DSA

FOUNDATIONAL CONCEPTS

1. Arrays & Hashing

Easy (2):

- Two Sum (1)
- Contains Duplicate (217)

Medium (4):

- Group Anagrams (49)
- Product of Array Except Self (238)
- Top K Frequent Elements (347)
- Longest Consecutive Sequence (128)

Hard (2):

- First Missing Positive (41)
- Largest Rectangle in Histogram (84)

2. Recursion

Easy (2):

- Factorial calculation
- Fibonacci Number (509)

Medium (4):

- Generate Parentheses (22)
- Pow(x, n) (50)
- Subsets (78)
- Merge Sort implementation

Hard (2):

- N-Queens (51)
- Expression Add Operators (282)

3. Binary Search

Easy (2):

- Binary Search (704)
- Search Insert Position (35)

Medium (4):

- Search in Rotated Sorted Array (33)
- Find Peak Element (162)
- Search a 2D Matrix (74)
- Koko Eating Bananas (875)

Hard (2):

- Median of Two Sorted Arrays (4)
- Split Array Largest Sum (410)

4. Linked Lists

Easy (2):

- Reverse Linked List (206)
- Merge Two Sorted Lists (21)

Medium (4):

- Remove Nth Node From End (19)
- Reorder List (143)
- Add Two Numbers (2)
- Copy List with Random Pointer (138)

Hard (2):

- Merge k Sorted Lists (23)
- Reverse Nodes in k-Group (25)

5. Stacks & Queues

Easy (2):

- Valid Parentheses (20)
- Implement Queue using Stacks (232)

Medium (4):

- Daily Temperatures (739)
- Evaluate Reverse Polish Notation (150)
- Decode String (394)
- Asteroid Collision (735)

Hard (2):

- Largest Rectangle in Histogram (84)
- Basic Calculator (224)

INTERMEDIATE CONCEPTS

6. Trees & Binary Search Trees

Easy (2):

- Maximum Depth of Binary Tree (104)
- Invert Binary Tree (226)

Medium (4):

- Binary Tree Level Order Traversal (102)
- Validate Binary Search Tree (98)
- Kth Smallest Element in BST (230)

- Construct Binary Tree from Preorder and Inorder (105)

Hard (3):

- Binary Tree Maximum Path Sum (124)
- Serialize and Deserialize Binary Tree (297)
- Binary Tree Postorder Traversal (145) - iterative

7. Two Pointers

Easy (2):

- Valid Palindrome (125)
- Two Sum II - Input Array Is Sorted (167)

Medium (4):

- 3Sum (15)
- Container With Most Water (11)
- Remove Duplicates from Sorted Array II (80)
- Sort Colors (75)

Hard (2):

- Trapping Rain Water (42)
- 4Sum (18)

8. Sliding Window

Easy (1):

- Best Time to Buy and Sell Stock (121)

Medium (5):

- Longest Substring Without Repeating Characters (3)
- Longest Repeating Character Replacement (424)
- Permutation in String (567)
- Minimum Size Subarray Sum (209)
- Fruit Into Baskets (904)

Hard (2):

- Minimum Window Substring (76)
- Sliding Window Maximum (239)

9. Fast & Slow Pointers

Easy (2):

- Linked List Cycle (141)
- Happy Number (202)

Medium (4):

- Linked List Cycle II (142)

- Find the Duplicate Number (287)
- Middle of the Linked List (876)
- Palindrome Linked List (234)

Hard (2):

- First Missing Positive (41)
- Circular Array Loop (457)

ADVANCED CONCEPTS

10. Dynamic Programming

Easy (2):

- Climbing Stairs (70)
- House Robber (198)

Medium (5):

- Coin Change (322)
- Longest Increasing Subsequence (300)
- Unique Paths (62)
- Word Break (139)
- Maximum Product Subarray (152)

Hard (3):

- Edit Distance (72)
- Regular Expression Matching (10)
- Best Time to Buy and Sell Stock IV (188)

11. Backtracking

Easy (1):

