MASTER DATA STRUCTURES & ALGORITHMS with C++ (C++17)

The Complete 7-Month Roadmap to Crack FAANG Interviews

From Zero to Hero: Solve LeetCode Top 150 Interview Questions

Starts: February 10, 2026

56 Live Sessions | Mentor Support | Lifetime Access

Limited to 30 Students Only

- **150+ LeetCode Problems**
- **Step-by-Step Visualizations**
- **Mock Interviews**
- **Resume Building & Reviews**
- **Private Discord Community**
- **Certificate of Completion**

ONLY $150 USD

Instructor: Mahmoud Salem

Experienced Software Engineer & Competitive Programmer

Google DSC Lead | 500+ LeetCode Problems Solved

+20 101 162 0431 ma7moudalysalem@gmail.com

1 WELCOME MESSAGE

Welcome to the **Data Structures & Algorithms with C++** course.
This course is designed to help you move from knowing how to write code to understanding how to *think algorithmically*. You will learn how to analyze problems, break them down, and build correct and efficient solutions.
This is not a crash course. It is a **structured, long-term journey** that rewards consistency and effort.

2 WHY THIS COURSE?

Accelerate Your Career

This is not just another coding tutorial. It is a **comprehensive career transformation program**. Whether you are a student aiming for internships or a professional targeting top-tier tech companies, this course bridges the gap between theory and high-performance coding interviews. You will learn to *think* like a software engineer, not just memorize solutions.

> *"The goal is not to solve every problem. The goal is to recognize patterns so well that new problems feel familiar."*
>
> — **Mahmoud Salem**

3 WHAT YOU WILL GAIN

By the end of this course, you will be able to:

- Solve **150+ carefully selected LeetCode problems**
- Identify common algorithmic patterns instantly
- Master **recursion, trees, graphs, and dynamic programming**
- Analyze **time and space complexity** confidently
- Approach technical interviews with **clarity and confidence**
- Write **clean, optimized C++ code** using modern STL
- Communicate your thought process effectively during interviews

</> What You Will Master

- **Core DSA**: Arrays, Linked Lists, Trees, Graphs, DP.
- **C++ STL**: Power up your solutions with standard libraries.
- **Pattern Recognition**: Don't memorize; learn to *see* the solution.
- **Time Complexity**: Analyze and optimize your code on the fly.
- **Interview Skills**: Communicate your thought process effectively.

Who Is This For?

- **Students**: CS undergrads preparing for placements.
- **Developers**: Looking to switch to product-based companies.
- **Career Changers**: Transitioning into software engineering.
- **Beginners**: Anyone willing to put in the hard work.

4 HOW THE COURSE IS STRUCTURED

Each topic is delivered using a proven structure:

- **Concept Sessions** — Intuition, mental models, visuals, and core ideas
- **Problem-Solving Sessions** — Live solving of interview-level problems
- **Hybrid Sessions** — Explanation combined with guided practice
- **Review Sessions** — Consolidation and Q&A at the end of each month

*Special emphasis is placed on **recursion**, as it forms the foundation for trees, backtracking, and dynamic programming.*

5 STUDENT COMMITMENT

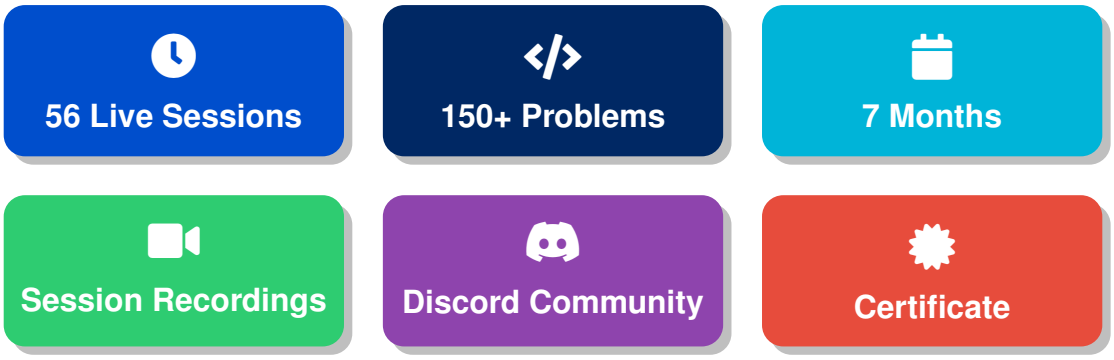
Important

To benefit fully from this course, students are expected to:

- Attend sessions regularly (**2 sessions per week**)
- Review session material after each class
- Attempt problems **independently** before looking at solutions
- Accept struggle as part of the learning process
- Dedicate **1-2 hours daily** for practice

Progress comes from consistency, not speed.

6 COURSE HIGHLIGHTS



Pro Tip

Session recordings are available for 48 hours after each live session. Make sure to watch them if you miss a class!

