



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## WORKSHEET 6

**Student Name:**Avreet Singh

**UID:** 22BCS16488

**Branch:** BE-CSE

**Section/Group:** 22BCS\_NTPP-602-A

**Semester:** 6<sup>th</sup>

**Date of Performance:** 20/02/2025

**Subject Name:** AP LAB - II

**Subject Code:** 22CSP-351

### 1. Aim:

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?

### 2. Source Code:

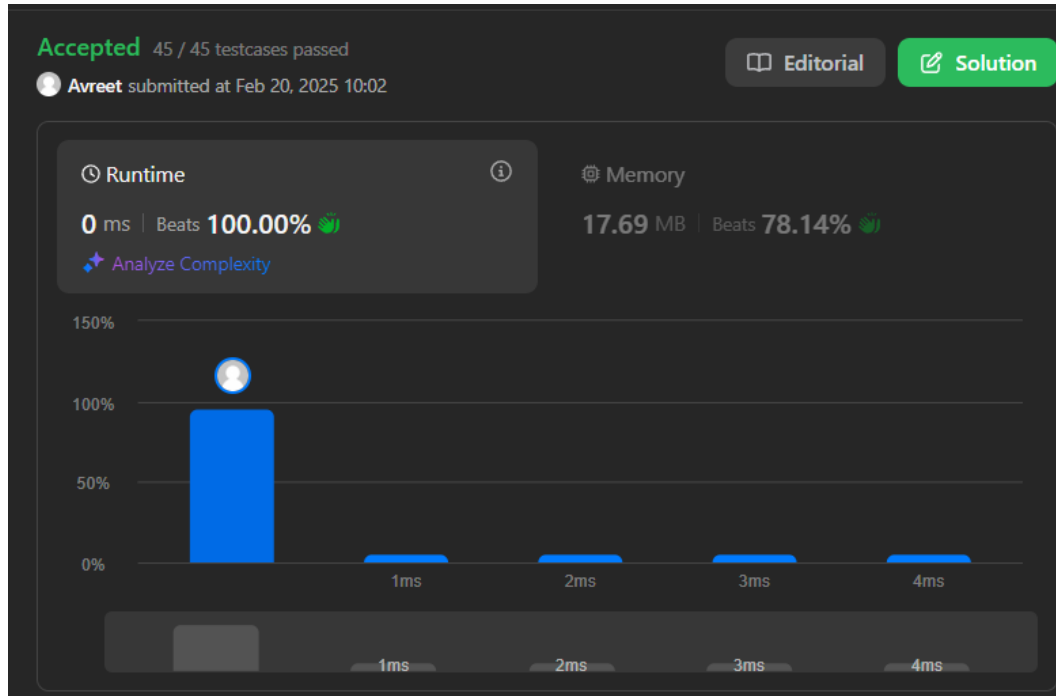
```
class Solution:
    def climbStairs(self, n: int) -> int:
        if n == 1:
            return 1
        a, b = 1, 2
        for _ in range(3, n + 1):
            a, b = b, a + b
        return b
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

### 3. Screenshots of outputs:



2.

**Aim:** You are given an array prices where prices[i] is the price of a given stock on the ith day. You want to maximize your profit by choosing a single day to buy one stock and choosing a different day in the future to sell that stock. Return the maximum profit you can achieve from this transaction. If you cannot achieve any profit, return 0.

### 4. Source Code:

```
class Solution:
    def maxProfit(self, prices: List[int]) -> int:
        min_price = float('inf')
        max_profit = 0
        for price in prices:
            min_price = min(min_price, price)
            profit = price - min_price
            max_profit = max(max_profit, profit)
        return max_profit
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 5. Screenshots of outputs:

