Company-wise DSA Qs List By Himanshu Gupta

Company Name	DSA Problem Link	Tricks for Solving Patterns
Google	Google DSA Problems	Focus on backtracking and dynamic programming.
Microsoft	Microsoft DSA Problems	Practice binary search, greedy algorithms.
Amazon	Amazon DSA Problems	Work on sliding window, heap, and DFS/BFS.
Facebook	Facebook DSA Problems	Master recursion, graphs, and bit manipulation.
Apple	Apple DSA Problems	Prioritize dynamic programming and divide and conquer.
Adobe	Adobe DSA Problems	Practice sorting and searching techniques.
Goldman Sachs	Goldman Sachs DSA Problems	Focus on arrays, hashing, and string manipulation.
Uber	Uber DSA Problems	Work on graphs, BFS/DFS, and Dijkstra's algorithm.
LinkedIn	LinkedIn DSA Problems	Focus on two-pointer and sliding window techniques.
Netflix	Netflix DSA Problems	Prioritize dynamic programming and DP on trees.
Twitter	Twitter DSA Problems	Work on greedy algorithms and interval problems.
Dropbox	Dropbox DSA Problems	Master recursion and binary search on sorted arrays.
Airbnb	Airbnb DSA Problems	Practice graph traversal and backtracking.
Salesforce	Salesforce DSA Problems	Work on greedy algorithms and dynamic programming.
Oracle	Oracle DSA Problems	Focus on matrix traversal and dynamic programming.

Company Name	DSA Problem Link	Tricks for Solving Patterns
PayPal	PayPal DSA Problems	Prioritize binary search and sorting problems.
Walmart	Walmart DSA Problems	Focus on hashing, prefix sums, and arrays.
Expedia	Expedia DSA Problems	Practice stack-based problems and recursion.
Snap	Snap DSA Problems	Focus on graphs and dynamic programming.
Yahoo	Yahoo DSA Problems	Work on linked lists and recursion.
DoorDash	DoorDash DSA Problems	Master binary trees and backtracking.
Stripe	Stripe DSA Problems	Practice greedy algorithms and string manipulation.
Lyft	Lyft DSA Problems	Focus on two-pointer and sliding window techniques.
Intuit	Intuit DSA Problems	Work on backtracking and dynamic programming.
IBM	IBM DSA Problems	Master dynamic programming and recursion problems.
Atlassian	Atlassian DSA Problems	Focus on graph traversal and dynamic programming.
Reddit	Reddit DSA Problems	Work on hashing and bit manipulation.
Pinterest	Pinterest DSA Problems	Master recursion and divide and conquer techniques.
Spotify	Spotify DSA Problems	Focus on sorting, searching, and heaps.
Bloomberg	Bloomberg DSA Problems	Work on arrays, dynamic programming, and graphs.
Cisco	Cisco DSA Problems	Focus on linked lists and dynamic programming.

Company Name	DSA Problem Link	Tricks for Solving Patterns
ByteDance	ByteDance DSA Problems	Master sorting algorithms and binary search.
Tesla	Tesla DSA Problems	Work on graph traversal and dynamic programming.
TikTok	TikTok DSA Problems	Prioritize dynamic programming and recursion.
Nvidia	Nvidia DSA Problems	Practice bit manipulation and backtracking.

50 tricks to identify DSA Patterns Link and other guides and cheat sheets required for DSA:

https://drive.google.com/drive/folders/1GbYapJnWJZtFlf2mSC1HqvgYLLL_BATJ?usp=drive_link

🛤️ 3-Month Logic-Building & Problem-Solving Roadmap by *HIMANSHU GUPTA (codeprime.io)*