7 COMPREHENSIVE CURRICULUM ROADMAP

Important Note

Each month includes **8 sessions** (2 per week). Sessions alternate between **Concept** lectures and **Problem-Solving** workshops. Total: **56 sessions** over 7 months.

Month	Focus Area	Key Topics & LeetCode Examples
Month 1	**Foundations, Arrays & Strings**	• Big-O Complexity, Memory Management, C++ STL Review • Two Pointers, Sliding Window, Prefix Sum • **Problems**: Two Sum, Best Time to Buy/Sell Stock, Maximum Subarray, Container With Most Water, 3Sum
Month 2	**Linked Lists, Stacks & Queues**	• Fast & Slow Pointers, Monotonic Stack, Deque • Cycle Detection, Merge Operations • **Problems**: Reverse Linked List, Valid Parentheses, LRU Cache, Min Stack, Sliding Window Maximum
Month 3	**Recursion & Backtracking**	• Call Stack Visualization, Base Cases, Pruning Techniques • Decision Trees, State Space Trees • **Problems**: Permutations, N-Queens, Subsets, Word Search, Combination Sum, Letter Combinations
Month 4	**Trees & Binary Search Trees**	• DFS (Preorder, Inorder, Postorder), BFS, Level-order • Tree Properties, BST Operations, Tree Construction • **Problems**: Max Depth, LCA, Validate BST, Serialize Tree, Binary Tree Paths, Invert Binary Tree
Month 5	**Heaps, Hashing & Sorting**	• Priority Queues, Custom Comparators, Heap Operations • Hash Maps, Hash Sets, Collision Handling • **Problems**: Top K Frequent, Merge Intervals, Group Anagrams, Kth Largest Element, Meeting Rooms

Month	Focus Area	Key Topics & LeetCode Examples
Month 6	**Graphs & Binary Search**	• Graph Representations, BFS/DFS on Graphs • Shortest Path (Dijkstra, BFS), Topological Sort, Union-Find • Binary Search Variants, Search Space Reduction • **Problems**: Number of Islands, Course Schedule, Clone Graph, Search in Rotated Array, Find Peak Element
Month 7	**Dynamic Programming & Final Review**	• Memoization vs Tabulation, 1D/2D DP, State Transitions • Classic DP Patterns: Knapsack, LIS, LCS, Grid DP • **Problems**: Climbing Stairs, Coin Change, LCS, House Robber, Edit Distance, Unique Paths

8 DETAILED SESSION BREAKDOWN

Month 1: Foundations, Arrays & Strings (February 2026)

#	Type	Topic	LeetCode
1	Concept	Course Overview + C++ Review	–
2	Concept	Time & Space Complexity, STL Containers	–
3	Concept	Array Problem Patterns	–
4	Solve	Two Pointers Technique	#1, #26, #27, #88
5	Concept	Sliding Window Technique	–
6	Solve	Sliding Window Problems	#121, #209, #643, #3
7	Concept	Strings & Character Handling	–
8	Solve	String Problems	#125, #242, #49, #344

Month 2: Linked Lists, Stack & Queue (March 2026)

#	Type	Topic	LeetCode
9	Concept	Singly Linked Lists	–
10	Solve	Linked List Problems	#206, #21, #83, #141
11	Concept	Fast & Slow Pointer Technique	–
12	Solve	Cycle / Middle Problems	#142, #876, #234
13	Concept	Stack Fundamentals	–
14	Solve	Stack Applications	#20, #155, #496
15	Concept	Queue & Deque	–
16	Solve	Queue Problems	#232, #933, #239

Month 3: Recursion & Backtracking (April 2026)

Important

This month is **intentionally extended** with extra practice sessions to ensure deep understanding of recursion concepts.

#	Type	Topic	LeetCode
17	Concept	Introduction to Recursion	–
18	Concept	Call Stack and Base Cases	–
19	Hybrid	Recursion on Arrays	#509, #70
20	Solve	Recursion Practice	#344, #206
21	Concept	Backtracking Fundamentals	–
22	Solve	Backtracking Problems	#46, #78, #77
23	Hybrid	Recursion Tree Analysis	#22, #17
24	Review	Full Recursion Review	Mixed

Month 4: Trees & Binary Search Trees (May 2026)

#	Type	Topic	LeetCode
25	Concept	Binary Tree Fundamentals	–
26	Solve	DFS Traversals	#94, #144, #145
27	Concept	BFS / Level Order Traversal	–
28	Solve	BFS Problems	#102, #199
29	Concept	Tree Properties	–
30	Solve	Height / Diameter	#104, #543
31	Concept	Binary Search Trees	–
32	Solve	BST Problems	#98, #230, #235

Month 5: Heap, Hashing & Sorting (June 2026)

