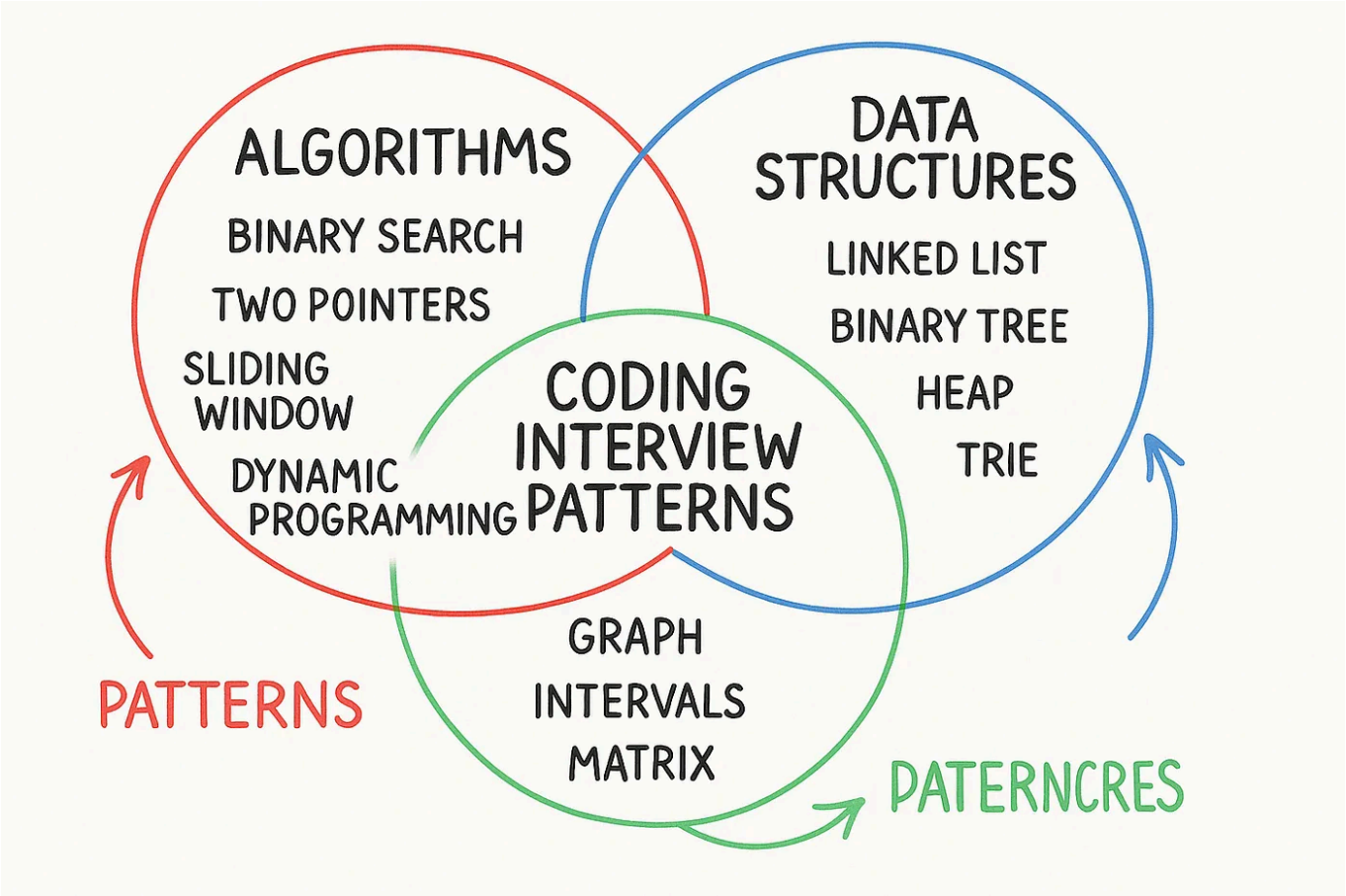
The Ultimate Guide to Coding Patterns for Cracking Top Tech Interviews (2025)

Anurag Goel Follow 5 min read · Jun 7, 2025

When it comes to cracking coding interviews at FAANG and top-tier startups, it's not about solving 1000+ questions — it's about mastering a **core set of patterns** that map to **150–200 high-yield problems.** This guide combines questions from Blind 75, Meta, Google, and top-sourced LeetCode patterns into one structured cheat sheet.



