**Financial Forecasting**

1. Concept of Recursion

Recursion is a programming technique where a function calls itself in order to solve smaller instances of the same problem. The key components of a recursive function are:

* Base Case: The condition under which the recursion stops. This prevents the function from calling itself indefinitely.
* Recursive Case: The part where the function calls itself with a modified argument, progressing towards the base case.

1. How Recursion Can Simplify Certain Problems:  
   Recursion simplifies problems by breaking them down into smaller, more manageable sub-problems. It’s especially useful for problems that can be divided into similar sub-problems, such as calculating factorials, traversing hierarchical structures (like file systems), or solving problems in dynamic programming.
2. Time Complexity:  
   The time complexity of the recursive algorithm is O(n), where n is the number of periods. This is because the function makes a single recursive call for each period, and there are n such calls.
3. Optimizing the Recursive Solution:

To optimize the recursive solution and avoid excessive computation, you can use memoization or iterative approaches.

* Memoization: Store results of sub-problems in a cache so that they don’t need to be recomputed. This is especially useful for problems with overlapping sub-problems. In the context of forecasting, memoization may not be necessary because each period is distinct, but it is a general optimization technique for recursive algorithms.
* Iterative Approach: For this particular problem, an iterative approach can be more efficient and straightforward. Instead of using recursion, you could use a loop to calculate the future value.