200+ SDE Interview Questions asked in

System Design, DSA, and Behavioral all in one place.

Arrays & Strings

- Two Sum variants (indices, count pairs, unique pairs)
- Best Time to Buy and Sell Stock (I/II/with cooldown/with fee)
- Product of Array Except Self (zeros & negatives)
- Longest Substring Without Repeating Characters
- Minimum Window Substring
- Longest Palindromic Substring
- Palindromic Substrings (count)
- Valid Anagram / Group Anagrams
- Rotate Array / String Rotation check
- Merge Intervals / Insert Interval

- Non-overlapping Intervals (erase overlaps)
- Meeting Rooms I/II
- Subarray Sum Equals K (positive & mixed)
- Maximum Subarray (Kadane) + circular variant
- Find Pivot Index / Equilibrium index
- Gas Station (circular tour)
- Container With Most Water
- Trapping Rain Water (two pointers & stack)
- 3Sum / 4Sum / 3Sum Closest
- Remove Duplicates from Sorted Array (I/II)
- Dutch National Flag (sort colors)
- Shortest Unsorted Continuous Subarray
- Word Break (I/II) – DP vs Trie
- Longest Common Prefix
- String to Integer (atoi) – edge cases

- Zigzag Conversion / String Compression
- Multiply Strings / Add Strings / Big integer
- Valid Parentheses / Minimum Add to Make Valid
- Backspace String Compare
- Find All Anagrams in a String
- Ransom Note / Construct String from Another
- Is Subsequence / Shortest Way to Form String
- Minimum Remove to Make Valid Parentheses
- Longest Repeating Character Replacement
- Sliding Window Maximum (deque)
- Find Minimum in Rotated Sorted Array (with duplicates)
- Search in Rotated Sorted Array (I/II)
- Find First and Last Position of Element (binary search)
- Longest Consecutive Sequence
- Set Matrix Zeroes (in-place)

- Spiral Matrix (I/II)
- Rotate Image (in-place 2D transform)
- Game of Life (in-place encoding)
- Word Search (board backtracking)
- Distinct Subsequences (count ways)

Hash / Stack / Queue

- LRU Cache (design) – hashmap + doubly linked list
- LFU Cache (design) – freq lists
- Min Stack / Max Stack (getMin in O(1))
- Evaluate Reverse Polish Notation
- Basic Calculator I/II/III (parentheses, precedence)
- Valid Parentheses with wildcards (* as) or empty)
- Next Greater Element (I/II) – monotonic stack
- Daily Temperatures – monotonic stack
- Stock Span / Online Stock Span
- Queue using Stacks / Stack using Queues

- Implement Circular Queue / Deque
- Design Hit Counter / Rate Limiter
- Design Browser History (two stacks)
- Decode String (k[encoded])
- Remove K Digits to form smallest number
- Asteroid Collision (stack)
- Candy Crush / String Explosion
- Largest Rectangle in Histogram
- Maximal Rectangle in Binary Matrix
- Celebrity Problem / Find the Town Judge

Linked Lists

- Reverse Linked List (I/II partial groups of k)
- Merge Two Sorted Lists / Merge K Sorted Lists (heap)
- Add Two Numbers (forward & reverse order)
- Remove Nth Node From End
- Reorder List / Odd Even Linked List
- Linked List Cycle (detect & entry point)
- Copy List with Random Pointer
- Partition List by value
- Sort List (merge sort on list)
- Rotate List / Swap Nodes in Pairs

Trees & BST

- Binary Tree Inorder/Preorder/Postorder Traversal (iterative & recursive)
- Level Order Traversal / Zigzag Level Order
- Diameter of Binary Tree
- Balanced Binary Tree / Height Balanced Check
- Maximum Depth / Minimum Depth of Binary Tree
- Path Sum I/II/III (count paths)
- Lowest Common Ancestor (Binary Tree)
- Lowest Common Ancestor (BST)
- Serialize and Deserialize Binary Tree (design)
- Construct Tree from Preorder & Inorder / Postorder
- Binary Tree Right Side View / Left View
- Flatten Binary Tree to Linked List
- Validate Binary Search Tree (bounds)
- Kth Smallest / Largest in BST
- Convert Sorted Array/List to BST

- Inorder Successor/Predecessor in BST
- Range Sum of BST / Trim BST
- Recover Binary Search Tree (swap fix)
- Count Complete Tree Nodes (binary search on tree)
- Sum of Distances in Tree (tree DP re-rooting)

Graphs

- Number of Islands (DFS/BFS/Union-Find)
- Clone Graph (BFS/DFS)
- Course Schedule I/II (topological sort)
- Alien Dictionary (topo + lexicographic constraints)
- Word Ladder I/II (BFS + heuristics)
- Pacific Atlantic Water Flow / Surrounded Regions
- Walls and Gates / Rotting Oranges
- Shortest Path in Binary Matrix / Maze solver
- Network Delay Time (Dijkstra)
- Cheapest Flights Within K Stops (Bellman-Ford/DP)

