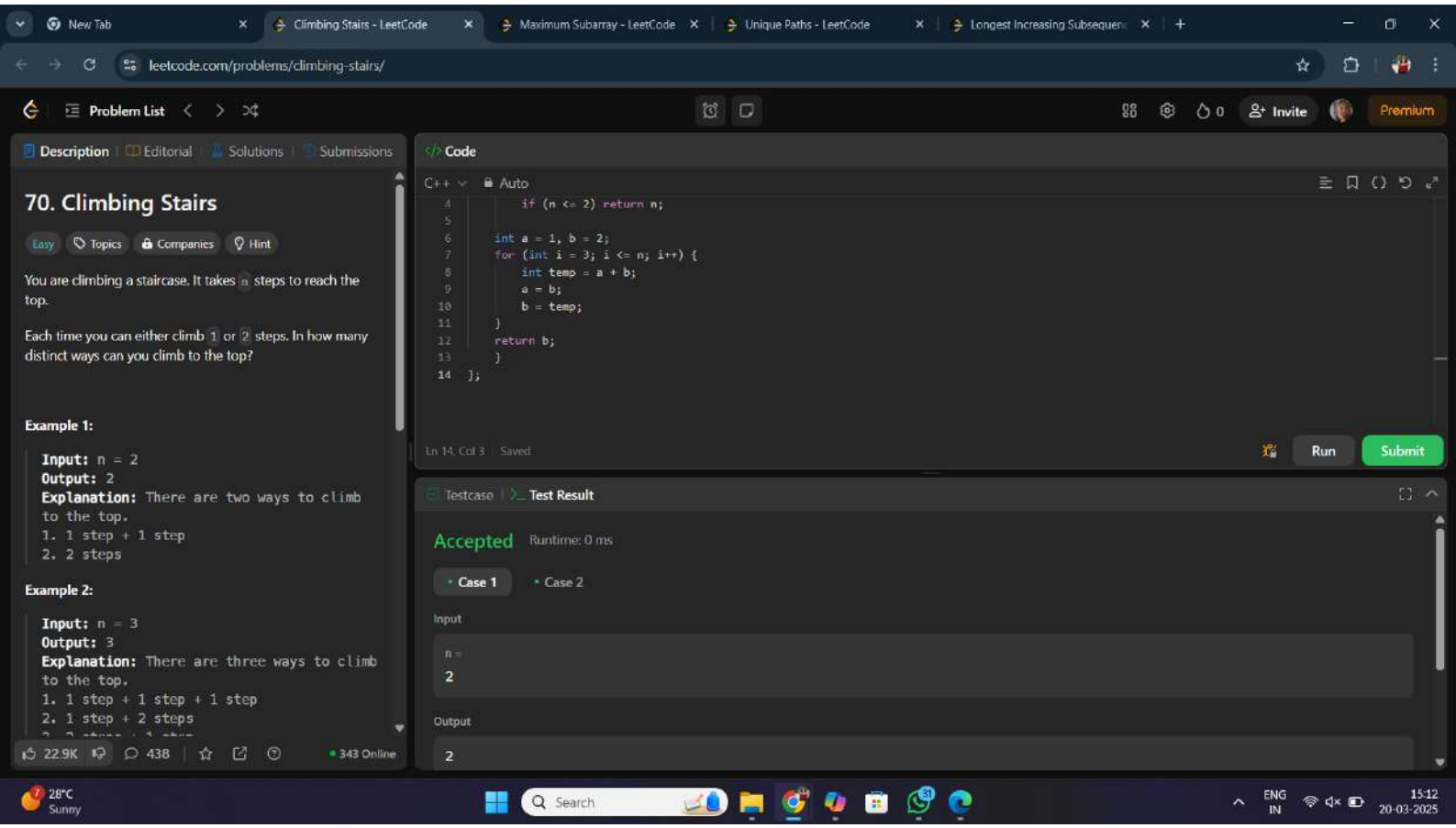
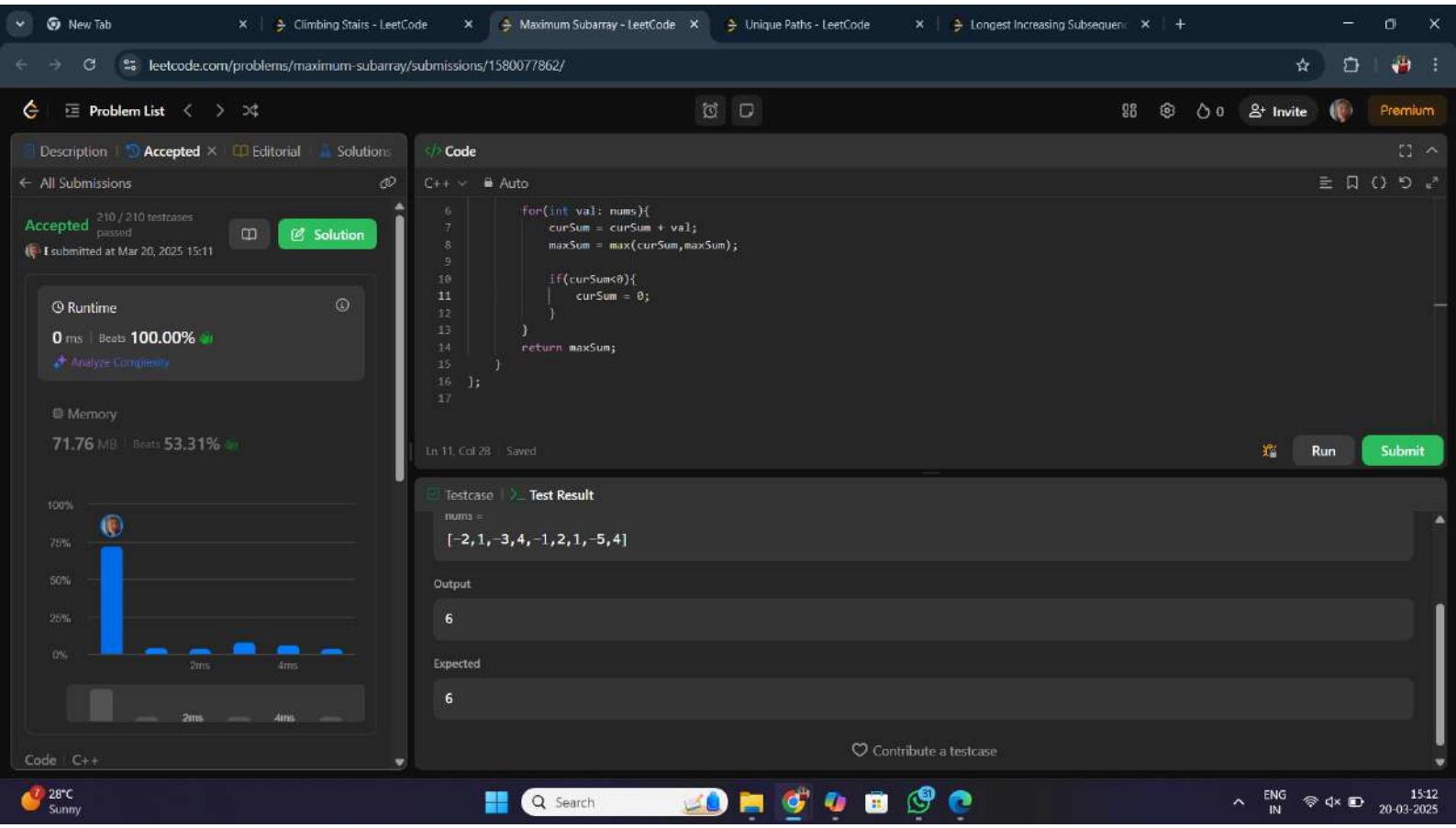
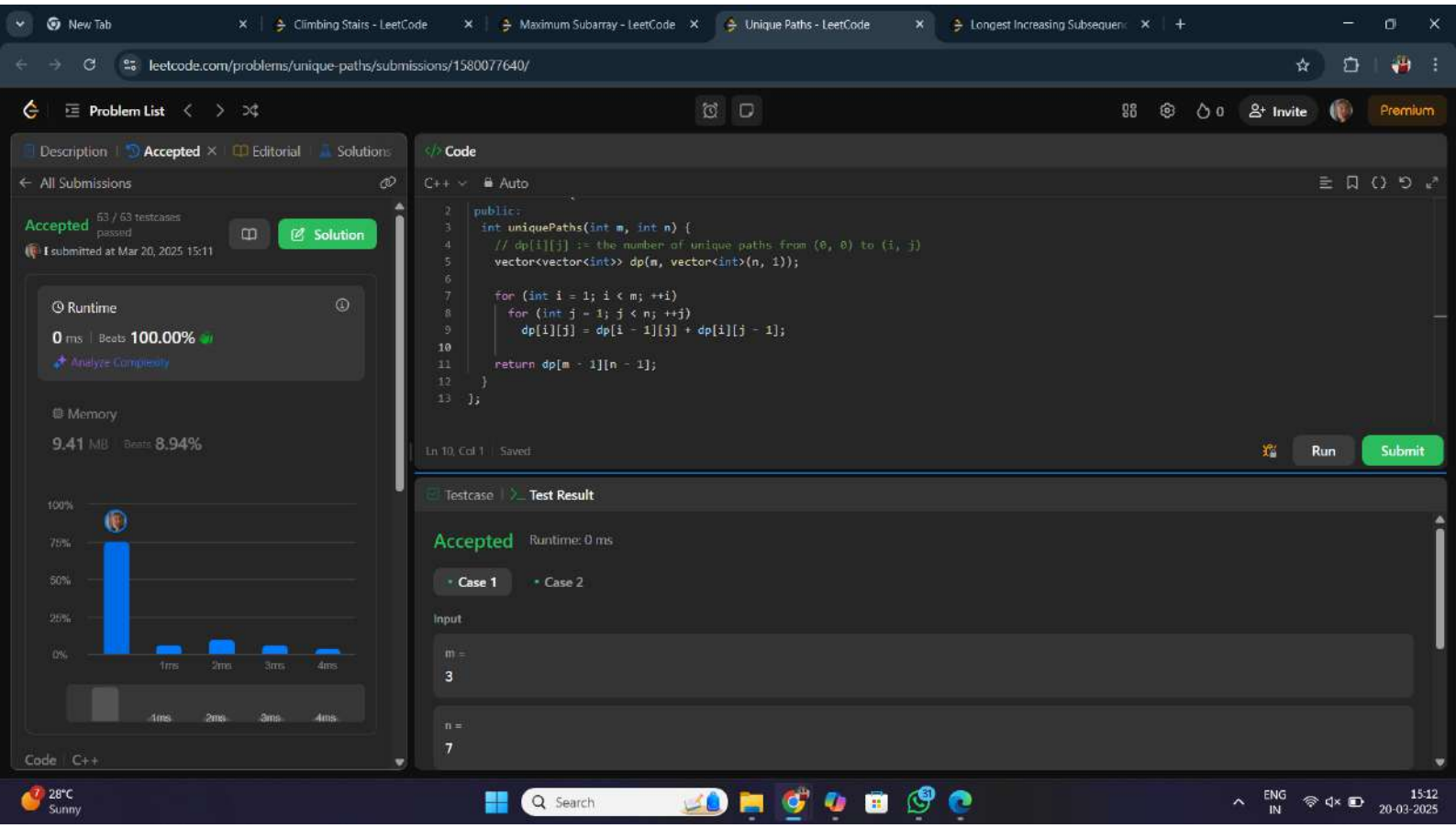
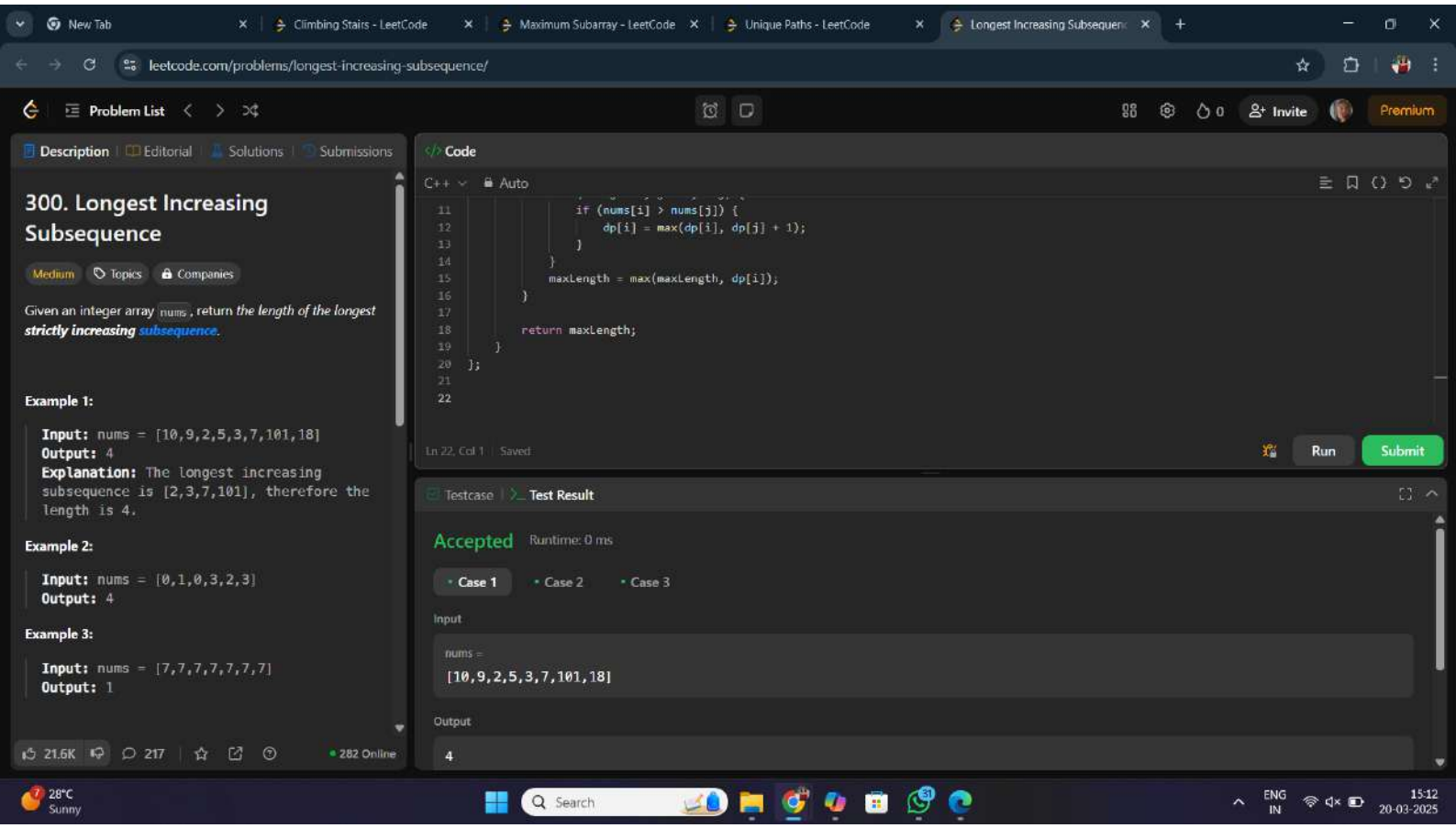

New Tab
Climbing Stairs - LeetCode
Maximum Subarray - LeetCode
Unique Paths - LeetCode
Longest Increasing Subsequen
leetcode.com/problems/climbing-stairs/
Problem List
Invite
Premium
Description
Editorial
Solutions
Submissions
70. Climbing Stairs
Easy
Topics
Companies
Hint
You are climbing a staircase. It takes n steps to reach the top.
Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top?
Example 1:
Input: n = 2
Output: 2
Explanation: There are two ways to climb to the top.
