GUJARAT TECHNOLOGICAL UNIVERSITY

Bachelor of Engineering
Subject Code: 3130702
Semester – III
Subject Name: Data Structures

Type of course: Compulsory

Prerequisite: Computer Programming & utilization

Rationale: Data structure is a subject of primary importance in Information and Communication Technology. Organizing or structuring data is important for implementation of efficient algorithms and program development. Efficient problem solving needs the application of appropriate data structure during program development.

Understanding of data structures is essential and this facilitates the understanding of the language. The practice and assimilation of data structure techniques is essential for programming. The knowledge of „C" language and data structures will be reinforced by practical exercises during the course of study. The course will help students to develop the capability of selecting a particular data structure.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks				Total Marks
L	T	P	C	Theory Marks		Practical Marks		
				ESE (E)	PA (M)	ESE (V)	PA (I)	
3	0	4	5	70	30	30	20	150

Contents:

Sr. No.	Topics	Teaching Hrs.	Module Weightage
1	INTRODUCTION TO DATA STRUCTURE: Data Management concepts, Data types – primitive and non-primitive, Performance Analysis and Measurement (Time and space analysis of algorithms-Average, best and worst case analysis), Types of Data Structures- Linear & Non Linear Data Structures.	04	10
2	LINEAR DATA STRUCTURE Array: Representation of arrays, Applications of arrays, sparse matrix and its representation Stack: Stack-Definitions & Concepts, Operations On Stacks, Applications of Stacks, Polish Expression, Reverse Polish Expression And Their Compilation, Recursion, Tower of Hanoi Queue: Representation Of Queue, Operations On Queue, Circular Queue, Priority Queue, Array representation of Priority Queue, Double Ended Queue, Applications of Queue Linked List: Singly Linked List, Doubly Linked list, Circular linked list ,Linked implementation of Stack, Linked implementation of Queue, Applications of linked list.	13	30
3	NONLINEAR DATA STRUCTURE : Tree-Definitions and Concepts, Representation of binary tree, Binary tree traversal (Inorder, postorder, preorder), Threaded binary tree, Binary search trees, Conversion of	13	30

	General Trees To Binary Trees, Applications Of Trees- Some balanced tree mechanism, eg. AVL trees, 2-3 trees, Height Balanced, Weight Balance, Graph-Matrix Representation Of Graphs, Elementary Graph operations,(Breadth First Search, Depth First Search, Spanning Trees, Shortest path, Minimal spanning tree)			
4	HASHING AND FILE STRUCTURES : Hashing: The symbol table, Hashing Functions, Collision-Resolution Techniques, File Structure: Concepts of fields, records and files, Sequential, Indexed and Relative/Random File Organization, Indexing structure for index files, hashing for direct files, Multi-Key file organization and access methods.	06	15	
5	Sorting & Searching: Sorting – Bubble Sort, Selection Sort, Quick Sort, Merge Sort Searching – Sequential Search and Binary Search	06	15	

Reference Books:

1. An Introduction to Data Structures with Applications. by Jean-Paul Tremblay & Paul G. Sorenson Publisher-Tata McGraw Hill.
2. Data Structures using C & C++ -By Ten Baum Publisher – Prenctice-Hall International.
3. Fundamentals of Computer Algorithms by Horowitz, Sahni,Galgotia Pub. 2001 ed.
4. Fundamentals of Data Structures in C++-By Sartaj Sahani.
5. Data Structures: A Pseudo-code approach with C -By Gilberg & Forouzan Publisher-Thomson Learning.

Course Outcomes:

Sr. No.	CO Statement	Marks % weightage
CO-1	Define and classify various data structures, storage structures and common operations on them	10
CO-2	Create various linear data structures with their representation and perform different operations on them	20
CO-3	Create various nonlinear data structures with their representation and perform different operations on them	20
CO-4	Apply various searching sorting techniques on data set	30
CO-5	Solve the given a problem using an appropriate data structure to achieve optimal performance and compare its performance with other possible data structures	20

List of Practical:

At least 10 practical should be performed by students using programming language.

1. Introduction to pointers. Call by Value and Call by reference.
2. Introduction to Dynamic Memory Allocation. DMA functions malloc(), calloc(), free() etc.
3. Implement a program for stack that performs following operations using array.

(a) PUSH (b) POP (c) PEEP (d) CHANGE (e) DISPLAY

4. Implement a program to convert infix notation to postfix notation using stack.
5. Write a program to implement QUEUE using arrays that performs following operations (a) INSERT (b) DELETE (c) DISPLAY
5. Write a program to implement Circular Queue using arrays that performs following operations. (a) INSERT (b) DELETE (c) DISPLAY
6. Write a menu driven program to implement following operations on the singly linked list.
    (a) Insert a node at the front of the linked list.
    (b) Insert a node at the end of the linked list.
    (c) Insert a node such that linked list is in ascending order.(according to info. Field)
    (d) Delete a first node of the linked list.
    (e) Delete a node before specified position.
    (f) Delete a node after specified position.
7. Write a program to implement stack using linked list.
8. Write a program to implement queue using linked list.
9. Write a program to implement following operations on the doubly linked list.
    (a) Insert a node at the front of the linked list.
    (b) Insert a node at the end of the linked list.
    (c) Delete a last node of the linked list.
    (d) Delete a node before specified position.
10. Write a program to implement following operations on the circular linked list.
    (a) Insert a node at the end of the linked list.
    (b) Insert a node before specified position.
    (c) Delete a first node of the linked list.
    (d) Delete a node after specified position.
10. Write a program which create binary search tree.
11. Implement recursive and non-recursive tree traversing methods inorder, preorder and post-order traversal.
12. Write a program to implement Queue Sort
13. Write a program to implement Merge Sort
14. Write a program to implement Bubble Sort
15. Write a program to implement Binary Search.