DSA with Python - Roadmap

Phase 1: Basics & Foundations (2-3 weeks)

- Python syntax for DSA: input/output, lists, sets, dicts
- Time & Space Complexity basics
- Basic math: primes, GCD, LCM, factorial, etc.

* Practice 10-15 easy problems on LeetCode or GFG.

Phase 2: Arrays & Strings (3-4 weeks)

- Sliding Window, Two Pointers
- Prefix Sum, Kadane's Algorithm
- String manipulation, Anagrams, Palindromes

* Solve 50+ Easy to Medium problems

Phase 3: Recursion & Backtracking (2 weeks)

- Recursion basics: factorial, Fibonacci
- Backtracking: N-Queens, Subsets, Permutations

* Practice classic problems from LeetCode

Phase 4: Hashing, Stacks & Queues (2 weeks)

- HashMap, Set, frequency maps
- Stack: Valid Parentheses, NGE
- Queue & Deque: Sliding window problems

Phase 5: Linked List & Trees (3 weeks)

- Singly & Doubly Linked List
- Binary Tree & BST, Tree Traversals
- DFS/BFS on trees

* 30-40 problems total

DSA with Python - Roadmap

Phase 6: Heaps, Tries, Graphs (4 weeks)

- PriorityQueue (heapq), Top K problems
- Trie for prefix problems
- Graphs: BFS, DFS, Dijkstra, Union-Find

Phase 7: Dynamic Programming (4 weeks)

- 1D & 2D DP: Knapsack, LIS, LCS
- Memoization vs Tabulation
- Subset Sum, Coin Change

* 30+ classic DP problems

Final Phase: Revision & Mock Interviews

- LeetCode Top Interview 150
- Company-specific sheets
- Weekly timed contests