

# **DSA**Self Paced



Detailed Course Syllabus



# 1) Introduction

## • Analysis of Algorithm

a) Background analysis through a Program and its functions.

#### • Order of Growth

- a) A mathematical explanation of the growth analysis through limits and functions.
- b) A direct way of calculating the order of growth

## • Asymptotic Notations

o Best, Average and Worst case explanation through a program.

#### • Big O Notation

- o Graphical and mathematical explanation.
- Calculation
- Applications at Linear Search

#### • Omega Notation

- o Graphical and mathematical explanation.
- Calculation.

#### • Theta Notation

- o Graphical and mathematical explanation.
- o Calculation.

## • Analysis of common loops

o Single, multiple and nested loops

#### • Analysis of Recursion

o Various calculations through Recursion Tree method

## • Space Complexity

- Basic Programs
- Auxiliary Space
- Space Analysis of Recursion
- Space Analysis of Fibonacci number

#### • Practice Problems

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.



# 2) Mathematics

#### Mathematics

- Count Digits
- Palindrome Numbers
- Factorial of Numbers
- GCD of Two Numbers
- LCM of Two Numbers
- Check for Prime
- Prime Factors
- Sieve of Eratosthenes
- Computing Power

#### • Practice Problems

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 3) Bit Magic

#### • Bitwise Operators in C++

- Operation of AND, OR, XOR operators
- o Operation of Left Shift, Right Shift and Bitwise Not

## • Bitwise Operators in Java

- o Operation of AND, OR
- o Operation of Bitwise Not, Left Shift
- Operation of Right Shift and unsigned Right Shift

## • Problem(With Video Solutions): Check Kth bit is set or not

- Method 1: Using the left Shift.
- Method 2: Using the right shift

## • Problem(With Video Solutions): Count Set Bits

- Method 1: Simple method
- o Method 2: Brian and Kerningham Algorithm
- Method 3: Using Lookup Table

## • Problems(With Video Solutions):

- o To check whether a number is a power of 2 or not
- Odd occurrences in an array.
- o Two numbers having odd occurrences in an array.
- o Generate power set using bitwise operators.



 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 4) Recursion

- Introduction to Recursion
- Applications of Recursion
- Writing base cases in Recursion
  - Factorial
  - o N-th Fibonacci number

## • Various problems on Recursion(With Video Solutions)

- o Print n to 1
- o Print 1 to n
- o Tail Recursion
- Checking Palindrome
- Sum of digits
- Rod cutting
- Subsets of a set
- o Tower of Hanoi Problem
- Josephus Problem

#### Practice Problems

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 5) Arrays

- Introduction and Advantages
- Types of Arrays
  - Fixed-sized array
  - Dynamic-sized array
- Operations on Arrays
  - Searching
  - Insertions
  - Deletion



- Arrays vs other DS
- o Reversing Explanation with complexity

## • Problems(With Video Solutions)

- Left Rotation of the array by 1
- Check if Sorted
- o Left Rotation of the array by D places
- Leaders in an Array
- Maximum Difference Problem
- Frequencies in Sorted Array
- Stock Buy and Sell Problem
- Trapping Rainwater Problem
- Maximum Consecutive Is
- Maximum Subarray Sum
- Longest Even-Odd Subarray
- o Maximum Circular sum subarray.
- Majority Element
- Minimum Consecutive Flips
- Sliding Window Technique
- o Prefix Sum Technique

#### Practice Problems

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 6) Searching

- Binary Search Iterative and Recursive
- Binary Search and various associated problems(With Video Solutions)
  - Index of First Occurence in Sorted Array
  - Index of Last Occurence in Sorted Array
  - o Count of occurrences of x in sorted element
  - Count of 1s in a binary sorted array
  - Find an element in sorted and rotated array
  - Peak element
  - Find an element in an infinite sized sorted array
  - o The square root of an integer

## • Two Pointer Approach Problems(With Video Solutions)

- o Find pair in an unsorted array which gives sum X
- Find pair in a sorted array which gives sum X



- Find triplet in an array which gives sum X
- Problems(With Video Solutions)
  - Median of two sorted arrays
  - Majority Element
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 7) Sorting

- Implementation of C++ STL sort() function in Arrays and Vectors
  - Time Complexities
- Sorting in Java
- Arrays.sort() in Java
- Collection.sort() in Java
- Stability in Sorting Algorithms
  - Examples of Stable and Unstable Algos
- Bubble Sort
- Selection Sort
- Insertion Sort
- Merge Sort
- Problems(With Video Solutions)
  - Intersection of 2 sorted arrays
  - Union of 2 sorted arrays
  - Count Inversions in arrays
- Partitions(With Video Solutions)
  - Naive
  - Lomuto
  - Hoare
- Quick Sort
  - Using Lomuto and Hoare
  - Time and Space analysis
  - Choice of Pivot and Worst case
  - o Tail call elimination
- Problems(With Video Solutions)
  - Kth Smallest element
  - o Chocolate Distribution Problem
  - Sorting arrays with 2 and 3 types of elements



