Experiment 7

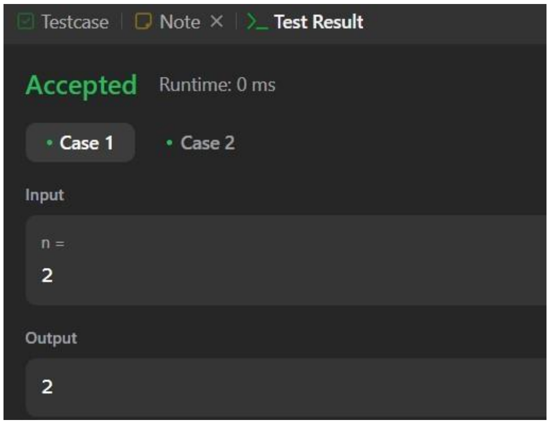
StudentName: Shubham **UID:**22BCS15853
Branch: CSE **Section/Group:** KRG 2 B
Semester: 6th **Date of Performance:**13/02/25
Subject Name: Advanced Programming Lab-2 **Subject Code:** 22CSP-351

Problem -1

1. **Aim:** Climbing Stairs
2. **Objective:**You are climbing a staircase. It takes n steps to reach the top.
3. **Implementation/Code:**

```
class Solution {
public:
    int climbStairs(int n) {
        if (n == 0 || n == 1) {
            return 1;
        }
        int prev = 1, curr = 1;
        for (int i = 2; i <= n; i++) {
            int temp = curr;
            curr = prev + curr;
            prev = temp;
        }
        return curr;
    }
};
```

4. **Output:**



Accepted Runtime: 0 ms

Input

n =
2

Output

2

Figure 1

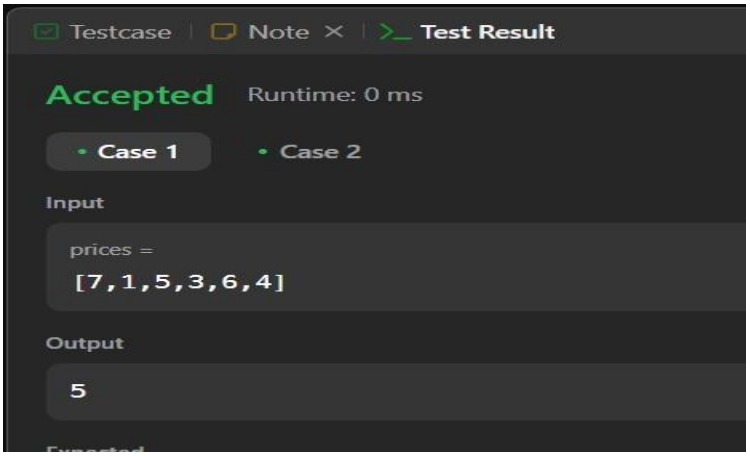
Problem-2

1. **Aim:** Best Time to Buy and Sell Stock
2. **Objectives:** You are given an array prices where prices[i] is the price of a given stock on the ith day.

3. **Implementation/Code:**

```
class Solution {
public:
    int maxProfit(vector<int>& prices) {
        int buy=prices[0];
        int profit=0;
        for(int i=0;i<prices.size();i++){
            if(prices[i]<buy){
                buy=prices[i];
            }
            else if(prices[i]-buy>profit){
                profit=prices[i]-buy;
            }
        }
        return profit;
    }
};
```

4. **Output:**



Accepted Runtime: 0 ms

Input

prices =
[7,1,5,3,6,4]

Output

5

Figure 2

Learning Outcomes:

- **Greedy Approach in Stock Trading** – Understanding how to track the minimum price and maximize profit efficiently.
- **Dynamic Programming in Climbing Stairs** – Recognizing the Fibonacci sequence application for optimized solutions.
- **Time Complexity Optimization** – Learning O(n) approaches for problems that could have been solved with brute force $O(n^2)$.
- **Efficient Memory Usage** – Using constant space (O(1)) instead of additional arrays or recursion stacks.
- **Iterative Problem-Solving** – Implementing loops effectively to avoid unnecessary computations.