### Recursive Algorithms

Concept of Recursion:

- Definition: Recursion is a method where the solution to a problem depends on solutions to smaller instances of the same problem. A recursive function calls itself with modified parameters until a base case is reached.

- Advantages: Simplifies problems that can be broken down into similar subproblems, such as calculating factorials, traversing trees, or implementing certain algorithms.

### Analysis of Recursive Algorithm for Financial Forecasting

Time Complexity:

- The recursive method `calculateFutureValue` has a time complexity of O(n), where `n` is the number of years. This is because the function makes a recursive call for each year, leading to a linear number of calls.

Optimization:

- Memoization: To avoid recalculating results for the same parameters, you can use memoization to store intermediate results. This is not necessary for this specific problem since each recursive call has unique parameters, but it is useful for more complex recursive problems where the same subproblems recur.

- Iterative Approach: An iterative solution can be more efficient for this problem, as it avoids the overhead of recursive function calls and reduces the time complexity to O(n) with a constant space requirement.

- Time Complexity: O(n), which is the same as the recursive approach.

- Space Complexity: O(1), as it uses only a fixed amount of additional space.

### Conclusion

Recursion can simplify the implementation of algorithms but may not always be the most efficient solution. In this case, while the recursive method provides a clear solution, the iterative method is more efficient in terms of space usage.