

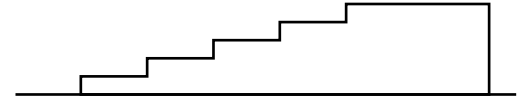
70. Climbing Stairs

Easy

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. $n = 5$



```
class Solution1:
    def climbStairs(self, n: int) -> int:
        dpset = [-1] * (n + 2)
```

```
    def dynamic(i):
        if i == 0 or i == 1:
            return 1
        if dpset[i] != -1:
            return dpset[i]
        left = dynamic(i - 1)
        right = dynamic(i - 2)
        dpset[i] = left + right
        return left + right
    return dynamic(n)
```

dpset[5] != -1

dpset = [-1, -1, -1, -1, -1, -1, -1]

0 1 2 3 4 5 6

left = 1

dynamic(2-2)

right = 1

dpset[2] = 1 + 1

[-1, -1, 2, -1, -1, -1, -1]

0 1 2 3

return 2

left = 2

right = 1

dpset[3] = 3

[-1, -1, -1, -1, -1, -1, -1]

[-1, -1, 2, -1, -1, -1, -1]

[-1, -1, 2, 3, -1, -1, -1]

[-1, -1, 2, 3, 5, -1, -1]

[-1, -1, 2, 3, 5, 8, -1]

