**Financial Forecasting**

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

Recursion is a programming technique where a function calls itself in order to solve a problem. This approach simplifies problems by breaking them down into smaller, more manageable sub-problems that are easier to solve. Each recursive call tackles a smaller instance of the original problem until a base case is reached, which is directly solvable without further recursion.

Recursion is particularly useful for problems that exhibit self-similar structure, such as computing factorials, traversing tree structures, and solving puzzles like the Tower of Hanoi. By allowing a problem to be solved in terms of smaller sub-problems, recursion can lead to more elegant and concise solutions.

**Analysis**

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

The time complexity of the recursive algorithm is O(n), where n is the number of periods. This is because the function makes a recursive call for each period, resulting in n recursive calls in total.

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

=>Optimizing the Recursive Solution: 1.Memoization: To avoid redundant computations in recursive algorithms, memoization can be used. Memoization involves storing the results of expensive function calls and reusing the stored results when the same inputs occur again. 2.Iterative Approach: Converting the recursive solution to an iterative one can improve performance and avoid the overhead of recursive calls.