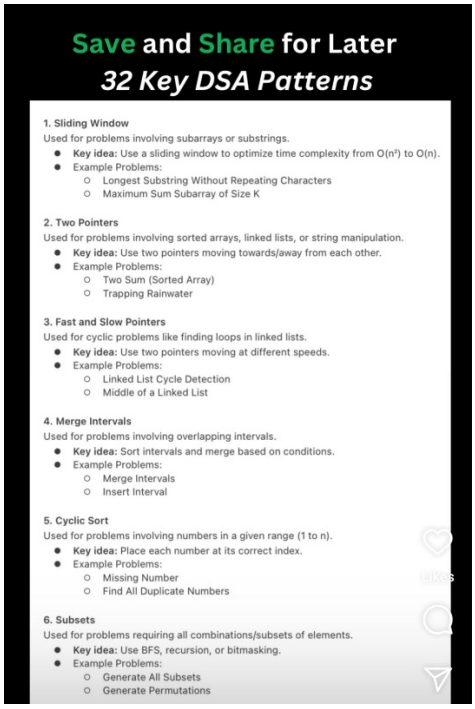
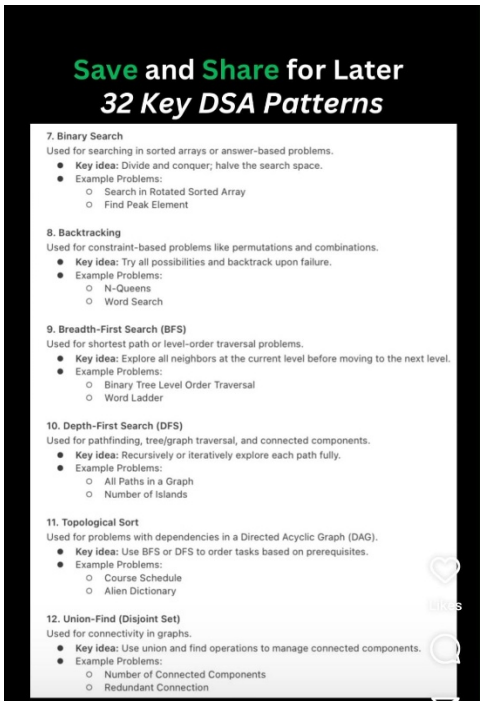
Tejaswini Atluri
10/17/2025

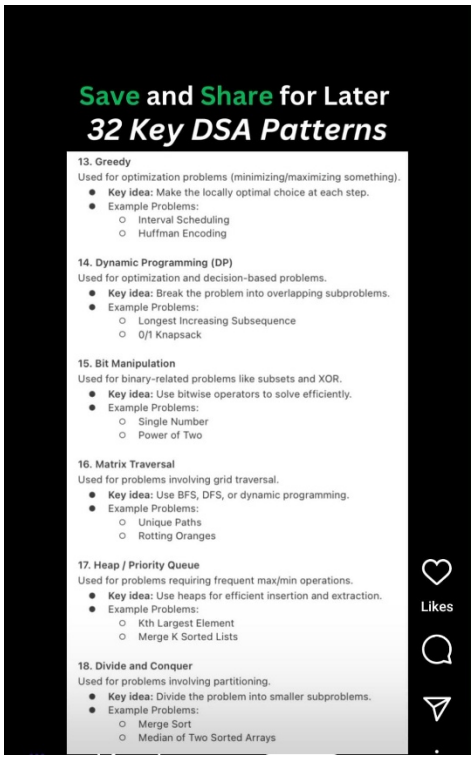
Useful resources which helped me with my interview preparation