Week	Focus Area	Daily Breakdown	Checkpoints
Week 1	**Foundation: Basics of Programming**	**Day 1-2:** Learn variables, data types, loops (for, while). **Day 3-4:** Conditionals, functions. **Day 5-6:** Arrays & Strings basics. **Day 7:** Solve 10 simple pattern-building questions (stars, triangles).	✅ Understand basic syntax and concepts. ✅ Solve **10 pattern-building problems**.
Week 2	**Introduction to Problem Solving**	**Day 1-2:** Learn dry-run and pseudocode. **Day 3-5:** Solve **10 simple DSA problems** (easy level, e.g., reverse array, Fibonacci). **Day 6-7:** Build **Mini Project 1: Calculator App**.	✅ Solve 10 DSA problems. ✅ Complete **Mini Project 1: Calculator App**.

Week	Focus Area	Daily Breakdown	Checkpoints
Week 3	**Level Up: Arrays & Logic Thinking**	**Day 1-2:** Advanced array techniques (sliding window, prefix sum). **Day 3-6:** Solve **15 array-based problems** (medium level). **Day 7:** Build **Mini Project 2: Tic-Tac-Toe Game**.	✅ Solve 25 total DSA problems (15 new). ✅ Complete **Mini Project 2: Tic-Tac-Toe Game**.
Week 4	**Strings & Problem-Solving Practice**	**Day 1-2:** String manipulation (reversal, palindromes, substrings). **Day 3-5:** Solve **15 string-based problems** (easy-medium). **Day 6-7:** Build **Mini Project 3: Text Manipulation Tool (e.g., Uppercase, Reverse)**.	✅ Solve 40 total DSA problems (15 new). ✅ Complete **Mini Project 3: Text Tool**.
Week 5	**Recursion & Logic Expansion**	**Day 1-2:** Learn recursion basics (factorial, Fibonacci). **Day 3-5:** Solve **10 recursion-based problems** (medium level). **Day 6-7:** Build **Mini Project 4: Recursive Maze Solver**.	✅ Solve 50 total DSA problems (10 new). ✅ Complete **Mini Project 4: Recursive Maze Solver**.
Week 6	**Data Structures: Stacks & Queues**	**Day 1-2:** Learn stacks and queues basics. **Day 3-6:** Solve **15 problems** (balanced parentheses, queue reversal). **Day 7:** Work on **Mini Project 5: Browser History Tracker (using stack)**.	✅ Solve 65 total DSA problems (15 new). ✅ Complete **Mini Project 5: Browser History Tracker**.
Week 7	**Dynamic Programming Introduction**	**Day 1-3:** Learn DP basics (knapsack, Fibonacci with memoization). **Day 4-6:** Solve **10 DP problems** (easy-medium). **Day 7:** Revise all past concepts/projects.	✅ Solve 75 total DSA problems (10 new). ✅ Master basic DP problems.

Week	Focus Area	Daily Breakdown	Checkpoints
Week 8	**Advanced Problem Solving (Graph)**	**Day 1-2:** Learn graph representation (adjacency list/matrix). **Day 3-5:** Solve **10 graph-based problems** (BFS/DFS). **Day 6-7:** Build a **Mini Project: Path Finder Visualizer**.	✅ Solve 85 total DSA problems (10 new). ✅ Complete Graph-Based Mini Project.
Week 9	**Advanced DSA (Sorting/Greedy)**	**Day 1-2:** Learn sorting algorithms (merge sort, quicksort). **Day 3-5:** Solve **15 problems** on sorting/greedy (e.g., activity selection, job scheduling).	✅ Solve 100 DSA problems (15 new).
Week 10-11	**Integration: Complex Projects**	**Day 1-7:** Brainstorm and build **Project 1: Expense Tracker with Charts**. **Next 7 days:** Build **Project 2: Multiplayer Rock-Paper-Scissors Game** (WebSockets, if possible).	✅ Integrate past knowledge into real-world projects. ✅ Master real-world application-building.
Week 12	**Final Touches and Mock Practice**	**Day 1-3:** Revise all concepts learned. **Day 4-6:** Solve 10 problems from past mistakes or blindspots. **Day 7:** Build a final project of your choice integrating at least one complex DSA concept.	✅ Complete final project. ✅ Feel confident in solving beginner-intermediate DSA problems.

