**What is Recursion?**

Recursion is a programming technique where a method calls itself to solve smaller instances of the same problem.

**Example Use Cases:**

* Fibonacci sequence
* Tree traversal
* Financial forecasting (predicting next value based on previous growth)

**Why Use Recursion?**

* Simplifies complex problems.
* Makes code more readable for problems with repetitive structure.
* Useful when future states depend on past states.

## ****Analysis****

### ****Time Complexity****

* Recursive function is called once per year: **O(n)** where n is the number of years.
* No overlapping sub problems, so it’s linear and manageable.

### **Optimization (When Needed)**

* If the logic becomes more complex (e.g., with variable rates or multiple dependencies), **memorization** or **iterative methods** can reduce repeated computations.
* **Tail recursion** **or dynamic programming** can also be used for efficiency and stack optimization.