Gammaprep Live Bootcamp - 16 Week Syllabus

This 16-week bootcamp by Gammaprep is an industry-aligned, mentor-driven program designed to make learners job-ready for software engineering, data, and system design interviews. The course is divided into 3 core modules that balance theory, implementation, and project-based learning.

Data Structures & Algorithms (8 Weeks)

A deep dive into problem-solving and algorithmic thinking to prepare for top technical interviews. This module will be taught in both C++ and Java to ensure complete language flexibility for interviews.

Week 1: Arrays, Strings, and Hashing

Learn efficient array manipulations, string operations, and hashing techniques to optimize lookups.

Week 2: Linked Lists, Stacks, and Queues

Understand how to build and manipulate these linear data structures and their real-world applications.

Week 3: Trees and Graphs

Explore tree traversal algorithms, graph representations, and BFS/DFS techniques with C++ STL and Java Collections.

Week 4: Dynamic Programming

Master overlapping subproblems, memoization, and tabulation through key examples like knapsack, LIS, and matrix chain multiplication.

Week 5: Greedy Algorithms

Learn to make locally optimal choices with global impact, covering problems like activity selection, Huffman coding, and minimum spanning tree.

Week 6: Backtracking and Recursion

Develop recursive solutions for constraint satisfaction problems like N-Queens, Sudoku, and subset generation.

Week 7: Searching and Sorting

Gain expertise in binary search, merge sort, quicksort, and other key algorithmic building blocks.

Week 8: Advanced Problem Solving

Tackle complex interview problems integrating multiple algorithmic paradigms and competitive programming patterns.

System Design (4 Weeks)

Understand how to design scalable, maintainable systems from scratch. This module includes both High-Level Design (HLD) and Low-Level Design (LLD).

Week 9: System Design Fundamentals (HLD)

Understand system design basics including scalability, reliability, consistency, and fault tolerance.

Week 10: Scalability and Performance

Dive into concepts like database sharding, caching, load balancing, queues, and message brokers.

Week 11: Database Design & Microservices Architecture

Design normalized databases, explore NoSQL vs SQL, and learn how to structure microservices.

Week 12: Low-Level Design (LLD)

Learn class design, OOP principles (SOLID), and UML diagrams. Topics include designing real-world systems like Parking Lot, BookMyShow, and Food Delivery apps.

Bonus Case Studies

Hands-on design discussions of large-scale systems like Netflix, Uber, Instagram, and Amazon.

Data Science & Machine Learning (4 Weeks)

This section focuses on foundational and applied data science and ML techniques used in real-world applications.

Week 13: Python for Data Science

Master Python libraries including NumPy, Pandas, and Matplotlib for data manipulation, analysis, and visualization.

Week 14: Statistics and Probability

Understand hypothesis testing, distributions, p-values, and data-driven decision-making.

Week 15: Machine Learning Algorithms

Implement algorithms like Linear Regression, Logistic Regression, Decision Trees, SVM, K-Means, and Random Forest.

Week 16: Deep Learning & Deployment

Introduction to neural networks using TensorFlow and PyTorch. Learn model evaluation, hyperparameter tuning, and deployment using Flask and FastAPI.

Bonus Topics

Feature engineering, model interpretability (SHAP, LIME), data preprocessing pipelines, and MLOps introduction.

Programming Languages Covered

Module	Languages / Tools
Data Structures & Algorithms	C++ and Java
System Design (HLD + LLD)	Language Independent (Conceptual & UML)
Data Science & Machine Learning	Python (NumPy, Pandas, Scikit-learn, TensorFlow, Flask)