💡 **How to Think of Logic**

1. **Break Down the Problem**: Read the question twice. Identify inputs, outputs, and constraints.
2. **Start with Examples**: Create test cases manually and simulate the solution step-by-step.
3. **Write Pseudocode**: Draft a high-level plan before coding.
4. **Ask "Why?" at Every Step**: Understand each operation; don't memorize solutions.
5. **Visualize**: Use diagrams or dry-run tables to debug.
6. **Optimize Gradually**: Start with brute force; iterate to optimize for efficiency.

📚 **100 DSA Questions for Logic Building**

Category	Question Name	Platform	Link
Basics & Warm-Up	Print a pattern of stars (triangle, pyramid)	GeeksforGeeks	Link
	Reverse a number	GeeksforGeeks	Link
	Check if a number is palindrome	GeeksforGeeks	Link
	Count digits in a number	GeeksforGeeks	Link
	Find factorial of a number	HackerRank	Link
Arrays	Reverse an array	LeetCode	Link
	Find the maximum and minimum of an array	GeeksforGeeks	Link
	Rotate an array by K steps	LeetCode	Link
	Move all zeroes to the end	LeetCode	Link
	Kadane's Algorithm (Maximum Subarray Sum)	LeetCode	Link
Strings	Reverse a string	LeetCode	Link
	Check if two strings are anagrams	LeetCode	Link
	Longest Common Prefix	LeetCode	Link
	Check if a string is a palindrome	LeetCode	Link
	Count and say	LeetCode	Link
Recursion	Fibonacci series using recursion	GeeksforGeeks	Link
	Tower of Hanoi	GeeksforGeeks	Link

Category	Question Name	Platform	Link
	Factorial using recursion	LeetCode	Link
	Reverse a linked list using recursion	LeetCode	Link
	Permutations of a string	GeeksforGeeks	Link
Sorting	Bubble sort	GeeksforGeeks	Link
	Selection sort	GeeksforGeeks	Link
	Merge sort	GeeksforGeeks	Link
	Quick sort	GeeksforGeeks	Link
	Insertion sort	GeeksforGeeks	Link
Searching	Binary search	LeetCode	Link
	Linear search	GeeksforGeeks	Link
	Search in a rotated sorted array	LeetCode	Link
	First and last position in a sorted array	LeetCode	Link
	Square root of a number (using binary search)	LeetCode	Link
Linked List	Reverse a linked list	LeetCode	Link
	Detect a cycle in a linked list	LeetCode	Link
	Merge two sorted linked lists	LeetCode	Link
	Remove Nth node from the end	LeetCode	Link

Category	Question Name	Platform	Link
	Find the middle of a linked list	LeetCode	Link
Stacks & Queues	Implement a stack using arrays	LeetCode	Link
	Evaluate postfix expression	GeeksforGeeks	Link
	Balanced parentheses	LeetCode	Link
	Next greater element	LeetCode	Link
	Implement a queue using stacks	LeetCode	Link
Dynamic Programming	0/1 Knapsack problem	GeeksforGeeks	Link
	Fibonacci using dynamic programming	LeetCode	Link
	Longest common subsequence	LeetCode	Link
	Longest increasing subsequence	LeetCode	Link
	Minimum steps to reach the end	LeetCode	Link
Graphs	BFS traversal	GeeksforGeeks	Link
	DFS traversal	GeeksforGeeks	Link
	Detect cycle in an undirected graph	GeeksforGeeks	Link
	Shortest path in a graph (Dijkstra's)	GeeksforGeeks	Link
	Topological sort	GeeksforGeeks	Link

Follow me on Insta @codeprime.io

Subscribe to my YouTube channel @itshimanshu2001