# Week 1

## Algorithms and Data Structures

## Exercise 7: Financial Forecasting

Recursion is a method of solving a problem where the solution depends on solving smaller instances of the same problem.

A recursive function:

- Calls itself.
- Has a base case (to stop recursion).
- Has a recursive case (to reduce the problem size).

Why use recursion?

- Simplifies complex problems (e.g., Fibonacci, Tree Traversal).
- Natural fit for problems with repeated sub-structure.

### Future Value Formula (for compound growth)

$$FV=PV\times(1+r)^n$$

Where:

- FV = Future Value
- PV = Present Value (initial amount)
- r = annual growth rate (in decimal)
- n = number of years

### How to Optimize?

This recursion is already tail-recursive (no repeated subproblems), so it is efficient enough for small n.

### Optimization Techniques:

- **Memoization:** Useful if results are reused (not needed here).
- **Iteration:** For large n, recursion could be replaced with iteration to avoid stack overflow.

# Output snapshot: