70. Climbing Stairs

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

Solution

public class Solution

{

    public int ClimbStairs(int n)

    {

        if (n == 1)

        {

            return 1;

        }

        int first = 1;

        int second = 2;

        int output = second;

        for (int i = 3; i <= n; i++)

        {

            output = first + second;

            first = second;

            second = output;

        }

        return output;

    }

}