# Exercise 7: Financial Forecasting (Theory Answers)

## 🔹 Step 1: Understand Recursive Algorithms

### 🔸 What is recursion, and how can it simplify certain problems?

Recursion is a programming technique where a function calls itself to solve a smaller instance of the same problem. It is particularly useful when a problem can be broken down into similar sub-problems.  
  
Recursion can simplify problems like financial forecasting by:  
- Eliminating the need for loops in repeated calculations.  
- Providing a clear and elegant solution when the problem has a naturally recursive structure.  
- Allowing developers to focus on the logic rather than the control flow.

## 🔹 Step 4: Analysis

### 🔸 What is the time complexity of a typical recursive algorithm used here?

In the context of financial forecasting using simple compound interest logic via recursion, the time complexity is typically O(n), where n is the number of years. This is because the function is called once per year, making it a linear recursive process.

### 🔸 How can the recursive solution be optimized to avoid excessive computation?

To optimize the recursive solution and avoid excessive computation:  
- Use  **memorization**  to store already computed results (if subproblems overlap).  
- Convert the recursion into an  **iterative** solution for better performance and to avoid stack overflow.  
- Tail recursion (if supported by the language/compiler) can also help in optimizing recursive calls.  
  
However, for straightforward compound calculations (like applying a growth rate yearly), recursion is already efficient since it performs a fixed number of steps.