ChatGPT

1-Year FAANG Interview Prep Path (Pattern-Based)

This roadmap focuses on **pattern recognition**, **muscle memory**, and **gradual mastery**, progressing in 6 structured phases over the year.

Phase System (6 Phases)

Phase	Focus	Duration
1	Pattern Building (Foundations)	0–2 months
2	Mastery of Medium-Level Problems	2–4 months
3	Trees, Graphs, Union-Find	4–6 months
4	Hard Pattern Sets + DP	6–8 months
5	Mock, Review, Optimize	8–10 months
6	Interview Simulation	10–12 months

Phase 1: Pattern Building (0–2 months)

Solve 20 problems per pattern. Focus on mastering intuition and templates.

1. Two Pointers (20 Problems)

- 1. Two Sum II
- 1. Valid Palindrome
- 1. Move Zeroes
- 1. Reverse String
- 1. Container With Most Water
- 1. 3Sum
- 1. 4Sum
- 1. Remove Duplicates from Sorted Array
- 1. Remove Duplicates II
- 1. Sort Colors
- 1. Remove N-th Node From End
- 1. Linked List Cycle
- 1. Linked List Cycle II
- 1. Middle of the Linked List
- 1. Capacity To Ship Packages
- 1. Two Sum Less Than K

- 1. Interval List Intersections
- 1. Squares of a Sorted Array
- 1. Max Number of K-Sum Pairs
- 1. Count Subarrays With Fixed Bounds

2. Sliding Window (20 Problems)

- 1. Minimum Size Subarray Sum
- 1. Sliding Window Maximum
- 1. Longest Substring Without Repeating Characters
- 1. Minimum Window Substring
- 1. Find All Anagrams in a String
- 1. Permutation in String
- 1. Max Consecutive Ones III
- 1. Max Number of Vowels
- 1. Subarray Product Less Than K
- 1. Min Operations to Reduce X to Zero
- 1. Get Equal Substrings Within Budget
- 1. Length of Longest Subarray With At Most K Frequency
- 1. Maximum Number of Vowels
- 1. Find Median from Data Stream
- 1. Repeated DNA Sequences
- 1. Substring with Concatenation of All Words
- 1. Grumpy Bookstore Owner
- 1. Number of Subarrays of Size K and Avg > Threshold
- 1. Using a Sliding Window Over Array
- 1. Minimum Difference Between Highest and Lowest

3. HashMap / Frequency Count (20 Problems)

- 1. Two Sum
- 1. 4Sum II
- 1. Subarray Sum Equals K
- 1. Subarray Sums Divisible by K
- 1. Contiguous Array
- 1. Count Number of Nice Subarrays
- 1. Binary Subarrays With Sum
- 1. Valid Sudoku
- 1. Group Anagrams
- 1. Sort Characters By Frequency
- 1. Top K Frequent Elements
- 1. Ransom Note
- 1. Isomorphic Strings
- 1. Valid Anagram
- 1. Minimum Speed to Arrive on Time
- 1. Task Scheduler
- 1. Maximum Frequency Stack
- 1. First Completely Painted Row or Column

- 1. Frequency of the Most Frequent Element
- 1. Contiguous Array

4. Prefix Sum (20 Problems)

- 1. Range Sum Query
- 1. Subarray Sum Equals K
- 1. Subarray Sums Divisible by K
- 1. Count Number of Nice Subarrays
- 1. Product of Array Except Self
- 1. Binary Subarrays With Sum
- 1. Contiguous Array
- 1. Longest Well-Performing Interval
- 1. Arithmetic Subarrays
- 1. Sum of All Odd Length Subarrays
- 1. Running Sum of 1D Array
- 1. Sum of Absolute Differences
- 1. Shifting Letters II
- 1. Jump Game VII
- 1. Range Sum Query 2D
- 1. Maximal Square
- 1. Car Pooling
- 1. Find the Highest Altitude
- 1. Maximum Value of K Coins From Piles
- 1. Length of Longest Subarray With At Most K Frequency

5. Sorting + Greedy (20 Problems)

- 1. Non-overlapping Intervals
- 1. Minimum Number of Arrows to Burst Balloons
- 1. Meeting Rooms II
- 1. Merge Intervals
- 1. Two City Scheduling
- 1. Boats to Save People
- 1. Queue Reconstruction by Height
- 1. Task Scheduler
- 1. Candy
- 1. Assign Cookies
- 1. Can Place Flowers
- 1. Score After Flipping Matrix
- 1. Meeting Rooms II
- 1. Wiggle Subsequence
- 1. Partition Labels
- 1. Hand of Straights
- 1. Last Stone Weight
- 1. Gas Station
- 1. Best Time to Buy and Sell Stock II
- 1. Number of Subarrays of Size K and Avg > Threshold

Phase 2: Mastery of Medium-Level Problems (2–4 months)

Focus: Apply known patterns, avoid brute force, improve runtime.