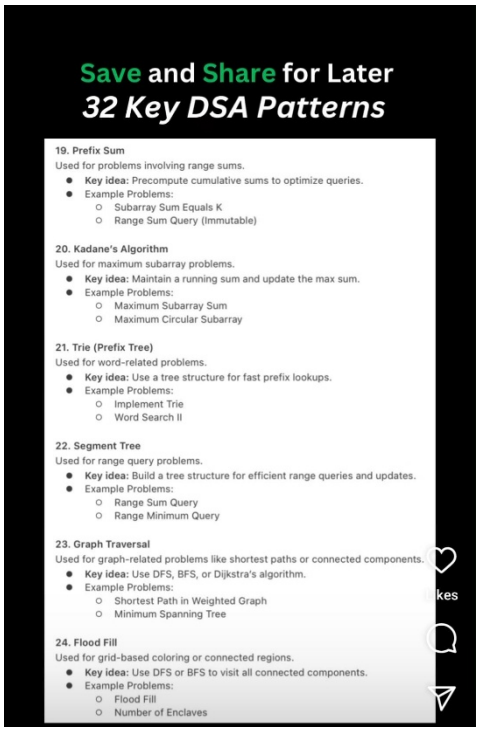
Every individual has their own unique approach to learning and preparation. The methods and resources mentioned below reflect the strategy that personally worked well for me. There is no fixed approach to follow, these are simply recommendations and insights that can be adapted and customized based on your personal learning style and preferences.

1. **Leetcode:** Most coding assessments / online assessments are heavily focused on data structures and algorithmic problems, many of which are inspired by Leet Code questions. The images below provide a quick reference guide covering various topics to help strengthen conceptual understanding.
   While understanding the underlying concepts is crucial, hands-on practice through coding is equally important: https://leetcode.com/.
   For additional clarity, numerous You Tube videos offer in-depth explanations of these concepts.




Save and Share for Later
32 Key DSA Patterns
7. Binary Search
Used for searching in sorted arrays or answer-based problems.
Key idea: Divide and conquer; halve the search space.
Example Problems:
Search in Rotated Sorted Array
Find Peak Element
8. Backtracking
Used for constraint-based problems like permutations and combinations.
Key idea: Try all possibilities and backtrack upon failure.
Example Problems:
N-Queens
Word Search
9. Breadth-First Search (BFS)
Used for shortest path or level-order traversal problems.
Key idea: Explore all neighbors at the current level before moving to the next level.
Example Problems:
Binary Tree Level Order Traversal
Word Ladder
10. Depth-First Search (DFS)
Used for pathfinding, tree/graph traversal, and connected components.
Key idea: Recursively or iteratively explore each path fully.
Example Problems:
All Paths in a Graph
Number of Islands
11. Topological Sort
Used for problems with dependencies in a Directed Acyclic Graph (DAG).
Key idea: Use BFS or DFS to order tasks based on prerequisites.
Example Problems:
Course Schedule
Alien Dictionary
12. Union-Find (Disjoint Set)
Used for connectivity in graphs.
Key idea: Use union and find operations to manage connected components.
Example Problems:
Number of Connected Components
Redundant Connection


Save and Share for Later
32 Key DSA Patterns
13. Greedy
Used for optimization problems (minimizing/maximizing something).
Key idea: Make the locally optimal choice at each step.
Example Problems:
Interval Scheduling
Huffman Encoding
14. Dynamic Programming (DP)
Used for optimization and decision-based problems.
Key idea: Break the problem into overlapping subproblems.
Example Problems:
Longest Increasing Subsequence
0/1 Knapsack
15. Bit Manipulation
Used for binary-related problems like subsets and XOR.
Key idea: Use bitwise operators to solve efficiently.
Example Problems:
Single Number
Power of Two
16. Matrix Traversal
Used for problems involving grid traversal.
Key idea: Use BFS, DFS, or dynamic programming.
Example Problems:
Unique Paths
Rotting Oranges
17. Heap / Priority Queue
Used for problems requiring frequent max/min operations.
Key idea: Use heaps for efficient insertion and extraction.
Example Problems:
Kth Largest Element
Merge K Sorted Lists
18. Divide and Conquer
Used for problems involving partitioning.
Key idea: Divide the problem into smaller subproblems.
Example Problems:
Merge Sort
Median of Two Sorted Arrays
Likes



Below are other leetcode links which are also helpful.
https://leetcode.com/studyplan/top-interview-150/
https://leetcode.com/explore/interview/card/top-interview-questions-easy/

2. https://www.youtube.com/@GregHogg - Frequently Asked Questions in FAANG companies.
   Fun fact: I follow him on insta. So, every day when I open it, I see his new uploaded video and learn something. So, it's a productive alternative to the usual reel scrolling. @greghogg5 – Instagram profile.

3. https://www.teamblind.com/post/new-year-gift---curated-list-of-top-75-leetcode-questions-to-save-your-time-OaM1orEU
   This is a very useful list of questions. Focuses more on important concepts. So, you could use it 2 days before your interview to quickly go through the important concepts.

