**FINANCIAL\_FORECASTING**

1.Understanding the problem

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

Recursion:

**Concept**: Recursion is a programming method where a function calls itself, either directly or indirectly, to solve a problem. Each recursive call addresses a smaller or simpler version of the original problem, and the overall solution is constructed from these simpler solutions.

**Base Case and Recursive Case:** A recursive function must include at least one base case that terminates the recursion to prevent infinite loops and those base cases can be assigned based on the problem statements.

4.Analysis

* Discuss the time complexity of your recursive algorithm.

The time complexity of a recursive algorithm is O(n), where n represents the number of recursive calls. In this case, the function generates a recursive call for each period, leading to a total of n recursive calls.

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

Optimizing the Recursive Solution:

**1. Memoization:** To prevent redundant calculations in recursive algorithms, memoization can be employed. This technique involves storing the results of previous function calls and reusing them when the same inputs occur again, thus reducing unnecessary computations.

**2. Iterative Approach:** Converting a recursive solution to an iterative one can enhance performance by eliminating the overhead associated with recursive calls. This can be especially useful when the recursion depth is large.