

CSE205: DATA STRUCTURES AND ALGORITHMS

L: 3 T : 0 P : 2 Credits: 4

Course Outcomes: Through this course students should be able to

- CO1 :: describe the process to find efficiency of algorithms using asymptotic notations.
- CO2 :: develop skills to compare various data structure algorithms.
- CO3 :: illustrate the importance of data structures in context of writing efficient programs.
- CO4 :: identify appropriate data structures in problem solving.
- CO5 :: recommend to improve existing code using learned algorithms and data structures.
- CO6 :: construct new solutions for programming problems.

- Unit I** **Introduction:** Basic Data Structures, Basic Concepts and Notations, Complexity analysis: time space and trade off, Omega Notation, Theta Notation, Big O notation
Arrays: Linear arrays: memory representation, Traversal, Insertion, Deletion, Searching, Merging and their complexity analysis.
Sorting and Searching: Bubble sort, Insertion sort, Selection sort
- Unit II** **Linked Lists:** Introduction, Memory representation, Allocation, Traversal, Insertion, Deletion, Header linked lists: Grounded and Circular, Two-way lists: operations on two way linked lists
- Unit III** **Stacks:** Introduction: List and Array representations, Operations on stack (traversal, push and pop), Arithmetic expressions: polish notation, evaluation and transformation of expressions
Queue: Array and list representation, operations (traversal, insertion and deletion), Priority Queues, Deques
- Unit IV** **Recursion:** Introduction, Recursive implementation of Towers of Hanoi, Merge sort, Quick sort
Trees : Binary trees - introduction (complete and extended binary trees), memory representation (linked, sequential), Pre-order traversal, In-order traversal, Post-order traversal using recursion, Binary Search Tree- searching, insertion, deletion
- Unit V** **AVL trees and Heaps:** AVL trees - introduction, AVL trees Insertion, AVL trees Deletion, Heaps - Insertion, Heapify, Deletion, Heap Sort, Huffman algorithm
- Unit VI** **Graphs:** Warshall's algorithm, Shortest path algorithm Floyd Warshall Algorithm (modified Warshall algorithm), Graph Traversal: BFS, DFS
Hashing : Hashing Introduction, Hash Functions, Hash Table, Closed hashing (open addressing), Linear Probing, Quadratic Probing, Double Hashing, Open hashing (separate chaining)

List of Practical:

- Arrays:** Program to implement insertion and deletion operations in arrays
- Searching:** Program to implement different searching techniques - linear and binary search
- Sorting:** Program to implement different sorting techniques – bubble, selection and insertion sort
- Linked List:** Program to implement searching, insertion and deletion operations in linked list
- Doubly Linked List:** Program to implement searching, insertion and deletion operations in doubly linked list
- Stacks:** Program to implement push and pop operations in stacks using both arrays and linked list
- Queues:** Program to implement enqueue and dequeue operations in queues using both arrays and linked list
- Recursions:** Program to demonstrate concept of recursions with problem of tower of Hanoi
- Recursive Sorting:** Program to implement recursive sorting techniques - merge sort, quick sort

Trees: Program to create and traverse a binary tree recursively

Binary Search Tree: Program to implement insertion and deletion operations in BST

Heaps: Program to implement insertion and deletion operations in Heaps and Heap Sort

Text Books:

1. DATA STRUCTURES by SEYMOUR LIPSCHUTZ, MCGRAW HILL EDUCATION

References:

1. DATA STRUCTURES AND ALGORITHMS by ALFRED V. AHO, JEFFREY D. ULLMAN AND JOHN E. HOPCROFT, PEARSON