RAYALASEEMA UNIVERSITY, KURNOOL

Common Framework of CBCS for Colleges in Andhra Pradesh w.e.f. 2017-2018
(Revised in April, 2016)

II YEAR IV SEMESTER
Paper-IV : DATA STRUCTURES

Course Objectives
To introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing efficient algorithms..

Course Outcomes
After completing this course satisfactorily, a student will be able to:
1. Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms
2. Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs.
3. Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs
4. Demonstrate different methods for traversing trees
5. Compare alternative implementations of data structures with respect to performance
6. Compare and contrast the benefits of dynamic and static data structures implementations
7. Describe the concept of recursion, give examples of its use, describe how it can be implemented using a stack .
8. Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing.

UNIT I
Concept of Abstract Data Types (ADTs)- Data Types, Data Structures, Storage Structures, and File Structures, Primitive and Non-primitive Data Structures, Linear and Non-linear Data Structures.
Linear Lists – ADT, Array and Linked representations, Pointers.
Arrays – ADT, Mappings, Representations, Sparse Matrices,   Sets – ADT, Operations
Linked Lists: Single Linked List, Double Linked List, Circular Linked List , applications

UNIT II
Stacks: Definition, ADT, Array and Linked representations, Implementations and Applications
Queues: Definition, ADT, Array and Linked representations, Circular Queues, Dequeues, Priority Queues, Implementations and Applications.

UNIT III
Trees: Binary Tree, Definition, Properties, ADT, Array and Linked representations, Implementations   and Applications. Binary Search Trees (BST) – Definition, ADT, Operations and Implementations, BST Applications. Threaded Binary Trees, Heap trees.

UNIT IV
Graphs – Graph and its Representation, Graph Traversals, Connected Components, Basic Searching Techniques, Minimal Spanning Trees
UNIT- V
Sorting and Searching: Selection, Insertion, Bubble, Merge, Quick, Heap sort, Sequential and Binary Searching.

REFERENCE BOOKS
1. D S Malik, Data Structures Using C++, Thomson, India Edition 2006.
2. Sahni S, Data Structures, Algorithms and Applications in C++, McGraw-Hill, 2002.
3. SamantaD, Classic Data Structures, Prentice-Hall of India, 2001.
4. Heilman G I,. Data Structures and Algorithms with Object-Oriented Programming, Tata McGraw-l lill. 2002. (Chapters I and 14).
5. Tremblay P, and Sorenson P G, Introduction to Data Structures with Applications, Tata McGraw-Hill,

Student activity:
> 1. **Create a visible stack using C-graphics**
> 2. **Create a visible Queue using C-graphics**

DATA STRUCTURES USING JAVA LAB

1. Write a Program to implement the Linked List operations

2. Write a Program to implement the Stack operations using an array.

3. Write Programs to implement the Queue operations using an array.

4. Write Programs to implement the Stack operations using a singly linked list.

5. Write Programs to implement the Queue operations using a singly linked list.

6. Write a program for arithmetic expression evaluation

7. Write a program to implement Double Ended Queue using a doubly linked list.

8. Write a program to search an item in a given list using Linear Search and Binary Search

9. Write a program for Quick Sort

10. Write a program for Merge Sort

11. Write a program on Binary Search Tree operations(insertion, deletion and traversals)

12. Write a program for Graph traversals