**Data Structures and Algorithms**

**Exercise 7: Financial Forecasting**

**Steps1:**

**Explain the concept of recursion and how it can simplify certain problems**

Recursion is a programming technique where a function solves a problem by breaking it down into smaller instances of the same problem and calling itself to solve these sub-problems. At its core, recursion consists of two essential components:

**Base Case:** The simplest, non-decomposable scenario that terminates the recursion (e.g., forecasting for 0 periods returns the current value).

**Recursive Case:** The function calls itself with a modified input (e.g., reducing the forecasting period by 1) to progress toward the base case.

**Step 4:**

**Discuss the time complexity of your recursive algorithm.**

Time Complexity

O(n) linear time, where n is the number of periods. Each recursive call processes one period, leading to n stack frames.

Example: Forecasting 10,000 periods requires 10,000 recursive calls

**Explain how to optimize the recursive solution to avoid excessive computation.**

If we use iterative method we can save the problem of stack overhead because when the input is very big the recursive calls will take time to return those calls and overall time complexity can increase.

**OUTPUT:**

