Recursion — *A problem-solving technique whereby an original problem is repeatedly reduced to smaller instances of itself until a base case is reached.*

Two Ways to Classify Recursion

1. Chain Recursion vs. Tree Recursion
2. Incremental vs. Divide-and-Conquer

Typical Resource Usage

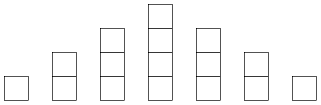
	chain	tree
incremental	N	2^N
divide-and-conquer	$\log N$	$N \log N$

Resource means memory or run-time.

1. Chain vs. Tree Recursion

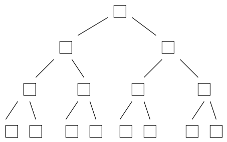
Chain Recursion: At each step, the problem is reduced to *exactly one* smaller instance of itself, plus some trivial work. Example:

$$\text{sum}(1, N) = \text{sum}(1, N-1) + N.$$



Tree Recursion: At each step, the problem is reduced to *at least two* smaller instances of itself, plus some trivial work. E.g.,

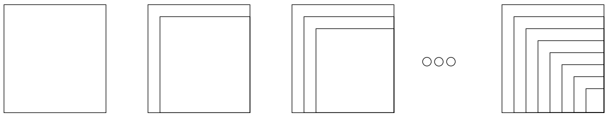
$$\text{fib}(N) = \begin{cases} \text{fib}(N-1) + \text{fib}(N-2) & \text{if} \quad N > 1 \\ N & \text{if} \quad N \leq 1 \end{cases}$$



2. Incremental vs. Divide-and-Conquer

Incremental: At each step, the size of the problem is reduced by *subtracting* a fixed amount from the given size. Example:

$$\text{sum}(1, N) = \text{sum}(1, N-1) + N.$$



Divide and Conquer: At each step, the size of the problem is reduced by *dividing* the given size by at least 2.
Examples: binary search, merge sort, GCD.