PROGRAMMING CLASSES

UNIT OF MAGENOTO SOFTWARE PVT LTD
AN ISO 9001:2015 CERTIFIED

SYLLABUS OF DATA STRUCTURE

Introduction to Data Structure
- ✓ Overview of data structure
- ✓ Environment Setup

Algorithms
- ✓ Algorithms Basics
- ✓ Asymptotic Analysis
- ✓ Greedy Algorithms
- ✓ Divide and Conquer
- ✓ Dynamic Programming

Performance Analysis
- ✓ Time complexity
- ✓ Space complexity

Asymptotic Notations-
- ✓ Big O
- ✓ Omega
- ✓ Theta notations

Arrays

Structures

Pointers

Dynamic Memory allocation
- ✓ Malloc()
- ✓ calloc()
- ✓ realloc()
- ✓ free()

- ✓ Stacks
- ✓ Stack Operations
- ✓ push()
- ✓ pop()
- ✓ peex()
- ✓ distzay()
- ✓ isEmpty()
- ✓ isFull()
- ✓ Stack implementation using arrays
- ✓ Applications
- ✓ Decimal to Binary
- ✓ String reverse
- ✓ Number reverse
- ✓ Recursion – Towers of Hanoi
- ✓ Balanced Parentheses
- ✓ Expressions
- ✓ Stack Implementation using pointer (dynamic)

Expression

- ✓ Introduction to Notations
- ✓ Importance of Notations in expression evaluation
- ✓ Conversion Algorithms
- ✓ Infix to prefix
- ✓ Infix to postfix
- ✓ Prefix to infix
- ✓ Prefix to postfix
- ✓ Postfix to infix
- ✓ Postfix to prefix
- ✓ Implementation of all the conversions

Queues

- ✓ Operations on Queue – enqueue(), dequeue()
- ✓ Queue implementation using static arrays
- ✓ Applications
- ✓ Queues Implementations using pointer (dynamic)

Circular queues

Double Ended queue (Deques)

Single linked list

- ✓ Introduction
- ✓ Construction
- ✓ Length
- ✓ Insertion
- ✓ Deletion
- ✓ Sort
- ✓ Reverse list
- ✓ Swap node data
- ✓ Swap nodes
- ✓ Applications

Stack implementation using linked list

Queue implementation using linked list

Doubly linked list

Circular linked list

Circular Doubly Linked List

Binary Tree

- ✓ Terminology
- ✓ Differences between Tree and Binary Tree
- ✓ Binary Tree Representations
- ✓ Expression Trees
- ✓ Traversals
- ✓ In-order
- ✓ pre-order
- ✓ post-order

Binary Search Tree

- ✓ Introduction to BST
- ✓ Insertion
- ✓ Deletion
- ✓ Search
- ✓ Implementation

Graph

- ✓ Introduction & Terminology
- ✓ Graph Representations

- ✓ Traversal
- ✓ BFS (Breadth First Search)
- ✓ DFS (Depth First Search)

Searching Algorithms

- ✓ Linear search
- ✓ Binary search

Sorting Algorithms

- ✓ Bubble sort
- ✓ Selection sort
- ✓ Insertion sort
- ✓ Heap sort
- ✓ Merge sort
- ✓ Quick sort

AVL Trees

- ✓ Introduction
- ✓ BST v/s AVL
- ✓ Rotations
- ✓ L-L-Rotation
- ✓ R-R-Rotation
- ✓ L-R-Rotation
- ✓ R-L-Rotation
- ✓ Insertion
- ✓ Deletion
- ✓ Traversal

Red Black Trees

- ✓ Introduction
- ✓ BST v/s AVL v/s RBT
- ✓ Rotations
- ✓ L-L-Rotation
- ✓ R-R-Rotation
- ✓ L-R-Rotation
- ✓ R-L-Rotation
- ✓ Insertion
- ✓ Deletion

B trees

- ✓ M-way Search Tree
- ✓ Search
- ✓ Insertion
- ✓ Deletion

Hashing

- ✓ Hash Table representation
- ✓ Hash function-Division Method
- ✓ Collision
- ✓ Collision Resolution Techniques
- ✓ Separate Chaining
- ✓ open addressing
- ✓ linear probing
- ✓ quadratic probing
- ✓ double hashing
- ✓ Rehashing

Priority Queue-Definition

- ✓ Operations-Insertion, Deletion,

Heap

- ✓ Definition
- ✓ Max Heap

- ✓ **Min Heap**
- ✓ **Insertion and deletion**

Pattern matching algorithms

- ✓ **Brute force**
- ✓ **Boyer –Moore algorithm**
- ✓ **Knuth-Morris-Pratt algorithm**

Tries

- ✓ **Standard Tries**
- ✓ **Compressed Tries**
- ✓ **Suffix tries**

Dynamic Programming

Greedy Method

Divide and conquer method