**Financial Forecasting**

**Recursive Algorithms**

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

**How it can simplify certain problems?**

Recursion simplifies certain problems by breaking them down into smaller, similar subproblems. Instead of writing complex loops or managing multiple variables, you let the function call itself with simpler inputs until it reaches a base case.

**Used in :** factorial, Fibonacci, tree/graph traversals, etc.,

**Time Complexity**

public static double futureValue(double principal, double rate, int years) {

if (years == 0) {

return principal;

}

return futureValue(principal, rate, years - 1) \* (1 + rate);

}

In here the recursion is depends on years. The recursive method calls itself years times.

**Time Complexity : O(n), Where n = years**

**Optimized Solution**

****Iterative Approach:****For large years, recursion can cause stack overflow. An iterative solution is more efficient and avoids stack issues

double result = principal;

for (int i = 0; i < years; i++) {

result \*= (1 + rate);

}

****Direct Formula:****For best performance, use the mathematical formula:

result = principal \* Math.pow(1 + rate, years)