# 20CS1202 - DATA STRUCTURES

(Common to CSE, IT and AI&DS)

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Category:** | Professional Core | **Credits:** | 3 |
| **Course Type:** | Theory | **Lecture - Tutorial - Practical:** | 3-0-0 |
| **Prerequisite:** | Knowledge in programming languages. | **Sessional Evaluation:**  **Univ. Exam Evaluation:**  **Total Marks:** | 40  60  100 |
| **Objectives** | * Master the implementation of linked data structures such as linked lists and binary trees. * Familiar with advanced data structures such as balanced search trees and priority queues. * Familiar with several sorting algorithms including quick sort, and merge sort. * Familiar with some graph traversals like DFS, BFS. | | |

|  |  |  |
| --- | --- | --- |
| **Course Outcomes** | Upon successful completion of this course students will be able to: | |
| CO1 | Understand concepts of Data Structures and Learn sorting & searching techniques. |
| CO2 | Implement stacks and queues using arrays. |
| CO3 | Gain knowledge in Linked lists and types. |
| CO4 | Understand the concepts of Binary trees, Binary search trees and Graphs. |
| CO5 | Explore the basics of balanced search trees - AVL trees, Splay trees. |
| CO6 | Acquire knowledge in B-Trees and Hash tables. |
| **Course Content** | UNIT - I  **Introduction to Data Structures:** Primitive, non-primitive, Linear, non-linear  **Searching:** Linear Search and Binary Search.  **Sorting Techniques:** Bubble Sort, Selection Sort, Quick sort, Merge sort, Insertion Sort, Sorting Efficiency.    UNIT - II  **Stacks:** Introduction, Stack operations, Implementation of Stacks using Arrays  **Applications:** Conversion from Infix to Postfix notation, Evaluation of Postfix Expression  **Queues**: Introduction, operations on Queues, Circular Queues, Priority Queues, Double Ended Queues (deques), Applications of Linear and Priority Queues.  UNIT - III  **Linked Lists:** Introduction, Linked List Operations, Applications.  **Types:** Singly, Doubly and Circularly Linked Lists.  **Implementation:** Stacks and Queues using Linked Lists.  UNIT - IV  **Tree**: Definition, Representation.  **Binary Tree**: Definition and Properties, Representation, Tree traversals.  **Binary Search Tree**: Definition and Properties, applications.  **Graphs:** Introduction, Basic terminologies, Representation, Graph traversals.  UNIT - V  **Balanced Search Trees:** AVL trees: Definition, operations  **Red-Black Trees:** Definition, Representation and operations,  **Splay Trees:** Definition, Splay Rotations.  UNIT - VI  **B-Trees**: Indexed Sequential Access Method (ISAM), m-way search trees, B-trees of order m, Height of B-Tree, Insertion and Deletion from B-Tree, Introduction to B+ trees.  **Hash Tables**: Dictionaries, Hash Table Structure, Hash Functions.  **Collision Resolution:** Linear Probing and Chaining. | |
| **Text Books and References** | Text Books:   1. Computer Programming and Data Structures by E. Balagurusamy, 4/e, McGraw Hill. 2. Data Structures and Algorithms – concepts, Techniques and Applications by G A V Pai, McGraw Hill. | |
| Reference Books:   1. C Programming & Data Structures, B. A. Forouzan and R. F. Gilberg, Third Edition, Cengage Learning. 2. An Introduction to Data structures with applications: Tremblay J P and Sorenson P G. | |
| **E-Resources** | 1. <https://nptel.ac.in/courses> 2. <https://freevideolectures.com/university/iitm> | |

**CO-PO Mapping:** 3-High Mapping, 2-Moderate Mapping, 1-Low Mapping, - -Not Mapping

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| **CO1** | 3 | 3 | - | - | - | - | - | - | - | - | - | - |
| **CO2** | 3 | 3 | 3 | - | - | - | - | - | - | - | - | - |
| **CO3** | 3 | 3 | 3 | 3 | - | - | - | - | - | - | 3 | - |
| **CO4** | 3 | 3 | 3 | 3 | - | - | - | - | - | - | 3 | - |
| **CO5** | 3 | 3 | 3 | 3 | - | - | - | - | - | - | 3 | - |
| **CO6** | 3 | 3 | 3 | 3 | - | - | - | - | - | - | 3 | - |