- Merge Overlapping Intervals
- Meeting the Maximum Guests
- Heap Sort
- Cycle Sort
- Counting Sort
- Radix Sort
- Bucket Sort
- Overview of Sorting Algorithms
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 8) Matrix

- Introduction to Matrix in C++ and Java
- Multidimensional Matrix
- Pass Matrix as Argument
- Printing matrix in a snake pattern
- Transposing a matrix
- Rotating a Matrix
- Check if the element is present in a row and column-wise sorted matrix.
- Boundary Traversal
- Spiral Traversal
- Matrix Multiplication
- Search in row-wise and column-wise Sorted Matrix
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 9) Hashing

• Introduction and Time complexity analysis



- Application of Hashing
- Discussion on Direct Address Table
- Working and examples on various Hash Functions
- Introduction and Various techniques on Collision Handling
- Chaining and its implementation
- Open Addressing and its Implementation
- Chaining V/S Open Addressing
- Double Hashing
- C++
  - Unordered Set
  - Unordered Map
- Java
  - HashSet
  - HashMap

## • Problems(With Video Solutions):

- Count Distinct Elements
- Count of the frequency of array elements
- The intersection of two arrays
- Union of two unsorted arrays
- o Pair with given sum in an unsorted array
- Subarray with zero-sum
- Subarray with given sum
- Longest subarray with a given sum
- Longest subarray with an equal number of 0's and 1's
- o Longest common span with the same sum in a binary array
- Longest Consecutive Subsequence
- o Count Distinct elements in every window
- More than n/k Occurences
- Optimized More than n/k Solution

## • Practice Problems

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 10) Strings

- Discussion of String DS
- Strings in CPP
- Strings in Java



## Problems(With Video Solutions):

- o Given a string, check if they are an anagram of each other.
- o Given a string, find the leftmost character that repeats.
- o Given a string, find the leftmost character that does not repeat.
- o Given a string, find the lexicographic rank of it in O(n) time.
- Implementation of the previously discussed lexicographic rank problem.
- Given a text string and a pattern string, find if a permutation of the pattern exists in the text.
- o Given two strings, check if they are rotations of each other or not.
- o Various Pattern Searching Algorithms.
- o Palindrome Check
- Rabin Karp Algorithm
- KMP Algorithm
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 11) Linked List

#### Introduction

- Implementation in CPP
- o Implementation in Java
- Comparison with Array DS
- Doubly Linked List
- Circular Linked List
- Loop Problems
  - Detecting Loops
  - Detecting loops using Floyd cycle detection
  - Detecting and Removing Loops in Linked List

#### Problems(With Video Solutions):

- o Middle of Linked List
- o Nth node from the end of linked list
- Deleting a Node without accessing Head pointer of Linked List
- o An iterative method to Reverse a linked list
- o Recursive method to reverse a linked list
- o Reverse in group of size k
- o Recursive Traversal in a Singly Linked List



- Segregating even-odd nodes of linked list
- o The intersection of two linked list
- Pairwise swap nodes of linked list
- o Clone a linked list using a random pointer
- LRU Cache Design
- Merge two Sorted Linked Lists
- o Palindrome Linked List
- Recursive Traversal in a Singly Linked List
- o Remove Duplicates from a Sorted Singly Linked List
- Sorted Insert in a Singly Linked List
- Reverse a Doubly Linked List

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 12) Stack

- Understanding the Stack data structure
- Applications of Stack
- Implementation of Stack in Array and Linked List
  - o In C++
  - o In Java

## Problems(With Video Solutions):

- Balanced Parenthesis
- Two stacks in an array
- K Stacks in an array
- Stock span problem with variations
- o Previous Greater Element
- Next Greater Element
- Largest Rectangular Area in a Histogram
- Understanding getMin() in Stack with O(1)
- Infix, Prefix and Postfix Introduction
  - Infix to Postfix (Simple Solution)
  - Infix to Postfix (Efficient Solution)
  - Evaluation of Postfix
  - Infix to Prefix (Simple Solution)



- Infix to Prefix (Efficient Solution)
- Evaluation of Prefix

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

## 13) Queue

- Introduction and Application
- Implementation of the queue using array and LinkedList
  - o In C++ STL
  - o In Java
  - Stack using queue
- Problems(With Video Solutions)
  - o Reversing a Queue
  - o Generate numbers with given digits
  - First Circular Tour

#### • Practice Problems

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 14) Deque

- Introduction and Application
- Implementation
  - o In C++ STL
  - o In Java
- Problems(With Video Solutions)
  - o Maximums of all subarrays of size k
  - o ArrayDeque in Java
  - o Design a DS with min max operations
- Practice Problems



 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

## **15) Tree**

#### Introduction

- Tree
- Application
- o Binary Tree
- Tree Traversal

## • Implementation of:

- Inorder Traversal
- Preorder Traversal
- Postorder Traversal
- Level Order Traversal (Line by Line)
- o Tree Traversal in Spiral Form

