# 题目

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?

# 思路

到达当前节点的走法，**只取决于上一次是走了 one step 或者 two steps**，所以比较容易求得递归方程。

countOfWays(n) = countOfWays(n-1) + countOfWays(n-2)

# 代码

有了递归方程就非常容易写出代码了。

int climbStairs(int n) {

if(n == 1) return 1;

if(n == 2) return 2;

int a = 1, b = 2, curr;

for(int i = 3; i <= n; i++) {

curr = a + b;

a = b;

b = curr;

}

return curr;

}

时间复杂度 O（n）