070. Climbing Stairs

070 Climbing Stairs

- **Dynamic Programming**

Description

You are climbing a stair case. It takes *n* steps to reach to the top.

Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top?

Note: Given *n* will be a positive integer.

Example 1:

```
Input: 2
Output:  2
Explanation:  There are two ways to climb to the top.

1. 1 step + 1 step
2. 2 steps
```

Example 2:

```
Input: 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
```

1. Thought line

2. Dynamic Programming

```cpp
class Solution {
public:
    int climbStairs(int n) {
        // f[n] = f[n-1] + f[n-2]
        vector<int> ladder(n+1);
        ladder[1] = 1, ladder[2] = 2;
        for (int i = 3; i<=n; ++i){
            ladder[i] = ladder[i-1] + ladder[i-2];
        }
        return ladder[n];
    }
};
```