1. Binary Search (20 Problems)

- 1. Search in Rotated Sorted Array
- 1. Find First and Last Position
- 1. Search in Rotated Sorted Array II
- 1. Find Minimum in Rotated Sorted Array
- 1. Find Peak Element
- 1. Koko Eating Bananas
- 1. Split Array Largest Sum
- 1. Search a 2D Matrix
- 1. Search a 2D Matrix II
- 1. Median of Two Sorted Arrays
- 1. Kth Smallest Element in a Sorted Matrix
- 1. Kth Smallest Number in Multiplication Table
- 1. Search in a Sorted Array of Unknown Size
- 1. Longest Increasing Subsequence
- 1. Find Peak Element
- 1. Find the Duplicate Number
- 1. Capacity To Ship Packages Within D Days
- 1. Find a Peak Element II
- 1. Find the Smallest Divisor Given a Threshold
- 1. Minimize the Maximum Difference of Pairs

2. Stack / Monotonic Stack (20 Problems)

- 1. Valid Parentheses
- 1. Evaluate Reverse Polish Notation
- 1. Min Stack
- 1. Daily Temperatures
- 1. Next Greater Element I
- 1. Next Greater Element II
- 1. Trapping Rain Water
- 1. Largest Rectangle in Histogram
- 1. Maximal Rectangle
- 1. Remove Duplicate Letters
- 1. Remove K Digits
- 1. Minimum Remove to Make Valid Parentheses
- 1. Longest Valid Parentheses
- 1. Reverse Substrings Between Each Pair of Parentheses
- 1. Online Stock Span
- 1. Asteroid Collision
- 1. Score of Parentheses

- 1. Number of Visible People in a Queue
- 1. Largest Rectangle in Histogram
- 1. Sum of Total Strength of Wizards

3. Linked List Techniques (20 Problems)

- 1. Add Two Numbers
- 1. Remove Nth Node From End of List
- 1. Merge Two Sorted Lists
- 1. Merge k Sorted Lists
- 1. Swap Nodes in Pairs
- 1. Reverse Nodes in k-Group
- 1. Rotate List
- 1. Remove Duplicates from Sorted List II
- 1. Partition List
- 1. Reverse Linked List II
- 1. Copy List with Random Pointer
- 1. Linked List Cycle
- 1. Linked List Cycle II
- 1. Reorder List
- 1. Insertion Sort List
- 1. Sort List
- 1. Intersection of Two Linked Lists
- 1. Reverse Linked List
- 1. Palindrome Linked List
- 1. Odd Even Linked List

Phase 3: Trees, Graphs, Union-Find (4–6 months)

1. Tree Traversal + Recursion (20 Problems)

- 1. Binary Tree Inorder Traversal
- 1. Binary Tree Preorder Traversal
- 1. Binary Tree Postorder Traversal
- 1. Binary Tree Level Order Traversal
- 1. Binary Tree Right Side View
- 1. Zigzag Level Order
- 1. Path Sum
- 1. Path Sum II
- 1. Binary Tree Maximum Path Sum
- 1. Balanced Binary Tree
- 1. Maximum Depth of Binary Tree
- 1. Minimum Depth of Binary Tree
- 1. Diameter of Binary Tree
- 1. Lowest Common Ancestor of a Binary Tree
- 1. Construct Binary Tree from Preorder and Inorder

- 1. Construct Binary Tree from Inorder and Postorder
- 1. Flatten Binary Tree to Linked List
- 1. Merge Two Binary Trees
- 1. Sum Root to Leaf Numbers
- 1. Range Sum of BST

2. Graphs + BFS/DFS (20 Problems)

- 1. Number of Islands
- 1. Rotting Oranges
- 1. Max Area of Island
- 1. Island Perimeter
- 1. Is Graph Bipartite?
- 1. Surrounded Regions
- 1. Number of Provinces
- 1. All Paths From Source to Target
- 1. Course Schedule
- 1. Course Schedule II
- 1. Clone Graph
- 1. Graph Valid Tree
- 1. Pacific Atlantic Water Flow
- 1. Reorder Routes to Make All Paths Lead to the City Zero
- 1. Number of Connected Components in an Undirected Graph
- 1. Redundant Connection
- 1. Keys and Rooms
- 1. Snakes and Ladders
- 1. Graph Valid Tree
- 1. All Paths from Source Lead to Destination

3. Union-Find (Disjoint Set) (20 Problems)

- 1. Number of Provinces
- 1. Redundant Connection
- 1. Accounts Merge
- 1. Number of Operations to Make Network Connected
- 1. Number of Connected Components
- 1. Smallest String With Swaps
- 1. Path With Minimum Effort
- 1. Min Cost to Connect All Points
- 1. Connecting Cities With Minimum Cost
- 1. Possible Bipartition
- 1. Evaluate Division
- 1. Graph Valid Tree
- 1. Couples Holding Hands
- 1. Regions Cut By Slashes
- 1. Making A Large Island
- 1. Minimize Malware Spread
- 1. Number of Provinces

- 1. Swim in Rising Water
- 1. Number of Closed Islands
- 1. Count Sub Islands