🚀 Why Patterns Over Problems?

Patterns help you instantly recognize what type of solution a problem needs, saving time and brainpower. Once you learn them, every LeetCode problem becomes a variation of something you already know.

🧠 16+ Must-Know Coding Patterns (with Key Problems + Insights)

Each pattern below includes:

- 💡 When to Use
- 🧪 Sample Problems (including Blind 75 & FAANG picks)
- 📌 Real-world use case (where applicable)

1. Two Pointers

💡 Best for traversing arrays/lists from both ends, especially when sorted.

Use Cases: Palindrome check, reverse operations, finding pair sums.

Key Problems:

- Two Sum II (LC 167)
- 3Sum (LC 15)
- Container With Most Water (LC 11)
- Trapping Rain Water (LC 42)
- Remove Duplicates from Sorted Array (LC 26)
- Move Zeroes (LC 283)
- Valid Palindrome (LC 125)
- Triangle Number (LC 611)
- Number of Subsequences Satisfy the Given Sum Condition (LC 1498)

2. Sliding Window

💡 Used when problems require optimization over subarrays or substrings.

Use Cases: Substring counts, max/min in a range, frequency maps.

Key Problems:

- Longest Substring Without Repeating Characters (LC 3)

- Minimum Window Substring (LC 76)
- Max Sliding Window (LC 239)
- Permutation in String (LC 567)
- Longest Repeating Character Replacement (LC 424)
- Maximum Sum of Subarrays of Size K (GFG)
- Max Consecutive Ones III (LC 1004)

3. Fast & Slow Pointers (Tortoise and Hare)

💡 Best for cycle detection and middle detection in linked structures.

Key Problems:

- Linked List Cycle (LC 141)
- Find Duplicate Number (LC 287)
- Palindrome Linked List (LC 234)
- Happy Number (LC 202)

4. Binary Search + Variants

💡 For sorted/rotated arrays, searching ranges, or optimizations.

Key Problems:

- Search in Rotated Sorted Array (LC 33)
- Find Peak Element (LC 162)

- Koko Eating Bananas (LC 875)
- Capacity to Ship Packages Within D Days (LC 1011)

5. Prefix Sum / Difference Arrays

💡 For range queries or cumulative calculations in O(1) time.

Key Problems:

- Subarray Sum Equals K (LC 560)
- Range Sum Query (LC 303)
- Contiguous Array (LC 525)

6. Dynamic Programming (DP)

💡 Break problems into overlapping subproblems.

Key Classic Problems:

- Climbing Stairs (LC 70)
- Longest Palindromic Substring (LC 5)
- Decode Ways (LC 91)
- Paint House (LC 256)
- Maximum Subarray (LC 53)
- Maximum Product Subarray (LC 152)
- Longest Repeating Substring (LC 1062)

Advanced:

- Longest Increasing Subsequence (LC 300)
- Word Break (LC 139)
- Coin Change (LC 322)
- Edit Distance (LC 72)
- Partition Equal Subset Sum (LC 416)

7. Backtracking

💡 Use when generating combinations, permutations, or exploring decision trees.

Key Problems:

- Subsets (LC 78)
- Permutations (LC 46)
- N-Queens (LC 51)
- Word Search (LC 79)
- Generate Parentheses (LC 22)

8. Greedy

💡 Take the best decision at each step — no going back.

Key Problems:

- Jump Game (LC 55)

- Merge Intervals (LC 56)
- Gas Station (LC 134)
- Candy (LC 135)
- Partition Labels (LC 763)
- Largest Number (LC 179)
- Can Place Flowers (LC 605)

9. Linked List Techniques

💡 Focused on pointer manipulation, reversing, merging, etc.

