DSA for FAANG preparation with Python and JavaScript

A comprehensive chase to excel any intreview for the Data Structures and Algorithms. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

Duration :　　Language : English　　Price : 25000

What you will learn?

- Analysis in Algorithms
- Data Structure Introduction
- Array Data Structure
- Interview Question on array
- Recursion in depth
- Divide and Conquer algorithm
- Applications of Divide and Conquer
- Linked List Data Structure
- Interview Question on Linked List
- Circular Linked List
- Doubly Linked List
- Skip List
- Stack and Queue
- Interview Question on Stack and Queue
- Hashing Data Structure
- Collision Resolution Techniques
- Tree Data Structure
- Tree Traversal
- Binary Search Tree
- Height Balanced Tree: AVL Tree

Features

- Free LCO DSA Bundle
- 18 hrs live support all seven day
- Course material
- Course resources
- On demand recorded videos

- Practical exercises
- Quizzes
- Assignments
- Course completion certificate

Requirements

- System with Internet Connection
- Interest to learn
- Dedication

Course Curriculum

Introduction

- Course Overview

Analysis in Algorithms

- Why we need Data structures and algorithms
- Introduction to algorithms and its analysis : Time and Space Complexity
- Asymptotic Notation: Big O, Omega and Theta Notation
- Recurrence Relation Solving: Substitution, Recursive Tree and Master's Theorem

Data Structure Introduction

- Memory Process - Stack and Heap
- Physical and logical data structures
- Abstract data types

Array Data Structure

- Introduction to arrays
- Concept of 1D and 2D array (row-major order and column-major order)
- Searching algorithm: linear, binary, ternary search
- Concept of inplace and outplace sorting algorithm
- Concept of stable and unstable sorting algorithm
- Sorting algorithm: comparison(selection, bubble, insertion, quicksort, mergesort, heapsort, shellsort)
- Sorting algorithm: Non-comparison(count sort, bucket sort, radix sort)

Interview Question on array

- Rotation of an array

- Finding of missing number in an array
- Division of two integers without using division operator
- Search in rotated array
- Target triplet
- Stock buy sell to maximize profit

Recursion in depth

- Introduction to recursion
- Tracing the recursion tree
- Types of recursion
- Complex recursion tree
- Classic Tower of Hanoi problem

Divide and Conquer algorithm

- Introduction to Divide and Conquer

Applications of Divide and Conquer

- Finding of maxima and minima
- Finding of power of an element
- Binary Search
- MergeSort
- QuickSort
- Selection Procedure
- Finding of number of inversions
- Strassens' matrix multiplication

Linked List Data Structure

- Introduction to linked list
- Insertion of a node(beginning, end and at any position) in linked list
- Deletion of a node(beginning, end and at any position) in linked list
- Searching of a node in linked list

Interview Question on Linked List

- Reversal of a node in linked list
- Count of all nodes in linkedlist
- Floyd's cycle detection algorithm
- Merge two linked list

Circular Linked List

- Circular Linked List Theory
- Insertion of a node in circular linked list
- Traversal of a node in circular linked list
- Deletion of a node in circular linked list
- Count of number of nodes in circular linked list
- Conversion of linked list to circular linked list

Doubly Linked List

- Doubly Linked List Theory
- Insertion of a node in doubly linked list
- Traversal of a node in doubly linked list
- Deletion of a node in doubly linked list

Skip List

- Introduction to skip list
- Build-in skip list
- Search in skip list
- Insertion in skip list
- Deletion in skip list

Stack and Queue

- Stack: Push and Pop operation
- Implementation of Stack using array and linked list
- Queue concept theory
- Implementation of Queue using array and linked list
- Circular Queue theory
- Implementation of Circular Queue

Interview Question on Stack and Queue

- Stack using queue conceptual understanding
- Implementation of stack using queue
- Queue using stack conceptual understanding
- Implementation of queue using stack
- Valid brackets
- Stock Spanning

Hashing Data Structure

- Introduction to Hashing Data Structure
- Hash Function and its types

Collision Resolution Techniques

- Chaining
- Open Addressing: Linear Probing, Quadratic Probing, Double Hashing, Perfect Hashing, Consistent Hashing
- Application: Bloom Filters
- Two Sum Problem

