**Exercise 7: Financial Forecasting**

**1. Understand Recursive Algorithms:**

* **Concept of Recursion:**
  + Recursion is a programming technique where a function calls itself to solve a smaller instance of the problem.
  + It is useful for problems that can be broken down into similar subproblems.
  + Recursive solutions are often more readable and concise compared to iterative ones.
* **Use in Financial Forecasting:**
  + Recursion can be used for forecasting models like Fibonacci series, compound interest calculation, or modeling time-series predictions.

**4. Analysis:**

* **Time Complexity of Recursive Algorithms:**
  + Depends on the problem and how many times the function calls itself.
  + For example, a naive recursive Fibonacci algorithm has exponential time complexity O(2^n).
  + Inefficient recursive algorithms can cause repeated computations and high execution time.
* **Optimization Techniques:**
  + **Memoization:** Store the results of subproblems to avoid recalculating them (top-down dynamic programming).
  + **Tabulation:** Use bottom-up dynamic programming to build the solution iteratively.
  + **Tail Recursion:** In some cases, modifying recursion to tail-recursive form can help compilers optimize the function calls.
  + By applying these techniques, recursive solutions can be transformed to run in linear or even constant time depending on the scenario.