100 Days of DSA Challenge (Java) – Crack MAANG Interviews

Phase 1: Beginner (Days 1-30)

Java Fundamentals & Basic Data Structures

Week 1: Java Basics & Complexity Analysis

- **Day 1:** Introduction to Java (Syntax, Variables, Data Types, Operators)
- **Day 2:** Control Statements (Loops, Conditionals)
- **Day 3:** Functions & Recursion Basics
- **Day 4:** Arrays (1D & 2D) – Declaration, Initialization, Traversal
- **Day 5:** Time & Space Complexity Analysis
- **Day 6:** Basic Sorting Algorithms (Bubble, Selection, Insertion Sort)
- **Day 7:** Practice Problems (Easy Array & Sorting Questions)

Week 2: Advanced Arrays & Strings

- **Day 8:** Searching Algorithms (Linear & Binary Search)
- **Day 9:** Two Pointers & Sliding Window Techniques
- **Day 10:** String Manipulation (Java String Class, Methods)
- **Day 11:** Hashing (HashMaps & HashSets)
- **Day 12:** Prefix Sum & Kadane's Algorithm
- **Day 13:** More Sorting Techniques (Merge & Quick Sort)
- **Day 14:** Practice Problems (Medium Array & String Questions)

Week 3: Linked Lists & Recursion

- **Day 15:** Introduction to Linked Lists (Singly Linked List Implementation)
- **Day 16:** Doubly & Circular Linked Lists
- **Day 17:** Linked List Operations (Insertion, Deletion, Reversal)
- **Day 18:** Recursion – Understanding & Writing Recursive Code
- **Day 19:** Backtracking Basics
- **Day 20:** Linked List Problems (Middle, Detect Loop, Merge, etc.)
- **Day 21:** Practice Problems (Recursion & Linked List Questions)

Week 4: Stacks & Queues

- **Day 22:** Introduction to Stacks (Stack Implementation, Applications)
- **Day 23:** Stack Problems (Next Greater Element, Balanced Parentheses)
- **Day 24:** Queues & Deques (Implementation, Circular Queue)

- **Day 25:** Queue Applications (Sliding Window Maximum, LRU Cache)
- **Day 26:** Monotonic Stack & Queue Problems
- **Day 27:** Practice Problems (Stacks & Queues)
- **Day 28-30:** Revision & Mock Test 1

Phase 2: Intermediate (Days 31-70)

Trees, Graphs, and Advanced Algorithms

Week 5: Trees

- **Day 31:** Introduction to Trees (Binary Trees, Traversals)
- **Day 32:** Binary Search Trees (BST) – Insertion, Deletion, Search
- **Day 33:** Lowest Common Ancestor, Diameter of Tree
- **Day 34:** AVL Trees, Red-Black Trees (Conceptual Overview)
- **Day 35:** Problems on BSTs & Trees
- **Day 36:** Binary Tree to Doubly Linked List, Flattening a Tree
- **Day 37:** Practice Problems (Trees & BSTs)

Week 6-7: Graphs & Greedy Algorithms

- **Day 38:** Introduction to Graphs (Adjacency List, Matrix Representation)
- **Day 39:** BFS & DFS Traversal Techniques
- **Day 40:** Topological Sorting, Cycle Detection in Graphs
- **Day 41:** Dijkstra's Algorithm & Bellman-Ford Algorithm
- **Day 42:** Floyd-Warshall Algorithm & Minimum Spanning Tree
- **Day 43:** Graph Problems (Strongly Connected Components, Bridges)
- **Day 44:** Practice Problems (Graphs)
- **Day 45:** Greedy Algorithms – Basics & Problems
- **Day 46:** Huffman Encoding, Activity Selection Problem
- **Day 47:** Greedy Problems (Job Scheduling, Minimum Platforms)
- **Day 48:** Practice Problems (Greedy Algorithms)
- **Day 49-50:** Mock Test 2 & Review

Week 8-9: Dynamic Programming & Advanced Recursion

- **Day 51:** Introduction to Dynamic Programming (Memoization, Tabulation)
- **Day 52:** 0/1 Knapsack Problem
- **Day 53:** Longest Common Subsequence (LCS), LIS
- **Day 54:** DP on Strings (Edit Distance, Regular Expression Matching)

- **Day 55:** DP on Trees & Graphs
- **Day 56:** DP Problems (Subset Sum, Coin Change, etc.)
- **Day 57:** More DP Problems & Practice
- **Day 58:** Backtracking Advanced Problems (N-Queens, Sudoku Solver)
- **Day 59-60:** Mock Test 3 & Optimization Techniques

Phase 3: Advanced (Days 71-100)

Complex Topics, System Design, and Mock Interviews

Week 10: Advanced DSA Topics

- **Day 71:** Tries – Prefix Trees & Applications
- **Day 72:** Suffix Arrays & Suffix Trees
- **Day 73:** Segment Trees – Range Queries
- **Day 74:** Fenwick Trees (Binary Indexed Trees)
- **Day 75:** Graph Algorithms – Network Flow, Bipartite Matching
- **Day 76:** Practice Problems (Tries, Segment Trees, Graphs)
- **Day 77-78:** Mock Test 4 & Review

Week 11: System Design Basics

- **Day 79:** Basics of System Design – Scalability, Load Balancing
- **Day 80:** Database Scaling – SQL vs NoSQL, Sharding, Indexing
- **Day 81:** Caching Strategies (LRU, LFU, Redis Basics)
- **Day 82:** Message Queues & Rate Limiting
- **Day 83:** CAP Theorem, Consistency & Availability
- **Day 84:** Designing URL Shortener, Distributed Systems Basics
- **Day 85-86:** Mock Test 5 (System Design & DSA Combined)

Week 12-13: Mock Interviews & Company-Specific Questions

- **Day 87-90:** Company-Specific Problems (Amazon, Google, Meta, etc.)
- **Day 91-93:** Advanced Coding Questions & Optimizations
- **Day 94-96:** Full-Length Mock Interviews
- **Day 97-99:** Final Interview Preparation – Resume, Behavioral Questions
- **Day 100:** Final Mock Test & Roadmap for Continued Learning

---

Conclusion:

This structured plan ensures students progress step-by-step, gaining strong DSA skills and interview readiness for top tech companies. By the end, they'll be proficient in Java-based problem-solving and system design, ready for MAANG-level interviews!