1. 1 step + 1 step
2. 2 steps
Example 2:
Input: n = 3
Output: 3
Explanation: There are three ways to climb to the top.
1. 1 step + 1 step + 1 step
2. 1 step + 2 steps
22.9K
438
343 Online
Code
C++
Auto
if (n <= 2) return n;
int a = 1, b = 2;
for (int i = 3; i <= n; i++) {
int temp = a + b;
a = b;
b = temp;
}
return b;
}
};
Ln 14, Col 3
Saved
Run
Submit
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Input
n =
2
Output
2
28°C
Sunny
Search
ENG
IN
15:12
20-03-2025


New Tab
Climbing Stairs - LeetCode
Maximum Subarray - LeetCode
Unique Paths - LeetCode
Longest Increasing Subsequen
leetcode.com/problems/maximum-subarray/submissions/1580077862/
Problem List
Invite
Premium
Description
Accepted
Editorial
Solutions
All Submissions
Accepted
210 / 210 testcases passed
submitted at Mar 20, 2025 15:11
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
71.76 MB | Beats 53.31%
100%
75%
50%
25%
0%
2ms
4ms
Code | C++
Code
C++
Auto
for(int val: nums){
curSum = curSum + val;
maxSum = max(curSum,maxSum);
if(curSum<0){
curSum = 0;
}
}
return maxSum;
}
};
Ln 11, Col 28 | Saved
Run
Submit
Testcase | Test Result
nums =
[-2,1,-3,4,-1,2,1,-5,4]
Output
6
Expected
6
Contribute a testcase
28°C
Sunny
Search
ENG
IN
15:12
20-03-2025


New Tab
Climbing Stairs - LeetCode
Maximum Subarray - LeetCode
Unique Paths - LeetCode
Longest Increasing Subsequen
leetcode.com/problems/unique-paths/submissions/1580077640/
Problem List
Invite
Premium
Description
Accepted
Editorial
Solutions
All Submissions
Accepted
63 / 63 testcases passed
submitted at Mar 20, 2025 15:11
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
9.41 MB | Beats 8.94%
100%
75%
50%
25%
0%
1ms
2ms
3ms
4ms
Code | C++
Code
C++
Auto
public:
int uniquePaths(int m, int n) {
// dp[i][j] := the number of unique paths from (0, 0) to (i, j)
vector<vector<int>> dp(m, vector<int>(n, 1));
for (int i = 1; i < m; ++i)
for (int j = 1; j < n; ++j)
dp[i][j] = dp[i - 1][j] + dp[i][j - 1];
return dp[m - 1][n - 1];
}
};
Ln 10, Col 1 | Saved
Run
Submit
Testcase | Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Input
m =
3
n =
7
28°C
Sunny
Search
ENG
IN
15:12
20-03-2025


New Tab
Climbing Stairs - LeetCode
Maximum Subarray - LeetCode
Unique Paths - LeetCode
Longest Increasing Subsequen
leetcode.com/problems/longest-increasing-subsequence/
Problem List
Invite
Premium
Description
Editorial
Solutions
Submissions
300. Longest Increasing Subsequence
Medium
Topics
Companies
Given an integer array nums, return the length of the longest strictly increasing subsequence.
Example 1:
Input: nums = [10,9,2,5,3,7,101,18]
Output: 4
Explanation: The longest increasing subsequence is [2,3,7,101], therefore the length is 4.
Example 2:
Input: nums = [0,1,0,3,2,3]
Output: 4
Example 3:
Input: nums = [7,7,7,7,7,7,7]
Output: 1
21.6K
217
282 Online
Code
C++
Auto
if (nums[i] > nums[j]) {
dp[i] = max(dp[i], dp[j] + 1);
}
}
maxLength = max(maxLength, dp[i]);
}
return maxLength;
}
};
Ln 22, Col 1
Saved
Run
Submit
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Case 3
Input
nums =
[10,9,2,5,3,7,101,18]
Output
4
28°C
Sunny
Search
ENG
IN
15:12
20-03-2025