**Data structures and Algorithms**

**Exercise 7: Financial Forecasting**

**Solution :**

**Step 1 : Understand Recursive Algorithms**

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

* **Recursion** is a method where a function calls itself to solve smaller parts of the same problem, and it continues until it reaches a stopping condition called the base case.
* It helps simplify problems like calculating factorial, Fibonacci series, or financial forecasting by writing cleaner and shorter code instead of using complex loops.
* Recursion is useful when a problem has a repeated pattern, but it should be used carefully to avoid too many function calls that can slow down the program or use too much memory.

**Step 2 : Setup (in code folder)**

**Step 3 : Implementation (in code folder)**

**Step 4 : Analysis**

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

* The time complexity of the recursive algorithm for financial forecasting is O(n), where n is the number of years.
* This is because the function calls itself once for each year, reducing the number of years by 1 in every step. So, if we want to calculate the future value for 15 years, the function will run 15 times.
* Since each call does a simple calculation and there are n calls in total, the overall time taken grows linearly with the number of years.

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

* Recursive solutions are easy to write, but they can become slow and use a lot of memory when the number of recursive calls is large. To avoid excessive computation, we can optimize the recursive algorithm in the following ways:

1. **Use Iteration Instead of Recursion :** Replace the recursive method with a loop. Iterative solutions are faster and more memory-efficient because they don’t use the call stack. For example, using a for loop to calculate future value is better than calling a function multiple times.
2. **Use Memoization :**If the problem involves repeating the same calculations (like in Fibonacci series), we can store already-computed results in a table and reuse them. This avoids duplicate work and makes the program faster.

* By using these techniques, we make the solution more efficient, especially for large inputs. In financial forecasting, an iterative method is usually the best option for performance and reliability.