**Exercise 7: Financial Forecasting**

**Scenario:**

You are developing a financial forecasting tool that predicts future values based on past data.

**Understanding Recursive Algorithms:**

**Recursion:** Recursion is a method where the solution to a problem depends on solutions to smaller instances of the same problem. A recursive function calls itself with smaller inputs to solve the overall problem.

Recursion simplifies problems that can be broken down into smaller, similar problems. It is commonly used in algorithms like factorial computation, Fibonacci series, and tree traversals.

**Analysis:**

* **Time Complexity:**
  + **Recursive Algorithm:** The time complexity of the naive recursive algorithm is O(n), where n is the number of years. This is because the algorithm calls itself once for each year.
  + **Optimized Recursive Algorithm (with Memoization):** The time complexity is also O(n), but it avoids redundant calculations by storing previously computed results. This makes it more efficient in practice, especially for larger values of n.
* **Optimization:**
  + **Memoization:** Memoization is a technique where you store the results of expensive function calls and reuse them when the same inputs occur again. This avoids the excessive computation that would otherwise occur with naive recursion.
  + **Bottom-Up Approach:** Alternatively, an iterative bottom-up approach could be used to calculate the future value, which also avoids the overhead of recursive calls.