70 Climbing Stairs (link)

Description

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
3. 2 steps + 1 step
```

Constraints:

- `1 <= n <= 45`

(scroll down for solution)

Solution

Language: cpp

Status: Accepted

```cpp
class Solution {
public:
    int climbStairs(int n) {
        if (n == 1) return 1;
        if (n == 2) return 2;

        vector<int> dp(n + 1);
        dp[1] = 1;
        dp[2] = 2;

        for (int i = 3; i <= n; ++i) {
            dp[i] = dp[i-1] + dp[i-2];
        }

        return dp[n];
    }
};
```