**1.Understand Recursive Algorithms**

***Q. Explain the concept of recursion and how it can simplify certain problems.***

Concept of Recursion:

Description: It is a technique where a function calls itself either directly or indirectly in order to solve the problem. It breaks down problems into smaller subproblems, solves each of the subproblems recursively, and finally solves the original problem.

Advantages:

Simplify the code for naturally recursive structured problems, for example, tree traversing and calculation of factorial.

Fewer needs for iterative loops, hence making the code more compact and readable.

Disadvantages:

Unless implemented correctly, this may result in excess memory usage with a potential stack overflow. May be of higher time complexity if overlapping sub-problems are not optimized.

**2.Analysis:**

***Q1. Discuss the time complexity of your recursive algorithm.***

**Time Complexity:**

* **Description:** The time complexity of the recursive algorithm is O(n), where n is the number of years. This is because each recursive call decreases the number of years by one until it reaches zero.
* **Optimized Approach:** To avoid excessive computation and stack overflow in deeper recursions, we can use memoization or convert the recursive solution to an iterative one.

***Q2. Explain how to optimize the recursive solution to avoid excessive computation.***

Memoization:

Stores previously computed results to avoid redundant calculations.

It uses a data structure, like HashMap, for result caching. This improves efficiency in cases of overlapping subproblems.

Iterative Approach:

Replaces recursion with loops for computing results; this reduces function call overhead.

This approach is memory efficient, faster, especially for large inputs, and avoids stack overflow issues.

Usage Recommendations:

Memoization: When a recursive approach is to be maintained, and there is a need to avoid redundant calculations.

Iterative approach: When better performance is required and the recursion depth may cause stack-overflow.

The redundant computation in financial forecasting is reduced by both methods, thus making the calculation of financial forecasting more efficient.