

<https://leetcode.com/problems/climbing-stairs/submissions/1595685612/>

Accepted 45 / 45 testcases passed

sms... submitted at Apr 03, 2025 20:51

Editorial

Solution

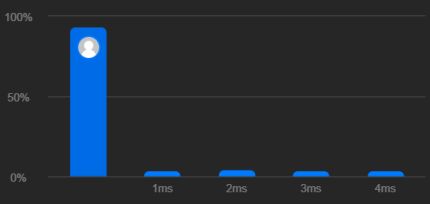
Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

8.67 MB | Beats 17.21%



```
1 class Solution {
2 public:
3     int climbStairs(int n) {
4         // dp[i] := the number of ways to climb to the i-th stair
5         vector<int> dp(n + 1);
6         dp[0] = 1;
7         dp[1] = 1;
8
9         for (int i = 2; i <= n; ++i)
10             dp[i] = dp[i - 1] + dp[i - 2];
11     }
12 }
```

Saved

Testcase

Test Result

Case 1

Case 2

+

n =

2

<https://leetcode.com/problems/jump-game/submissions/1595684328/>

Accepted 174 / 174 testcases passed

sms... submitted at Apr 03, 2025 20:49

Editorial

Solution

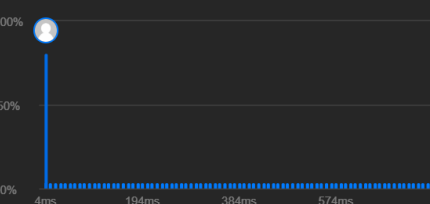
Runtime

0 ms | Beats 100.00%

Analyze Complexity

Memory

52.16 MB | Beats 87.33%



```
3 bool canJump(vector<int>& nums) {
4     int i = 0;
5
6     for (int reach = 0; i < nums.size() && i <= reach; ++i)
7         reach = max(reach, i + nums[i]);
8
9     return i == nums.size();
10 }
11 ;
```

Saved

Testcase

Test Result

Accepted

Runtime: 0 ms

Case 1

Case 2

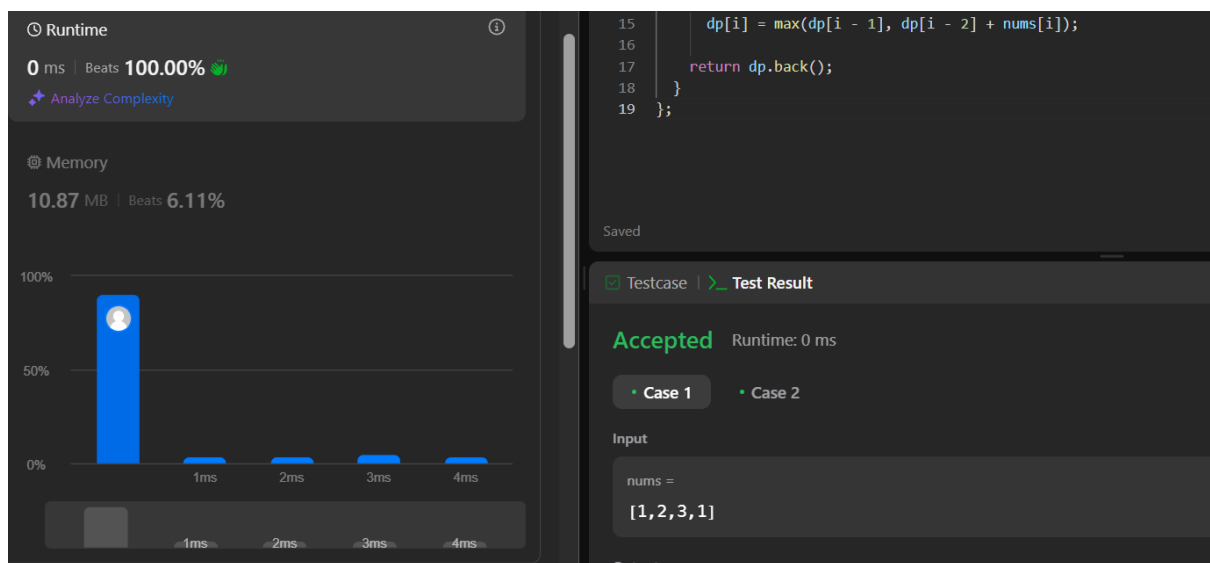
Input

nums =

[2, 3, 1, 1, 4]

Output

<https://leetcode.com/problems/house-robber/submissions/1595683173/>



<https://leetcode.com/problems/maximum-subarray/submissions/1595682268/>

