**Financial ForeCasting**

**1.Understand Recursive Algorithms**

**Recursion Overview:**

* **Concept:** Recursion is a programming technique where a function calls itself to solve smaller instances of the same problem. It is often used to break down complex problems into simpler ones.
* **Components:**
  + **Base Case:** The condition under which the recursion stops. It prevents infinite recursion and provides the result for the simplest instance of the problem.
  + **Recursive Case:** The part where the function calls itself with a modified argument to work towards the base case.

**Example of Recursion:**

* **Factorial Calculation:** n!=n×(n−1)! with the base case 0!=1.

**2. Setup**

**Define the Problem:**

* We want to predict the future value of an investment based on its past growth rate. This can be done using a recursive approach where we predict the value for each future period.

**Formula for Future Value Calculation:** Future Value=Current Value×(1+Growth Rate) power of n , Where n is the number of periods.

**4.Analysis**

**Time Complexity of Recursive Algorithm:**

* **Time Complexity:** The time complexity of the calculateFutureValue function is O (n), where n is the number of periods. This is because the function makes n recursive calls.
* **Space Complexity:** The space complexity is also O (n) due to the stack space used by recursive calls. Each call adds a new frame to the call stack.

**Optimizing Recursive Solution:**

1. **Memoization:**
   * Store the results of previously computed values to avoid redundant calculations. This is particularly useful for problems where the same sub problems are solved multiple times.
2. **Iterative Solution:**
   * For this specific problem, an iterative approach could be more efficient. Here’s how you might implement it: