



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment-3.3

Student Name: Nabha Varshney

UID: 20BCS4995

Branch: CSE

Section/Group: 20BCS-DM-704 (A)

Semester: 6th

Date of Performance: 10th May 2023

Subject Name: Competitive Coding II

Subject Code: 20CSP- 351

Aim – To demonstrate the concept of Dynamic Programming

Objective-

- The objective is to build problem solving capability and to learn the basic concepts of data structures.
- The implementation of climbing stairs using dynamic programming.
- The implementation of best time to buy and sell the stock.

1) Best time to buy and sell the stock

<https://leetcode.com/problems/best-time-to-buy-and-sell-stock/>

Code –

```
class Solution {
public:
    int maxProfit(vector<int>& prices) {
        int n = prices.size();
        int buyAtPrice = INT_MAX;
        int maxProfit = 0;
        for(int i=0; i<n; i++){
            buyAtPrice = min(buyAtPrice, prices[i]);
            maxProfit = max(maxProfit, prices[i] - buyAtPrice);
        }
        return maxProfit;
    }
};
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Output -

The screenshot shows a web browser with multiple tabs open, including 'Best Time to Buy and Sell Stock', 'Climbing Stairs - LeetCode', and 'Firewall Authentication Keepalive'. The active tab is 'Climbing Stairs - LeetCode', displaying the LeetCode interface for the problem 'Best Time to Buy and Sell Stock II'. The submission is accepted, with a runtime of 159 ms and memory usage of 93.3 MB. The code is written in C++ and is shown in the 'Details' panel on the right.

```
class Solution {
public:
    int maxProfit(vector<int>& prices) {
        int n = prices.size();
        int buyAtPrice = INT_MAX;
        int maxProfit = 0;
        for(int i=0; i<n; i++){
            buyAtPrice = min(buyAtPrice, prices[i]);
            maxProfit = max(maxProfit, prices[i] - buyAtPrice);
        }
        return maxProfit;
    }
};
```

2) Climbing Stairs

<https://leetcode.com/problems/climbing-stairs/>

Code -

```
class Solution {
public:
    int climbStairs(int n) {
        int a=0 ,b=1;
        for(int i=0;i<n;i++){
            int temp=a+b;
            a=b;
            b=temp;
        }
        return b;
    }
};
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Output –

The screenshot shows a web browser displaying the LeetCode submission page for the 'Climbing Stairs' problem. The page is in dark mode. The left sidebar shows the problem description, editorial, solutions (11.4K), and submissions. The main content area shows the submission details for user 'DEATHTRADER' on May 06, 2023, at 23:34. The submission is accepted in C++ with a runtime of 0 ms, 100% beats, and 6 MB memory. The code is a recursive solution. The right sidebar shows the user's profile, a distribution chart, and a code editor.

LeetCode

Problem List

Premium

Accepted

Next question

More challenges

- 746. Min Cost Climbing Stairs
- 509. Fibonacci Number
- 1137. N-th Tribonacci Number

All statuses

All languages

Accepted in a few seconds

C++

DEATHTRADER

May 06, 2023 23:34

Details

+ Solution

C++

Runtime 0 ms

Beats 100%

Memory 6 MB

Beats 51.37%

Click the distribution chart to view more details

Notes

Write your notes here

Related Tags

Select tags 0/5

```
class Solution {
public:
    int climbStairs(int n) {
        int a=0 ,b=1;
        for(int i=0;i<n;i++){
            int temp=a+b;
            a=b;
            b=temp;
        }
    }
};
```

Console

Run

Submit