Google Apprenticeship Prep Guide 2025 - Daily Checklist

# Week 1 – Arrays & Strings

## Day 1

☐ Read: Big-O Notation basics

☐ Solve: Two Sum (https://leetcode.com/problems/two-sum/)

☐ Solve: Best Time to Buy and Sell Stock (https://leetcode.com/problems/best-time-to-buy-and-sell-stock/)

## Day 2

☐ Solve: Valid Anagram (https://leetcode.com/problems/valid-anagram/)

☐ Solve: Group Anagrams (https://leetcode.com/problems/group-anagrams/)

☐ Write notes: Hashing patterns

## Day 3

☐ Solve: Longest Substring Without Repeating Characters (https://leetcode.com/problems/longest-substring-without-repeating-characters/)

☐ Solve: Valid Palindrome (https://leetcode.com/problems/valid-palindrome/)

☐ Practice: Explain thought process aloud

## Day 4

☐ Solve: Product of Array Except Self (https://leetcode.com/problems/product-of-array-except-self/)

☐ Solve: Maximum Subarray (https://leetcode.com/problems/maximum-subarray/)

☐ Review Big-O

## Day 5

☐ Solve: Container With Most Water (https://leetcode.com/problems/container-with-most-water/)

☐ Solve: 3Sum (https://leetcode.com/problems/3sum/)

☐ Write notes on Two-Pointer pattern

## Day 6

☐ Solve: 2 random Easy from Arrays/Strings (LeetCode)

☐ Build: Simple Python/Java program → reverse string

## Day 7 (Review)

☐ Revise all problems solved

☐ Mock explain 'Two Sum' aloud

☐ Rest & reflect

# Week 2 – Linked Lists, Hashing, Stacks, Queues

## Day 8

☐ Solve: Reverse Linked List (https://leetcode.com/problems/reverse-linked-list/)

☐ Solve: Merge Two Sorted Lists (https://leetcode.com/problems/merge-two-sorted-lists/)

## Day 9

☐ Solve: Linked List Cycle (https://leetcode.com/problems/linked-list-cycle/)

☐ Solve: Remove Nth Node From End (https://leetcode.com/problems/remove-nth-node-from-end-of-list/)

## Day 10

☐ Solve: LRU Cache (https://leetcode.com/problems/lru-cache/)

☐ Understand: Hashmap + Doubly Linked List

## Day 11

☐ Solve: Valid Parentheses (https://leetcode.com/problems/valid-parentheses/)

☐ Solve: Min Stack (https://leetcode.com/problems/min-stack/)

## Day 12

☐ Write notes on Stack/Queue operations

☐ System Design: What is an API? Example REST call

## Day 13

☐ Build: To-Do CLI app (Python/Java) with add/remove tasks

☐ Push to GitHub

## Day 14 (Review)

☐ Revise all Linked List/Stack/Queue problems

☐ Mock explain LRU Cache aloud

# Week 3 – Trees & Binary Search

## Day 15

☐ Solve: Binary Tree Inorder Traversal (https://leetcode.com/problems/binary-tree-inorder-traversal/)

☐ Solve: Maximum Depth of Binary Tree (https://leetcode.com/problems/maximum-depth-of-binary-tree/)

## Day 16

☐ Solve: Validate BST (https://leetcode.com/problems/validate-binary-search-tree/)

☐ Solve: Lowest Common Ancestor of BST (https://leetcode.com/problems/lowest-common-ancestor-of-a-binary-search-tree/)

## Day 17

☐ Solve: Binary Search (https://leetcode.com/problems/binary-search/)

☐ Solve: Search in Rotated Sorted Array (https://leetcode.com/problems/search-in-rotated-sorted-array/)

## Day 18

☐ Solve: Find First and Last Position (https://leetcode.com/problems/find-first-and-last-position-of-element-in-sorted-array/)

☐ Review recursion patterns

## Day 19

☐ Learn: Tree traversal BFS vs DFS

☐ System Design: Database basics (SQL tables, keys)

## Day 20

☐ Practice SQL queries on HackerRank (SELECT, WHERE, GROUP BY)

