Week 1: Introduction to Data Structures and Algorithms

- Overview of data structures and algorithms
- Why data structures and algorithms matter in programming
- Common terminology used in data structures and algorithms

Week 2: Arrays and Lists

- What are arrays and lists?
- How to implement arrays and lists in code
- Differences between arrays and lists
- Performance characteristics of arrays and lists

Week 3: Stacks and Queues

- What are stacks and queues?
- How to implement stacks and queues in code
- Differences between stacks and queues
- Applications of stacks and queues

Week 4: Trees

- What are trees?
- How to implement trees in code

- Different types of trees (binary, AVL, etc.)
- Traversing trees (pre-order, in-order, post-order)

Week 5: Graphs

- What are graphs?
- How to implement graphs in code
- Different types of graphs (directed, undirected, weighted, unweighted)
- Traversing graphs (breadth-first search, depth-first search)

Week 6: Sorting and Searching Algorithms

- Overview of common sorting and searching algorithms
- How to implement sorting and searching algorithms in code
- Performance characteristics of different algorithms
- Choosing the right algorithm for the task at hand

Week 7: Dynamic Programming

- Overview of dynamic programming
- How to implement dynamic programming in code
- Applications of dynamic programming

Week 8: Big O Notation

- What is Big O notation?
- How to analyze the time complexity of algorithms using Big O notation
- Comparing the performance of different algorithms using Big O notation

Week 9: Hash Tables

- What are hash tables?

- How to implement hash tables in code
- Collision resolution strategies for hash tables
- Applications of hash tables

Week 10: Advanced Topics

- Advanced data structures and algorithms (B-trees, skip lists, etc.)
- Parallel algorithms
- Approximation algorithms
- Randomized algorithms