- Reconstruct Itinerary (Hierholzer / Eulerian path)
- Redundant Connection (Union-Find)
- Graph Valid Tree / Count Components
- Kruskal / Prim Minimum Spanning Tree (design question)
- Evaluate Division (graph with weights)
- Accounts Merge (Union-Find)
- Critical Connections (Tarjan bridges)
- Course Scheduling with prerequisites window (advanced topo)
- Minimum Height Trees
- Bipartite Graph Check / Possible Bipartition

Searching & Sorting

- Quickselect (Kth Largest Element)
- Top K Frequent Elements (heap/bucket)
- Sort Colors (DNF) / 3-way partition
- Merge K Sorted Arrays (heap) / external sort idea
- Search a 2D Matrix (I/II)
- Find Peak Element / Mountain Array peak
- Median of Two Sorted Arrays (binary search on partitions)
- Sort Characters By Frequency / Relative Sort Array
- H-Index / H-Index II
- Find Duplicate Number (Floyd vs binary search on answer)

Heaps & Greedy

- Meeting Rooms II (min-heap)
- Task Scheduler (cooldown) – greedy + heap
- Reorganize String / Rearrange String k Distance Apart
- IPO / Maximize Capital (two heaps)
- Hand of Straights / Divide Array in Sets of K
- Boats to Save People / Assign Cookies
- Gas Station (greedy proof)
- Jump Game I/II (greedy farthest reach)
- Partition Labels
- Minimum Number of Arrows to Burst Balloons
- Non-overlapping Intervals (erase minimum)
- Candy (two-pass greedy)
- Lemonade Change
- Split Array Largest Sum (binary search + greedy check)
- Minimum Cost to Connect Sticks (heap)

Dynamic Programming

- Climbing Stairs / Min Cost Climbing Stairs
- House Robber I/II/III
- Coin Change (min coins, count combinations)
- Longest Increasing Subsequence (O(n log n))
- Edit Distance (Levenshtein)
- Decode Ways I/II
- Unique Paths I/II (with obstacles)
- Partition Equal Subset Sum / Target Sum
- Burst Balloons (interval DP)
- Palindromic Substrings / Longest Palindromic Subsequence
- Regular Expression Matching / Wildcard Matching
- Best Time to Buy/Sell Stock with K transactions
- Word Break DP (count ways vs boolean)
- Interleaving String
- Scramble String

- Matrix Chain Multiplication / Minimum Score Triangulation
- Paint House I/II (with k colors)
- Minimum Falling Path Sum / Triangle
- Cherry Pickup / Grid DP with states
- Knapsack 0/1 & Unbounded variants (pattern)

Math / Bit / Geometry

- Reverse Bits / Number of 1 Bits
- Single Number I/II/III (bitwise)
- Missing Number / Find All Missing / Duplicates
- Power of Two/Three/Four / Fast Pow (binary exponentiation)
- Count Primes / Sieve of Eratosthenes
- Greatest Common Divisor / Euclid / LCM problems
- Happy Number / Ugly Number / Super Ugly Number
- Rectangle Overlap / Line Sweep basics
- Random Pick with Weight / Reservoir Sampling
- Kth Smallest in Multiplication Table (binary search on answer)

Meritshot
EDUCATION

System Design (often asked at 2–3+ years)

- Design a URL Shortener
- Design a News Feed / Timeline
- Design a Rate Limiter (per user/IP) – token bucket
- Design a Key-Value Store (caching + persistence)
- Design an E-commerce Cart & Checkout
- Design a Messaging/Chat System (WhatsApp-lite)
- Design a File Storage Service (Dropbox/Google Drive)
- Design a Web Crawler
- Design a Ride-Hailing/Dispatch System
- Design a Logging & Metrics System
- Design a Notification Service (email/SMS/push)
- Design a Payment System (idempotency, retries)
- Design a Search Autocomplete
- Design a Video Streaming Platform (Netflix-lite)
- Design a Pastebin

- Design a Social Graph (follow/unfollow, recommendations)
- Design a Leaderboard (time-decay, top-N)
- Design a Distributed Job Queue
- Design a Real-time Analytics Pipeline
- Design a Feature Flag Service

Behavioral (Amazon/Google-style LPs)

- Tell me about a time you owned an ambiguous problem end-to-end. (Ownership)
- Describe a situation you disagreed with a senior engineer and what happened. (Disagree & Commit)
- Tell me about a time you improved performance or reduced cost. (Deliver Results)
- Give an example when you prioritized customer needs over engineering preferences. (Customer Obsession)

- Tell me about a production incident you handled—root cause and prevention. (Dive Deep)
- Describe a time you moved fast with incomplete information. (Bias for Action)
- Tell me about mentoring a teammate and the impact. (Hire/Develop the Best)
- Describe a time you simplified a complex system or process. (Invent & Simplify)
- Tell me about a failure—what did you learn and change? (Learn & Be Curious)
- How do you handle conflicting deadlines across teams? (Earn Trust)

Your one-step destination for your Career Upskilling

Lets make a community of 20k+ learnerrs