**Course Title:** DSA (Data Structure Algorithm) with Java

**Duration:** 30 hours

**Module 1**: Introduction to Data Structures and Algorithms

* Data Structures and Algorithms
* Measuring Running time of Algorithms
* Big O Notation
* Big O analysis of Algorithms
* Finding Time complexity
* Worst case, Best case and Average Case Analysis
* Common Complexities
* Abstract Data Types

**Module 2**: Linked List

* Linked List Introduction
* Traversing and Searching a Single Linked List
* Finding references in a single linked list
* Insertion in a Single Linked List
* Deletion in a Single Linked List
* Reversing a Single Linked List
* Sorting a Linked list using Bubble Sort
* Sorting a Linked list using Merge Sort
* Doubly linked list
* Insertion in a doubly linked List
* Deletion from doubly linked list
* Reversing a doubly linked list
* Circular linked list
* Insertion in a circular Linked List
* Deletion in a circular linked list
* Linked List with Header Node

**Module 3**: Stack and Queue

* Introduction to Stack and Queue
* Array Implementation of Stack
* Linked List Implementation of Stack
* Array Implementation of Queue
* Linked List implementation of Queue
* Queue through Circular Linked List
* Circular Queue
* Deque
* Priority Queue
* Checking validity of an expression containing nested parentheses
* Evaluating Arithmetic Expressions
* Polish Notations
* Converting infix expression to postfix expression
* Evaluation of postfix expression

**Module 4**: Recursion

* Introduction
* Flow of control in Recursive functions
* Winding and unwinding phase
* Factorial
* Printing numbers from 1 to n
* Sum of digits of an integer
* Base conversion
* Finding nth power of a number
* Fibonacci Series
* Tower of Hanoi
* Tail recursion
* Recursion vs. Iteration

**Module 5**: Binary Tree

* Introduction to trees
* Strictly Binary Tree and Extended Binary Tree
* Full binary tree and Complete Binary Tree
* Array Representation of Binary trees
* Linked Representation of Binary Trees
* Traversal in Binary Tree
* Preorder Traversal
* Inorder Traversal
* Postorder Traversal
* Level order traversal
* Finding height of a Binary tree
* Constructing Binary tree from Traversals

**Module 6**: Binary Search Tree

* Introduction
* Traversal in Binary Search Tree
* Searching in a Binary Search Tree
* Nodes with Minimum and Maximum key
* Insertion in a Binary Search Tree
* Deletion in a Binary Search Tree

**Module 7**: Heap

* Introduction
* Heap Representation
* Insertion in Heap
* Deletion
* Building a heap
* Heap Applications

**Module 8**: Sorting

* Introduction
* Selection Sort
* Selection Sort : Example
* Selection Sort in Java
* Bubble Sort
* Bubble Sort : Example
* Bubble Sort in Java
* Improvement in Bubble Sort
* Insertion Sort
* Insertion Sort : Example
* Insertion Sort in Java
* Shell Sort
* Shell Sort : Example
* Shell Sort in Java
* Merging two Sorted Arrays
* Recursive Merge Sort
* Recursive Merge Sort in Java
* Iterative Merge Sort
* Iterative Merge Sort in Java
* Quick Sort
* Binary tree sort
* Binary Tree Sort in Java
* Heap Sort
* Radix Sort

**Module 9**: Searching

* Linear Search
* Variations of Linear Search
* Binary Search
* Implementation of Binary Search
* Analysis of Binary Search

**Module 10**: Hashing

* Direct Addressing
* Hashing
* Collisions
* Hash Functions
* Open Addressing : Linear Probing
* Open Addressing : Quadratic Probing
* Open Addresssing : Double Hashing
* Deletion in Open Addressed Tables
* Implementation of Open Addressing
* Separate Chaining