Tree Data Structure

- Introduction to Binary Tree
- Complete Binary Tree and almost complete binary tree
- Full binary tree and representation using array and linked list

Tree Traversal

- Introduction to tree traversal
- Inorder Traversal
- Preorder Traversal
- Postorder Traversal

Binary Search Tree

- Introduction to Binary Search Tree
- Insertion and Deletion in BST
- Inorder traversal in BST gives sorted array
- Searching in Binary Search Tree
- Deletion in Binary Search Tree

Height Balanced Tree: AVL Tree

- Introduction: Why AVL Tree?
- Creation of an AVL Tree
- Insertion in AVL Tree
- Searching in AVL Tree
- Deletion in AVL Tree

Height Balanced Tree: Red Black Tree

- Introduction: Why Red Black Tree?
- Properties of Red Black Tree
- Creating of Red Black Tree
- Insertion Rules in Red Black Tree
- Searching in Red Black Tree
- Deletion in Red Black Tree

B and B+ Tree: Usage in Databases

- Creation of B and B+ Tree
- Insertion in B and B+ Tree
- Searching in B and B+ Tree
- Deletion in B and B+ Tree

Interview Question on Tree

- Checking of whether the tree is symmetric or not
- Count of number of possible BSTs in a given number of nodes
- Catalan number concept to find the number of BST
- Level order traversal of a tree
- Flip or inverse of a binary tree
- Same tree problem
- Inorder iterator
- Binary Tree Zigzag level order traversal

Graph Traversal Algorithms

- Introduction to Graph Traversal Algorithms
- Introduction to Depth First Search
- DFS Psuedocode and illustration using an example
- DFS Coding Implementation
- Introduction to Breadth First Search
- BFS Psuedocode and illustration using an example

Interview Questions on Graph

- Clone of a graph
- DFS and Cycle detection with University course problem
- Islan problem

Heap Data Structure

- Introduction to Heap Data Structure
- Maxheap and Minheap Overview
- Insertion in Minheap
- Deletion in Minheap
- Creation of Minheap
- Mathematical derivation to analyse the complexity of creation of minheap
- HeapSort algorithm and why it is not stable algorithm

Interview Based Question on Heap Data Structure

- Maximum Product of three numbers in an array
- Finding of K-closest points from an origin

Greedy Algorithm

- Introduction to greedy algorithm

Application of greedy algorithm

- Fractional Knapsack Problem
- Minimum Spanning Tree: Kruskal and Prim's Algorithm
- Single Source Shortest Path: Dijkstra's algorithm
- Huffman Coding
- Optimal Merge Pattern
- Job Sequencing with Deadline

Dynamic Programming

- Introduction to Dynamic Programming
- Overlapping subproblem in dynamic programming
- Tabulation in dynamic programming
- Memoization in dynamic programming

Application of Dynamic Programming

- Fibonacci Series
- Longest Common Subsequence
- 0/1 Knapsack Problem
- Sum of subset
- All Pair Shortest Path: Floyd Warshall Algorithm
- Bellman Ford Algorithm

Interview Problems on Dynamic Programming

- Knapsack - Coke, Pepsi, Redbull
- Largest sum of subset
- Coin change problem
- Largest sum
- Minimum path to reach target

String Matching Algorithms

- Introduction to String matching algorithms
- Naive String Matching algorithms
- Rabin Karp Algorithm
- Kuth-Morris-Pratt(KMP) Pattern Matching

Interview Problems on String

- Word in a sentence
- Inplace duplicates
- Longest substring
- Palindrome makes and breaks

NP-Hard and NP-Complete Problem

- NP-Hard Problem
- NP-Complete Problem

Approaching Design

- Understanding and clarification
- Business usecase of the problem and knowing the consumers
- Iron out the Functional requirements
- Importance of discussing the trade-offs based on the usecase in picture
- Mastering the art of selling design
- Data model approaches and fitment
- LLD modelling and future readiness of design
- Explaining the features of design like adherance to proper design patterns

Introduction to System Design

- Introduction to system design
- Importance of architecture
- Distinction between HLD and LLD

- Importance of data modelling
- Importance of documentation in design

Practicing some real designs

- Rate limitting
- Uber riders app
- Whatsapp messaging
- food delivery app building
- Booking app building
- Video streaming systems
- Q&A