## Day 21 (Review)

☐ Revise Trees & Binary Search problems

☐ Explain aloud: Binary Search with edge cases

# Week 4 – Graphs + Sorting

## Day 22

☐ Solve: Number of Islands (https://leetcode.com/problems/number-of-islands/)

☐ Solve: Clone Graph (https://leetcode.com/problems/clone-graph/)

## Day 23

☐ Solve: Course Schedule (https://leetcode.com/problems/course-schedule/)

☐ Solve: Pacific Atlantic Water Flow (https://leetcode.com/problems/pacific-atlantic-water-flow/)

## Day 24

☐ Solve: Merge Intervals (https://leetcode.com/problems/merge-intervals/)

☐ Solve: Insert Interval (https://leetcode.com/problems/insert-interval/)

## Day 25

☐ Solve: Meeting Rooms II (https://leetcode.com/problems/meeting-rooms-ii/)

☐ Review sorting algorithms

## Day 26

☐ System Design: Client-server architecture with Google Docs example

## Day 27

☐ Mini Project: Graph traversal visualizer (Python/Java text-based)

## Day 28 (Review)

☐ Revise all Graph & Sorting problems

☐ Mock explain BFS vs DFS

# Week 5 – Dynamic Programming (DP)

## Day 29

☐ Solve: Climbing Stairs (https://leetcode.com/problems/climbing-stairs/)

☐ Solve: House Robber (https://leetcode.com/problems/house-robber/)

## Day 30

☐ Solve: Coin Change (https://leetcode.com/problems/coin-change/)

☐ Solve: Longest Increasing Subsequence (https://leetcode.com/problems/longest-increasing-subsequence/)

## Day 31

☐ Solve: Longest Common Subsequence (https://leetcode.com/problems/longest-common-subsequence/)

☐ Solve: Edit Distance (https://leetcode.com/problems/edit-distance/)

## Day 32

☐ Review: Recursion → Memoization → Tabulation

## Day 33

☐ System Design: Caching (how login sessions work)

☐ Read about Redis basics

## Day 34

☐ Practice: Solve 2 medium DP problems of choice

## Day 35 (Review)

☐ Revise DP problems

☐ Explain aloud: Climbing Stairs with recursion & DP

# Week 6 – Full-Stack Basics + Mock Coding

## Day 36

☐ Solve: 3 random problems from previous weeks

☐ Review Arrays + Strings

## Day 37

☐ Learn HTML/CSS basics

☐ Build a simple webpage

## Day 38

☐ Learn JavaScript DOM basics

☐ Add interactivity to webpage

## Day 39

☐ Build simple React app (optional)

☐ System Design: URL shortener

## Day 40

☐ Database basics: Indexing & Joins

## Day 41

☐ Mini Project: Simple Blog app (Flask/Node.js) storing posts in SQLite/JSON

## Day 42 (Review)

☐ Revise GitHub workflow: Branch → PR → Merge

☐ Push project to GitHub

# Week 7 – Interview Simulation + Polishing

## Day 43

☐ Solve: 1 new LeetCode medium

☐ Review: 2 previous problems (explain aloud)

## Day 44

☐ Mock Interview: Pramp/ChatGPT roleplay

☐ System Design: To-Do App

## Day 45

☐ Solve: 1 easy + 1 medium problem

☐ Communication practice: Explain aloud

## Day 46

☐ Mock Interview: Interviewing.io (if available)

☐ System Design: E-commerce cart

## Day 47

☐ Solve: 2 random problems from Trees/Graphs

☐ Review Googly-ness qualities (collaboration, curiosity)

## Day 48

☐ Build: Mock PR review with friend or self-review on GitHub

## Day 49 (Review)

☐ Revise all solved problems

☐ Final mock interview practice

# Week 8 – Final Prep & Revision

## Day 50

☐ Revise Arrays, Strings, Hashing patterns

## Day 51

☐ Revise Linked Lists, Stacks, Queues

## Day 52

☐ Revise Trees & Binary Search

## Day 53

☐ Revise Graphs & Sorting

## Day 54

☐ Revise DP problems

☐ System Design final notes

## Day 55

☐ Revise Git, Collaboration skills

☐ Review resume & projects

## Day 56 (Final Review)

☐ Solve 2 easy problems only

☐ Rest, relax & prepare for interview