**Exercise 7: Financial Forecasting**

**Recursion Concept**

Recursion is a programming technique where a method calls itself to solve smaller instances of the same problem. It simplifies problems by:

* Breaking complex problems into smaller, identical subproblems
* Using a base case to terminate the recursion
* Building up solutions from the simplest case

For financial forecasting, recursion is particularly useful for:

* Calculating compound growth over multiple periods
* Modeling scenarios where each period's growth depends on previous periods
* Implementing flexible forecasting models that can adapt to different time horizons

**Time Complexity Analysis**

1. **Basic Recursive Solution**:
   * Time Complexity: O(2^n) - Exponential due to redundant calculations
   * Space Complexity: O(n) - Call stack depth
2. **Optimized Solution (Memoization)**:
   * Time Complexity: O(n) - Each period calculated once
   * Space Complexity: O(n) - Storage for memoization
3. **Alternative Approaches**:
   * Iterative solution: O(n) time, O(1) space
   * Formula-based solution: O(1) for both (using Math.pow)

Optimization Techniques

1. **Memoization**:
   * Stores previously computed results to avoid redundant calculations
   * Dramatically improves performance for recursive solutions

**Output**



