**Recursion:**

Recursion is a programming technique where a function calls itself, directly or indirectly, to solve a problem. It breaks down a problem into smaller, similar sub-problems until a base case is reached. The base case is a simple, non-recursive condition that stops the recursion, preventing an infinite loop.

* How it Simplifies Certain Problems:

1. Elegance and Readability: For problems with naturally recursive structures (e.g., factorials, Fibonacci sequences, tree traversals), recursive solutions can be more intuitive and concise.
2. Divide and Conquer: It naturally lends itself to problems that can be broken into smaller, identical sub-problems.
3. Mirroring Mathematical Definitions: Many mathematical formulas are inherently recursive, making their direct implementation straightforward using recursion.

**Time Complexity of the Recursive Algorithm:**

The recursive algorithm implemented (`calculateFutureValueRecursive`) has a time complexity of O(n), where 'n' is the number of 'periods'.

**Optimize the Recursive Solution to Avoid Excessive Computation:**

In the financial forecasting recursive algorithm, optimization is essential to prevent excessive computation and memory usage. The recursive method calculates the future value by calling itself for each preceding year, which leads to **O(n) time and O(n) space complexity** due to the call stack. Although the calculation is straightforward, it becomes inefficient for large values of n.

To optimize this, we can use an **iterative approach**, which performs the same computation using a simple loop. This removes the overhead of recursive function calls and reduces the **space complexity to O(1)**. The iterative version multiplies the value year-by-year in a loop, making it more efficient and safer for large forecasts.

Alternatively, **memoization** could be applied if the problem had overlapping subproblems, but for a strictly linear financial model, **iteration is the preferred optimization** for better performance and scalability.