4. LinkedIn is a very important resource. It is very useful for applying for jobs as well as networking. Have alerts set for roles which you are interested in so that you get mails daily and try to apply every day.

5. Try to build some projects and showcase them on your GitHub. You can also upload your LeetCode solutions, as this demonstrates consistent practice and ongoing effort to improve your coding skills.

6. Frequently search if there are any new roles in career pages and apply.

7. ChatGPT is very helpful while learning! It can solve all your doubts and give you detailed explanations. It is the best learning tool if used wisely.

8. Check out latest questions/ interview experiences for each company from online resources such as Glassdoor, Discord etc.

My Interview Process

ROUND 1: Online Assessment

The first round was online, and it consisted of **three sections**.

<u>Section 1: Coding Challenge</u>

This round included **two coding questions** with a total time limit of **75 minutes**. I could use any programming language of my choice.

- The first question was of *easy* difficulty and could be solved using basic logical reasoning.
- The second question was of *medium* difficulty and focused on dynamic programming concept.

I would strongly recommend practicing problems on LeetCode to get familiar with the type and level of questions asked. For entry-level or new graduate positions, questions generally range between easy and medium levels.

<u>Section 2: Email and Text Message Simulation</u>

This round simulated a professional communication environment through emails and text messages. I was asked to read and interpret various scenarios, then select the most appropriate course of action from multiple-choice options. There was no single "correct" answer, the goal was to assess your decision-making style and workplace behavior.

For example, a sample scenario could be:

After a discussion with your manager, you still don't fully understand the project requirements. What would you do?
A) Ask additional questions and clarify before starting the task.
B) Analyze the information further for some time, and if you still feel uncertain, approach your manager for clarification.
C) Proceed with what you think is correct.

In such cases, option C would generally be less favorable, while A and B reflect proactive and thoughtful approaches. This round is straightforward and provides insight into your work culture fit and problem-solving approach.

Section 3: Behavioral Assessment

The final round focused on behavioral and personality-based questions. It involved choosing between two statements, depending on which one best aligned with your personal preferences or work style.

ROUND 2: Final online interview round

The final interview stage consisted of three separate interviews, each lasting one hour. The interviews were a mix of technical and behavioral rounds, designed to evaluate both problem-solving and leadership skills.

Interview 1: Technical round

The first interview focused entirely on coding and problem-solving. It began with a brief introduction. I was asked a coding problem, and I was provided with a shared online editor, where both the interviewer and I could view and edit the code in real-time.

I began by explaining my pseudocode and walking through my thought process before implementing the actual solution. After coding, I demonstrated the approach using an example and discussed its time complexity. I followed this method to clearly convey my problem-solving approach to the interviewer and to identify any potential implementation issues early in the process.

The interviewer also had follow-up questions based on my code such as, what would happen if we modified this particular line of code, how would you handle a specific edge case, etc.

Since time permitted, I was also asked to discuss an alternative approach.

Tip: Even if you are not sure of an approach, always attempt to reason through it. Demonstrating curiosity and the ability to adapt to new concepts leaves a strong impression.

Interview 2: Behavioral Questions

The second interview was focused on behavioral questions based on Amazon's Leadership Principles. The interviewer assessed how well my previous experiences reflected these values.
For Amazon's leadership principles refer to this: https://www.amazon.jobs/content/en/our-workplace/leadership-principles

I structured my responses using the STAR method: Situation, Task, Action, and Result.
To know more about STAR method, refer to this: https://amazon.jobs/content/en/how-we-hire/interview-loop

The interviewer asked around 10 behavioral questions, each targeting a different leadership principle. It's beneficial to prepare multiple examples from varied experiences (previous work, projects, internships, team collaborations, etc.) so that each response highlights a different quality.

Interview 3: Technical + Behavioral Questions.

The final interview combined both technical and behavioral elements. I was asked two behavioral questions and one coding problem. After coding, I was asked to explain the time complexity, discuss edge cases and potential improvements to my implementation.

At the end of each interview, I was given about five minutes to ask questions to the interviewer. This was a valuable opportunity to learn more about the company's culture, team structure and ongoing projects. Asking thoughtful questions at this stage demonstrates genuine interest and curiosity about the organization.