- Letter Case Permutation (784)

Medium (5):

- Permutations (46)
- Subsets (78)
- Combination Sum (39)
- Word Search (79)
- Palindrome Partitioning (131)

Hard (3):

- N-Queens (51)
- Word Search II (212)
- Sudoku Solver (37)

12. Graph Algorithms

Easy (2):

- Find the Town Judge (997)
- Number of Islands (200)

Medium (4):

- Course Schedule (207)
- Pacific Atlantic Water Flow (417)
- Clone Graph (133)
- Rotting Oranges (994)

Hard (2):

- Word Ladder II (126)
- Critical Connections in a Network (1192)

SPECIALIZED DATA STRUCTURES

13. Heap/Priority Queue

Easy (1):

- Last Stone Weight (1046)

Medium (4):

- Kth Largest Element in an Array (215)
- Top K Frequent Elements (347)
- Meeting Rooms II (253)
- Task Scheduler (621)

Hard (2):

- Merge k Sorted Lists (23)
- Find Median from Data Stream (295)

14. Trie

Medium (3):

- Implement Trie (208)
- Design Add and Search Words (211)
- Replace Words (648)

Hard (2):

- Word Search II (212)
- Palindrome Pairs (336)

15. Union Find (Disjoint Set)

Medium (3):

- Number of Connected Components in an Undirected Graph (323)
- Accounts Merge (721)
- Redundant Connection (684)

Hard (2):

- Most Stones Removed with Same Row or Column (947)
- Number of Islands II (305)

16. Segment Tree / Fenwick Tree

Medium (2):

- Range Sum Query - Mutable (307)
- Count of Smaller Numbers After Self (315)

Hard (2):

- Range Sum Query 2D - Mutable (308)
- Reverse Pairs (493)

REVISED 3-4 WEEK STUDY PLAN

Week 1: Foundations (Critical for everything else)

- **Days 1-2:** Arrays & Hashing, Two Pointers
- **Days 3-4:** Recursion, Binary Search
- **Days 5-7:** Linked Lists, Stacks & Queues

Week 2: Trees & Core Patterns

- **Days 1-3:** Trees & BST (crucial for many other problems)
- **Days 4-5:** Fast & Slow Pointers, Sliding Window
- **Days 6-7:** Basic DP problems

Week 3: Advanced Concepts (Your weak areas)

- **Days 1-4:** Dynamic Programming (intensive focus)
- **Days 5-7:** Backtracking

Week 4: Graphs & Integration

- **Days 1-3:** Graph algorithms
- **Days 4-7:** Heaps, advanced problems, mock interviews

PRIORITY TOPICS (Focus first 2 weeks)

1. Dynamic Programming

Easy (2):

- Climbing Stairs (70)
- House Robber (198)

Medium (5):

- Coin Change (322)
- Longest Increasing Subsequence (300)
- Unique Paths (62)
- Word Break (139)
- Maximum Product Subarray (152)

Hard (3):

- Edit Distance (72)
- Regular Expression Matching (10)
- Best Time to Buy and Sell Stock IV (188)

Pattern Recognition:

- Overlapping subproblems
- Optimal substructure
- State transition equations

2. Backtracking

Easy (1):

- Generate Parentheses (22) - though marked medium, good starter

Medium (5):

- Permutations (46)
- Subsets (78)
- Combination Sum (39)
- Word Search (79)
- Letter Combinations of a Phone Number (17)

Hard (3):

- N-Queens (51)
- Word Search II (212)
- Sudoku Solver (37)

Pattern Recognition:

- Choose → Explore → Unchoose
- Pruning conditions
- State space tree traversal

3. Graph Algorithms (Related to DP/Backtracking)

Easy (2):

- Flood Fill (733)
- Number of Islands (200)

Medium (4):

- Course Schedule (207)
- Pacific Atlantic Water Flow (417)
- Clone Graph (133)
- Rotting Oranges (994)

Hard (2):

- Word Ladder II (126)

