L	T	P	C
2	1	0	3

Course Code: CSE209

DATA STRUCTURES & ALGORITHMS

Course Objectives:

This course will help the learner to understand the algorithm performance analysis, linear and non-linear data structures. The learner will be able to choose appropriate data structures and efficient searching and sorting techniques for a given application.

UNIT - I **11 Periods**

Basic Terminologies & Introduction to Algorithm and Data Organisation: Introduction - Algorithm specification - Recursion - Performance analysis. Asymptotic Notation - The Big-O - Omega and Theta notation. Programming Style - Refinement of Coding - Time complexity-Space complexity- Trade Off - Testing - Data Abstraction

UNIT - II **11 Periods**

Linear Data Structure: Array - Abstract data type-Polynomial ADT - Stack - Stack ADT - Evaluation of expression - Queue - Queue ADT - Linked-list and its types - SLL, CLL, DLL - Various Representations - Operations & Applications of Linear Data Structures.

UNIT - III **12 Periods**

Non-linear Data Structure: Trees (Binary Tree, Threaded Binary Tree, Binary Search Tree, B & B+ Tree, AVL Tree, Splay Tree) and Graphs (Directed, Undirected) - Various Representations: Tree - List - left child-right sibling - degree two tree - Graph - Adjacency matrix - adjacency List - Multi list - weighted edges. **Operations:** insertion - deletion-search and traversal algorithms and complexity analysis. Applications of Non-Linear Data Structures.

UNIT - IV **11 Periods**

Searching and Sorting on Various Data Structures: Sequential Search - Binary Search - Breadth First Search - Depth First Search - Insertion Sort - Selection Sort - Shell Sort - Divide and Conquer Sort - Merge Sort - Quick Sort - Heap Sort - Introduction to Hashing - static hashing - Dynamic hashing.
File: Organisation (Sequential, Direct, Indexed Sequential, Hashed) and various types of accessing schemes.

TEXTBOOKS

1. E. Horowitz, S. Sahni, and S. Anderson-Freed. *Fundamentals of Data Structures in C*, Second Edition, University Press, 2008.
2. Alfred V. Aho, John E. Hopperoft, Jeffrey D. UIlman, *Data Structures and Algorithms*, Pearson Education, First Edition, Reprint 1987.

REFERENCES

1. Donald E. Knuth, *The Art of Computer Programming: Volume 1: Fundamental Algorithms.*, Addison-Wesley, Pearson Education, Third Edition, 1997
2. Thomas, H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, *Introduction to Algorithms*, Third Edition, 2010.
3. Morin, Pat. *Open Data Structures: An Introduction. Vol. 2*, Athabasca University Press, Thirty first Edition, 2013.

UNITWISE LEARNING OUTCOMES

Upon successful completion of each unit, the learner will be able to

Unit I	• Define asymptotic notations for time complexity analysis • Find time complexity for a given algorithm
Unit II	• Choose appropriate representation of a Linear data structure • Identify linear data structures like stack, queue and linked lists operations • Employ appropriate linear data structure in application development
Unit III	• Choose appropriate representation of a tree and graph • Identify non-linear data structures like Trees and graphs operations • Employ appropriate non-linear data structure in application development
Unit IV	• Select appropriate searching and sorting algorithm based on time complexity analysis for a given application • Choose appropriate hashing technique for the given key valued data

COURSE LEARNING OUTCOMES

Upon successful completion of this course, the learner will be able to

- Define asymptotic notations for time complexity analysis
- Find time complexity for a given algorithm
- Choose appropriate representation of a Linear and non linear data structure
- Identify linear and non linear data structures operations
- Employ appropriate linear and non linear data structure in application development
- Select appropriate searching and sorting algorithm based on time complexity analysis for a given application
- Choose appropriate hashing technique for the given key valued data