Recursion is a method of solving a problem where a function calls itself as a subroutine. This allows the function to be repeated several times as it can call itself during its execution. Recursion simplifies problems by breaking them down into smaller, more manageable sub-problems.

**Time Complexity:**

* **Simple Recursion**: The time complexity is O(n) because each call reduces the problem size by 1 until it reaches the base case. However, without memoization, the same sub-problems may be solved multiple times.
* **Optimized Recursion (Memoization)**: The time complexity is O(n) because each value is computed once and stored in the memo array, avoiding redundant calculations.

**Optimization:**

* **Memoization**: Storing the results of expensive function calls and reusing them when the same inputs occur again.
* **Avoiding Excessive Computation**: By using memoization, we ensure that each unique sub-problem is solved only once, significantly reducing the number of computations and improving efficiency.

This approach provides a clear example of how recursion can be used in financial forecasting and how optimization techniques like memoization can enhance performance.