- Alien Dictionary (269) - Premium

Pattern Recognition:

- BFS for shortest path
- DFS for connectivity
- Topological sort for dependencies

SUPPORTING TOPICS (Week 3-4)

4. Tree Algorithms

Easy (2):

- Maximum Depth of Binary Tree (104)
- Symmetric Tree (101)

Medium (4):

- Binary Tree Level Order Traversal (102)
- Construct Binary Tree from Preorder and Inorder (105)
- Validate Binary Search Tree (98)
- Lowest Common Ancestor of BST (235)

Hard (2):

- Binary Tree Maximum Path Sum (124)
- Serialize and Deserialize Binary Tree (297)

5. Two Pointers

Easy (2):

- Two Sum II (167)
- Valid Palindrome (125)

Medium (4):

- 3Sum (15)
- Container With Most Water (11)
- Remove Nth Node From End of List (19)
- Sort Colors (75)

Hard (2):

- Trapping Rain Water (42)
- Minimum Window Substring (76)

6. Sliding Window

Easy (1):

- Best Time to Buy and Sell Stock (121)

Medium (4):

- Longest Substring Without Repeating Characters (3)

- Minimum Size Subarray Sum (209)
- Longest Repeating Character Replacement (424)
- Permutation in String (567)

Hard (2):

- Sliding Window Maximum (239)
- Minimum Window Substring (76)

7. Binary Search

Easy (2):

- Binary Search (704)
- Search Insert Position (35)

Medium (4):

- Search in Rotated Sorted Array (33)
- Find Peak Element (162)
- Search a 2D Matrix (74)
- Koko Eating Bananas (875)

Hard (2):

- Median of Two Sorted Arrays (4)
- Split Array Largest Sum (410)

ADVANCED TOPICS (If time permits)

8. Heap/Priority Queue

Easy (1):

- Kth Largest Element in a Stream (703)

Medium (4):

- Kth Largest Element in an Array (215)
- Top K Frequent Elements (347)
- Meeting Rooms II (253)
- Reorganize String (767)

Hard (2):

- Merge k Sorted Lists (23)
- Find Median from Data Stream (295)

9. Trie

Medium (3):

- Implement Trie (208)
- Word Search II (212)
- Design Add and Search Words Data Structure (211)

Hard (2):

- Word Break II (140)
- Palindrome Pairs (336)

10. Union Find

Medium (3):

- Number of Connected Components (323)
- Accounts Merge (721)
- Redundant Connection (684)

Hard (2):

- Number of Islands II (305)
- Minimum Spanning Tree problems

3-4 Week Study Plan

Week 1: DP Foundation

- Day 1-2: Linear DP (Climbing Stairs, House Robber, Coin Change)
- Day 3-4: 2D DP (Unique Paths, Edit Distance)
- Day 5-7: Advanced DP (LIS, Word Break, Maximum Product)

Week 2: Backtracking + Graph

- Day 1-3: Basic Backtracking (Permutations, Subsets, Combinations)
- Day 4-5: Advanced Backtracking (N-Queens, Word Search II)
- Day 6-7: Graph fundamentals (BFS/DFS, Number of Islands)

Week 3: Trees + Two Pointers

- Day 1-3: Tree traversals and construction
- Day 4-5: BST problems
- Day 6-7: Two Pointers pattern

Week 4: Integration + Hard Problems

- Day 1-2: Sliding Window
- Day 3-4: Binary Search
- Day 5-7: Mix of hard problems from each topic

Daily Practice Structure:

1. **Morning (1.5 hours):** Focus topic problems
2. **Evening (1 hour):** Review and optimize solutions
3. **Weekend:** Mock interviews and hard problem sessions

Key for DP Success:

1. Always identify the state clearly
2. Write the recurrence relation
3. Implement top-down first, then bottom-up
4. Optimize space complexity

Key for Backtracking Success:

1. Draw the decision tree
2. Identify pruning conditions early
3. Practice the choose-explore-unchoose pattern
4. Time complexity analysis is crucial