**Exercise 7: Financial Forecasting**

Recursion is a programming technique where a method calls itself to solve smaller instances of the same problem. It is particularly useful for problems that can be broken down into similar subproblems. In financial forecasting, recursion helps to model compound growth year by year. For example, to calculate the future value of an investment, we can compute the value for one year and recursively call the same method for the remaining years.

**Time Complexity of Recursive Algorithm**

The basic recursive function for financial forecasting has a linear time complexity of O(n), where n is the number of years. This is because each recursive call handles one year of growth and continues until it reaches year 0. There is no repeated calculation, so the function completes in a single pass through the years.

**Optimization of Recursive Solution**

Although the simple recursive solution is efficient enough for this problem, it can be optimized further in scenarios involving overlapping subproblems. This is done using memorization, where results of previous calculations are stored and reused. By caching values already computed for specific years, the function avoids redundant work, especially if the same values are needed multiple times in complex forecasts. This improves performance and reduces the number of recursive calls.