

Description

Solution

Discuss (999+)

Submissions

70. Climbing Stairs

Easy10206316Add to ListShare

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

Accepted1,375,921Submissions2,717,914

Seen this question in a real interview before?

YesNo

Companies

Related Topics

Similar Questions

Show Hint 1

JavaAutocomplete

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

```
class Solution {
    public int climbStairs(int n) {

        if(n == 0 || n == 1)
            return 1;

        int[] arr = new int[n + 1];

        arr[0] = 1;
        arr[1] = 1;

        for(int i = 2; i <= n; i++)
            arr[i] = arr[i - 1] + arr[i - 2];

        return arr[n];
    }
}
```

Your previous code was restored from your local storage. [Reset to default](#)

Console

Contribute

Run Code

Submit

Problems

Pick One

< Prev

40

Next >

Console

Contribute

Run Code

Submit