DSA 20-Day Challenge

Week 1 – Easy Foundation (Arrays & Strings)

Day 1: Find the Largest Element in an Array

Day 2: Check if Array is Palindrome

Day 3: Reverse a String (Without using built-in reverse())

Day 4: Count Frequency of Each Element in Array

Day 5: Move All Zeros to the End

Day 6: Find Second Largest Number

Day 7: Remove Duplicates from Sorted Array

Week 2 – Moderate Problems (Hashing, 2-Pointers, Searching)

Day 8: Two Sum Problem (Return indices)

Day 9: Longest Common Prefix

Day 10: Intersection of Two Arrays

Day 11: Majority Element (Moore’s Voting Algorithm)

Day 12: Check Anagram

Day 13: Subarray With Given Sum (Positive numbers)

Day 14: Missing Number (0 to n using XOR method)

Week 3 – Core DSA (Sorting, Recursion, Stacks)

Day 15: Merge Sort (Implement full algorithm)

Day 16: Find All Permutations of a String (Recursion)

Day 17: Valid Parentheses (Stack)

Day 18: Next Greater Element (Stack)

Day 19: Binary Search on Rotated Sorted Array

Day 20: Sliding Window — Longest Substring Without Repeating Characters