100 Days of DSA Challenge - C++ Edition

Beginner Phase (Days 1-30) - C++ Basics & Core DSA Foundations

Week 1: C++ Fundamentals & Basic Programming Concepts

- **Day 1:** Introduction to C++ (Syntax, Variables, Data Types, I/O)
- **Day 2:** Operators, Conditional Statements (if-else, switch-case)
- **Day 3:** Loops (for, while, do-while), Pattern Printing
- **Day 4:** Functions & Recursion Basics
- **Day 5:** Arrays (1D & 2D), Introduction to Pointers
- **Day 6:** Strings & String Manipulation Techniques
- **Day 7:** Solving Basic Problems (Functions, Arrays, Strings)

Week 2: Introduction to Data Structures

- **Day 8:** Time Complexity Analysis (Big-O Notation)
- **Day 9:** Searching Algorithms (Linear & Binary Search)
- **Day 10:** Sorting Algorithms (Bubble, Selection, Insertion Sort)
- **Day 11:** Merge Sort & Quick Sort
- **Day 12:** Stack (Implementation using Arrays & STL)
- **Day 13:** Queue (Implementation using Arrays & STL)
- **Day 14:** Solving Problems on Stacks & Queues

Week 3: Linked Lists & Problem-Solving

- **Day 15:** Singly Linked List (Insertion, Deletion, Traversal)
- **Day 16:** Doubly & Circular Linked Lists
- **Day 17:** Problems on Linked Lists (Reversal, Cycle Detection, etc.)
- **Day 18:** Recursion Deep Dive - Backtracking Introduction
- **Day 19:** Two Pointers & Sliding Window Technique
- **Day 20:** Solving Problems using Two Pointers & Sliding Window
- **Day 21:** Recap & Coding Contest Practice

Week 4: Advanced Recursion & Hashing

- **Day 22:** Advanced Backtracking Problems (N-Queens, Sudoku Solver)
- **Day 23:** Hashing (Maps, Sets, Hash Tables)

- **Day 24:** Hashing-based Problems
- **Day 25:** Bit Manipulation & Important Tricks
- **Day 26:** Greedy Algorithms & Problems
- **Day 27:** Prefix Sum & Kadane's Algorithm (Max Subarray Sum)
- **Day 28:** Recap & Competitive Coding Practice
- **Day 29:** Mock Interview 1 (Easy Level Problems)
- **Day 30:** Beginner Phase Summary & Progress Check

Intermediate Phase (Days 31-70) - Trees, Graphs & Dynamic Programming

Week 5-6: Trees (BST & Binary Trees)

- **Day 31:** Introduction to Trees & Binary Tree Basics
- **Day 32:** Tree Traversals (Preorder, Inorder, Postorder, Level Order)
- **Day 33:** Binary Search Tree (BST) & Operations
- **Day 34:** Lowest Common Ancestor (LCA) & BST Problems
- **Day 35:** Height, Diameter, & Balanced Trees
- **Day 36:** Problems on Trees (Views, Zigzag Traversal, etc.)
- **Day 37:** Heaps & Priority Queues (Min & Max Heap)
- **Day 38:** Heap-based Problems (Top K Elements, Median in Stream)
- **Day 39:** Trie Data Structure (Implementation & Applications)
- **Day 40:** Solving Problems using Tries

Week 7: Graphs - BFS, DFS & Shortest Paths

- **Day 41:** Graph Representation (Adjacency List & Matrix)
- **Day 42:** Breadth-First Search (BFS) Algorithm
- **Day 43:** Depth-First Search (DFS) Algorithm
- **Day 44:** Cycle Detection in Graphs (Directed & Undirected)
- **Day 45:** Shortest Path Algorithms (Dijkstra's & Bellman-Ford)
- **Day 46:** Topological Sorting & Applications
- **Day 47:** Solving Problems on Graphs

Week 8-10: Dynamic Programming (DP) & Advanced Algorithms

- **Day 48:** Introduction to DP (Recursion + Memoization)
- **Day 49:** Fibonacci, Climbing Stairs Problem (Basic DP)

- **Day 50:** 0/1 Knapsack & Subset Sum Problem
- **Day 51:** Longest Common Subsequence (LCS)
- **Day 52:** DP on Strings (Edit Distance, Longest Palindromic Subsequence)
- **Day 53:** DP on Trees & Graphs
- **Day 54:** Matrix Chain Multiplication & Other DP Problems
- **Day 55:** Recap & DP-based Coding Practice
- **Day 56:** Segment Trees (Implementation & Applications)
- **Day 57:** Fenwick Tree / Binary Indexed Tree (BIT)
- **Day 58:** Range Query Problems (RMQ, Lazy Propagation)
- **Day 59:** Solving Problems on Segment Trees
- **Day 60:** Mock Interview 2 (Intermediate Level Problems)

Advanced Phase (Days 71-100) - Hard Problems, Mock Interviews & Optimization

Week 11-12: Advanced Topics & Optimization

- **Day 71:** String Algorithms (KMP, Rabin-Karp)
- **Day 72:** Bitwise Algorithms (XOR, Count Set Bits)
- **Day 73:** Advanced Graphs (Floyd Warshall, Prim's, Kruskal's Algorithm)
- **Day 74:** Disjoint Set Union (DSU) & Applications
- **Day 75:** System Design Basics (Load Balancing, Caching, Scaling)
- **Day 76:** Memory Management & Optimization Techniques
- **Day 77:** Recap & Competitive Programming Practice
- **Day 78:** Mock Interview 3 (Company-Specific Problems)

Week 13-14: Company-Specific & Final Mock Interviews

- **Day 79:** Google Interview Questions
- **Day 80:** Amazon Interview Questions
- **Day 81:** Meta/Facebook Interview Questions
- **Day 82:** Microsoft Interview Questions
- **Day 83:** Apple Interview Questions
- **Day 84:** Netflix Interview Questions
- **Day 85:** Solving Hardest Leetcode Problems
- **Day 86:** Mock System Design Interview
- **Day 87:** Code Optimization Techniques
- **Day 88:** Speed Coding & Time Management

- **Day 89:** Final Mock Interview 4 (Hard Level Problems)
- **Day 90:** Debugging Techniques & Best Practices
- **Day 91-99:** Daily Mock Interviews & Refining Skills
- **Day 100:** Final Assessment & Roadmap Beyond 100 Days

Conclusion & Next Steps

Congratulations on completing the 100 Days of DSA Challenge in C++! 🎉

By now, you have mastered essential data structures, algorithms, and problem-solving techniques, making you interview-ready for top tech companies. Keep practicing, participate in coding contests, and refine your skills further!

Good luck! 🚀