## • Problems(With Video Solutions):

- o Size of Binary Tree
- Maximum in Binary Tree
- Height of Binary Tree
- o Print Nodes at K distance
- Print Left View of Binary Tree
- Children Sum Property
- Check for Balanced Binary Tree
- o Maximum Width of Binary Tree
- Convert Binary Tree to Doubly Linked List
- Construct Binary Tree from Inorder and Preorder
- o Tree Traversal Spiral Form
- o The diameter of a Binary Tree
- LCA problem with an efficient solution
- Burn A Binary Tree from a Leaf
- o Count Nodes in a complete Binary Tree
- o Serialize and Deserialize a Binary tree
- o Iterative Inorder Traversal
- Iterative Preorder Traversal (Simple)
- o Iterative Preorder Traversal (Space Optimized)

#### Practice Problems



 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

## 16) Binary Search Tree

- Background, Introduction and Application
- Implementation of Search in BST
  - o In CPP
  - o In Java
- Insertion in BST
  - o In CPP
  - o In Java
- Deletion in BST
  - o In CPP
  - o In Java
- Floor in BST
  - o In CPP
  - o In Java
- Self Balancing BST
- AVL Tree
- Red Black Tree
- Set in C++ STL
- Map in C++ STL
- BST Introduction
- TreeSet in java
- TreeMap in Java
- Problems(With Video Solutions):
  - o The ceiling of a key in BST
  - Ceiling on the left side in an array
  - o Find Kth Smallest in BST
  - Check for BST
  - Fix BST with Two Nodes Swapped
  - Pair Sum with given BST
  - Vertical Sum in a Binary Tree
  - Vertical Traversal of Binary Tree
  - Top View of Binary Tree
  - Bottom View of Binary Tree
- Practice Problems



 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 17) **Heap**

- Introduction & Implementation
- Binary Heap
  - Insertion
  - Heapify and Extract
  - o Decrease Key, Delete and Build Heap
- Heap Sort
- Priority Queue in C++
- PriorityQueue in Java
- Problems(With Video Solutions):
  - Sort K-Sorted Array
  - o Buy Maximum Items with Given Sum
  - K Largest Elements
  - Merge K Sorted Arrays
  - Median of a Stream
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 18) Graph

- Introduction to Graph
- Graph Representation
  - o Adjacency Matrix
  - o Adjacency List in CPP and Java
  - Adjacency Matrix VS List
- Breadth-First Search
  - Applications
- Depth First Search
  - Applications



- Problems(With Video Solutions):
  - Shortest Path in an Unweighted Graph
  - Detecting Cycle
    - In the Undirected Graph
    - In the Directed Graph
  - Topological Sorting
    - Kahn's BFS Based Algorithm
    - DFS Based Algorithm
- Shortest Path in Directed Acyclic Graph
- Prim's Algorithm/Minimum Spanning Tree
  - o Implementation in CPP
  - o Implementation in Java
- Dijkstra's Shortest Path Algorithm
  - o Implementation in CPP
  - o Implementation in Java
- Bellman-Ford Shortest Path Algorithm
- Kruskal's Algoritm
- Kosaraju's Algorithm
- Articulation Point
- Bridges in Graph
- Tarjan's Algorithm
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

## 19) Greedy

- Introduction
- Activity Selection Problem
- Fractional Knapsack
- Job Sequencing Problem
- Huffman Coding
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.



# 20) Backtracking

- Concepts of Backtracking
- Rat In a Maze
- N Queen Problem
- Sudoku Problem
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 21) Dynamic Programming

- Introduction
- Dynamic Programming
  - Memoization
  - Tabulation
- Problems(With Video Solutions):
  - Longest Common Subsequence
  - Coin Change Count Combinations
  - Edit Distance Problem
    - Naive Approach
    - DP Approach
  - Longest Increasing Subsequence Problem
    - Naive Approach
    - Efficient Approach
  - Maximum Cuts
  - Minimum coins to make a value
  - o Minimum Jumps to reach at the end
  - o 0-1 knapsack problem
    - Naive Approach
    - Efficient Approach
  - Optimal Strategy for a Game
  - Variation of Longest Common Subsequence
  - Variation of Longest Increasing Subsequence
  - Egg Dropping Problem
  - Count BST with nkeys
  - Maximum Sum with No Consecutive
  - Subset Sum Problem



- o Matrix Chain Multiplication
- o Palindrome Parititioning

 This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# **22) Trie**

- Introduction
  - Representation
  - Search
  - Insert
  - o Delete
- Count Distinct Rows in a Binary Matrix
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

## 23) Segment Tree

- Introduction
- Construction
- Range Query
- Update Query
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.

# 24) Disjoint Set

- Introduction
- Find and Union Operations
- Union by Rank
- Path Compression



- Kruskal's Algorithm
- Practice Problems
  - This track contains many practice problems for the users which are considered important and must-do as far as Data Structure and Algorithm is concerned.