Elementary Algorithm Bootcamp

Week 1: Foundations & Problem-Solving Basics

Day 1: Bit Manipulation

- Binary number system refresher.
- Bitwise operators: AND, OR, XOR, NOT, shifts.
- Tricks: Check if a number is even/odd, swap numbers, find unique elements.
- Problems: Count set bits, single number (XOR), power of two.

Day 2: Binary Search

- Concept of "divide and conquer."
- Iterative vs. recursive implementation.
- Applications: Searching in sorted arrays, rotated arrays, and answer prediction (aggressive cows).
- Problems: Find first/last occurrence, search in rotated array.

Day 3: Recursion Basics

- Principle of recursion: base case, recursive case, call stack.
- Classic examples: Factorial, Fibonacci, Tower of Hanoi.
- Recursion tree visualization.
- Problems: Sum of digits, palindrome check, reverse a string.

Week 2: Hierarchical Structures & Advanced Recursion

Day 4: Recursion Deep Dive

- Backtracking: Generate permutations, subsets.
- Memoization to optimize recursion.
- Problems: N-Queens, combination sum, climbing stairs.

Day 5: Trees Introduction

- Terminology: Root, node, leaf, height, depth.

- Types: Binary trees, BSTs, balanced trees.

- Traversal: Pre-order, in-order, post-order (theory).

Day 6: Tree Operations

- Implement traversal algorithms (iterative & recursive).

- BST operations: Insert, delete, search.

- Problems: Validate BST, find min/max depth, invert a tree.

Week 3: Optimization & Dynamic Programming

Day 7: Tree Problem Solving

- Advanced problems: Lowest common ancestor, path sum, serialize/deserialize.

- Segment trees (optional time-permitting).

Day 8: Dynamic Programming (DP) Fundamentals

- Overlapping subproblems and optimal substructure.

- Top-down (memoization) vs. bottom-up (tabulation).

- Classic examples: Fibonacci, coin change.

Day 9: DP Techniques & Patterns

- Knapsack problem (0/1 and unbounded).

- Longest common subsequence (LCS), longest increasing subsequence (LIS).

- Problems: Edit distance, house robber, unique paths.

Week 4: Graph Theory & Final Challenges

Day 10: Advanced DP Problems

- Matrix chain multiplication.

- DP on trees: Maximum path sum.

- State transition practice.

Day 11: Graphs Introduction

- Terminology: Nodes, edges, directed vs. undirected graphs.

- Representations: Adjacency matrix vs. adjacency list.

- Traversal: BFS and DFS (theory + implementation).

Day 12: Graph Algorithms

- Shortest path: Dijkstra's algorithm (greedy), Bellman-Ford.

- Cycle detection, topological sorting.

- Problems: Number of islands, clone graph, course schedule.