Exercise 7: Financial Forecasting

1. Understanding Recursive Algorithms:

Recursion is a programming technique where a function calls itself to solve a problem by breaking it down

into smaller subproblems. This approach is especially useful for tasks that exhibit the property of self-similarity.

Key Concepts:

- Base Case: A condition under which the recursion stops.

- Recursive Case: A part of the function that calls itself with a modified input.

Benefits:

- Reduces code complexity for problems naturally defined in terms of smaller subproblems.

- Enables elegant solutions to divide-and-conquer problems like tree traversals, sorting, and forecasting.

Drawbacks:

- Every recursive call consumes stack memory, which may lead to stack overflow if not managed properly.

- Inefficient for overlapping subproblems unless optimized (e.g., using memoization).

Application in Forecasting:

- Financial forecasting using recursion models future values based on past values and a constant or changing rate.

- Recursion simplifies the implementation of compound interest models where each future value depends on the previous one.