Key Problems:

- Add Two Numbers (LC 2)
- Merge Two Sorted Lists (LC 21)
- Reverse Linked List (LC 206)
- Copy List with Random Pointer (LC 138)

10. Tree Traversals & Binary Tree Patterns

💡 Recursive or iterative depth/level traversal, LCA, path problems.

Key Problems:

- Inorder Traversal (LC 94)
- Level Order Traversal (LC 102)

- Diameter of Binary Tree (LC 543)
- Lowest Common Ancestor (LC 235)

11. Graphs (BFS / DFS)

💡 For connected components, shortest paths, and graph cloning.

Key Problems:

- Number of Islands (LC 200)
- Number of Distinct Islands (LC 694)
- Clone Graph (LC 133)
- Course Schedule (LC 207)
- Pacific Atlantic Water Flow (LC 417)
- Word Ladder (LC 127)

12. Tries (Prefix Trees)

💡 Fast prefix-based searching. Useful for dictionaries and autocomplete.

Key Problems:

- Implement Trie (LC 208)
- Word Search II (LC 212)
- Replace Words (LC 648)

13. Heap / Priority Queue

💡 Top K problems, stream medians, and multi-way merges.

Key Problems:

- Top K Frequent Elements (LC 347)
- Merge K Sorted Lists (LC 23)
- Find Median from Data Stream (LC 295)

14. Monotonic Stack / Queue

💡 Used to maintain increasing/decreasing order for window problems.

Key Problems:

- Daily Temperatures (LC 739)
- Next Greater Element (LC 496)
- Largest Rectangle in Histogram (LC 84)
- Asteroid Collision (LC 735)
- Remove K Digits (LC 402)
- Finding Number of Visible Mountains (LC 2345)
- Shorted Array with Sum at Least K (LC 862)
- Maximum Width Ramp (LC 962)
- Remove Duplicate Letters (LC 316)
- 132 Pattern (LC 456)
- Sum of Subarray Minimums (LC 907)

- Smallest Subsequence of Distinct Characters (LC 1081)
- Minimum Number of Coins for Fruits (LC 2944)
- Maximum Length of Semi-Decreasing Subarrays (LC 2863)
- Longest Continuous Subarray With Absolute Diff Less Than or Equal to Limit (LC 1438)

15. Intervals

💡 Handling overlapping time or ranges, merging or scheduling.

Key Problems:

- Insert Interval (LC 57)
- Merge Intervals (LC 56)
- Meeting Rooms I/II (LC 252, LC 253)
- Non-overlapping Intervals (LC 435)

16. Bit Manipulation

💡 For toggling bits, single/missing numbers, and binary tricks.

Key Problems:

- Number of 1 Bits (LC 191)
- Reverse Bits (LC 190)
- Single Number (LC 136)
- Missing Number (LC 268)

✍️ Final Words

If you solve 150 problems and **deeply understand** why the pattern works, you don't just pass interviews — you become interview-proof.

Let me know if you'd like this:

- 📄 Printable PDF
- 🧠 Notion Board with Tracker
- 🧪 75-day Auto-prep Planner

Happy cracking, **Enjoyed the article?** Follow me on X.com for more interesting content, and check out my website at anuraggoel.in !😊🚀

Thank you for being a part of the community

Before you go:

- Be sure to **clap** and **follow** the writer 👏
- Follow us: **X** | **LinkedIn** | **YouTube** | **Newsletter** | **Podcast** | **Differ** | **Twitch**
- **Start your own free AI-powered blog on Differ** 🚀
- **Join our content creators community on Discord** 👨🏻‍💻
- For more content, visit **plainenglish.io** + **stackademic.com**

Interview | Interview Questions | Leetcode | System Design Interview | Coding

Published in Stackademic

42K followers · Last published 2 hours ago

Follow

Stackademic is a learning hub for programmers, devs, coders, and engineers. Our goal is to democratize free coding education for the world.

Written by Anurag Goel

433 followers · 43 following

Follow

I write and build fun stuff about finance, engineering & life. It's serious business, but I keep it light. Check out my website: anuraggoel.in .

No responses yet

Soltonbekov

What are your thoughts?

More from Anurag Goel and Stackademic

In Stackademic by Anurag Goel

Cracking Behavioral Interviews for Staff, Principal & Lead Engineer...