#	Type	Topic	LeetCode
33	Concept	Heap and Priority Queue	–
34	Solve	Heap Problems	#215, #347, #295
35	Concept	Hash Tables	–
36	Solve	Hashing Problems	#1, #217, #451
37	Concept	Sorting Algorithms	–
38	Solve	Sorting Problems	#912, #56, #57
39	Hybrid	Custom Sorting / Comparators	#179, #75
40	Review	Monthly Review	Mixed

Month 6: Binary Search & Graphs (July 2026)

#	Type	Topic	LeetCode
41	Concept	Binary Search Fundamentals	–
42	Solve	Binary Search Problems	#704, #33, #34
43	Concept	Graph Representation (Adj List/Matrix)	–
44	Solve	DFS / BFS on Graphs	#200, #133, #417
45	Concept	Topological Sorting	–
46	Solve	Topological Problems	#207, #210, #269
47	Hybrid	Grid Graph Problems	#695, #994, #286
48	Review	Graph Review	Mixed

Month 7: Dynamic Programming & Final Revision (August 2026)

Key Takeaway

This is the **culmination** of everything you've learned. DP builds on recursion, trees, and pattern recognition. Stay focused!

#	Type	Topic	LeetCode
49	Concept	Dynamic Programming Introduction	–
50	Solve	1D Dynamic Programming	#70, #198, #213
51	Concept	Knapsack Pattern	–
52	Solve	Knapsack Problems	#416, #494, #322
53	Concept	LIS and LCS Patterns	–
54	Solve	LIS / LCS Problems	#300, #1143, #72
55	Solve	Advanced DP (Grid / 2D)	#62, #64, #221
56	Final	**Full Course Revision**	Top 150

9 PROBLEM-SOLVING PATTERNS YOU WILL MASTER

Two Pointers | Sliding Window | Fast & Slow

Binary Search | BFS / DFS | Backtracking

Monotonic Stack | Union-Find | Dynamic Programming

Key Takeaway

Mastering these **9 core patterns** will allow you to solve **90%+** of coding interview questions!

10 PREREQUISITES

✓ Basic understanding of any programming language	✗ No prior DSA knowledge required
✓ Familiarity with C++ syntax (variables, loops, functions)	✗ No competitive programming experience needed
✓ A computer with stable internet connection	✗ No computer science degree required
✓ Dedication to practice **1-2 hours daily**	✗ No previous interview experience needed

Pro Tip

Basic familiarity with C++ is enough. The course includes a **focused C++ review** session tailored specifically for problem solving in the first week!

11 FREQUENTLY ASKED QUESTIONS

Is this course suitable for beginners?

Yes! The course is designed for committed learners. Difficulty increases gradually with continuous support. We start from the basics and build up systematically.

Will we really solve all LeetCode Top 150 problems?

Absolutely. The course roadmap is meticulously designed to cover the entire LeetCode Top 150 Interview Questions list, plus additional problems for practice.

What if I miss a live session?

All sessions are **recorded** and available for **48 hours** after the live session. You can catch up and ask questions in the Discord community.

What if I struggle with problems?

Struggling is **expected and encouraged!** It is a necessary part of building problem-solving skills. The Discord community and weekly Q&A sessions are here to help.

Will I get a certificate?

Yes! Upon successful completion of the course and assignments, you will receive an official **Certificate of Completion** that you can add to your LinkedIn profile.

What is the refund policy?

We offer a **7-day money-back guarantee**. If you're not satisfied after the first week, you can request a full refund — no questions asked.

12 ENROLLMENT & INVESTMENT

Ready to Transform Your Career?

This course is an investment in your future. The skills you gain here are the same ones used by engineers at **Google**, **Meta**, **Amazon**, **Microsoft**, and **Apple**.

$150 USD

One-Time Payment | No Hidden Fees | Lifetime Value

Start Date: February 10, 2026
Duration: 7 Months
Class Size: Max 30 Students
Sessions: 56 Live Class
Community: Private Discord
Certificate: Upon Compl

7-Day Money-Back Guarantee

Not satisfied after the first week? Get a full refund, no questions asked.

Reserve Your Spot Today:

+20 101 162 0431

ma7moudalysalem@gmail.com

> *"This course is designed for students who want **real improvement**, not shortcuts. If you stay **consistent** and **trust the process**, your problem-solving skills will improve **dramatically**."*
>
> — **Mahmoud Salem**

13 ABOUT YOUR INSTRUCTOR

Mahmoud Salem

Software Engineer & Competitive Programmer

- 500+ LeetCode Problems Solved
- Former Google DSC Lead
- 4+ Years of Teaching Experience
- Multiple Hackathon Winner
- Mentored 200+ Students
- Expert in C++ and Python
- Active Open Source Contributor
- Published Technical Writer

Don't Wait — Start Your Journey Today!

Limited to 30 students. Secure your spot before it's too late.

+20 101 162 0431
ma7moudalysalem@gmail.com