If you're aiming for a Staff, Principal, or Lead Engineer role, here's the hard truth:

Jun 7 21 3

In Stackademic by Dylan Cooper

My New Hobby: Watching Copilot Slowly Drive Microsoft Engineers...

A recent GitHub Copilot controversy has sparked widespread attention among...

May 27 851 43

In Stackademic by Dylan Cooper

Fired the AI Director. Disbanded the Python Core Team. Is Microso...

When AI-First Means People-Last: Microsoft's Layoffs Leave Python and AI Teams Gutted.

May 29 699 30

In Stackademic by Anurag Goel

My Uber Interview Experience—Cracking the SSE(L5A) Role 🚀

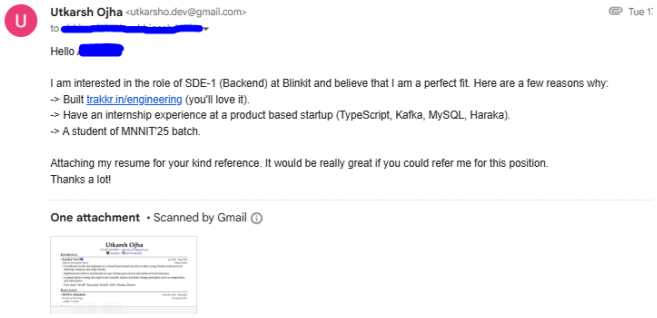
Hello, everyone. I'm excited to share that I cracked the Uber interview! 🎉 last year in...

Jan 19 306 7

See all from Anurag Goel

See all from Stackademic

Recommended from Medium

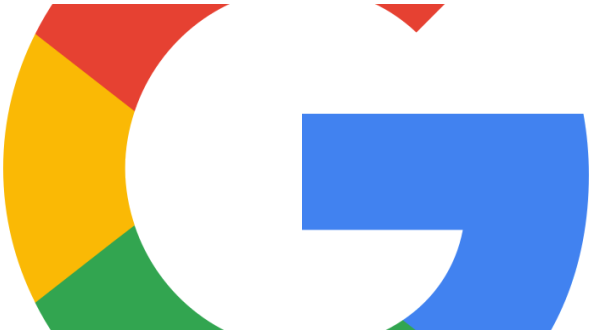


Utkarsh Ojha

Blinkit SDE-1 Interview Experience (fresher)

How I applied?

4d ago 53



In The 1-Hour DSA Prep R... by The Abstract Engi...

Max Distance with Limited Energy — Google Phone Screening...

You're given an initial amount of energy k and an array of wind speeds for n days. Each day...

Jun 16 63

In Stackademic by Kavya's Programming Path

"My ICICI Java Developer Interview Experience — DSA, Spring, and a...

Spoiler: If you're preparing for ICICI or any core backend Java role — DSA and Spring...

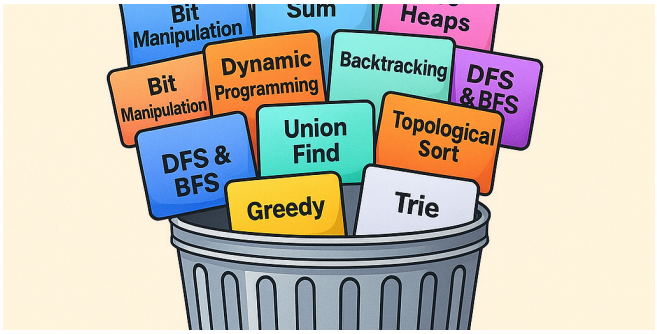
Shiwangi Kumari

My Amazon SDE1 Interview Experience

This blog has been long pending! I actually drafted it while going through the interview...

Jun 7 32

Feb 13 123 5



Himanshu Singour

These 16 DSA Patterns Did What 3000 LeetCode Problems Couldn't

When I started with DSA, I did what everyone else does. I opened LeetCode. Picked Easy....

Jun 19 634 10

Anthony D. Mays

5 Ways to Use AI for Coding Interview Prep (Without Cheating)

The Age of AI is here

May 27 10 2

See more recommendations