

Course Name : Computer Engineering Group

Course Code : CO/CM/IF/CD

Semester : Fourth

Subject Title : Data Structure

Subject Code : 12111

Teaching and Examination Scheme:

Teaching Scheme			Examination Scheme					
TH	TU	PR	PAPER HRS	TH	PR	OR	TW	TOTAL
03	--	04	03	100	50#	--	25@	175

NOTE:

- **Two tests each of 25 marks to be conducted as per the schedule given by MSBTE.**
- **Total of tests marks for all theory subjects are to be converted out of 50 and to be entered in mark sheet under the head Sessional Work. (SW)**

Rationale:

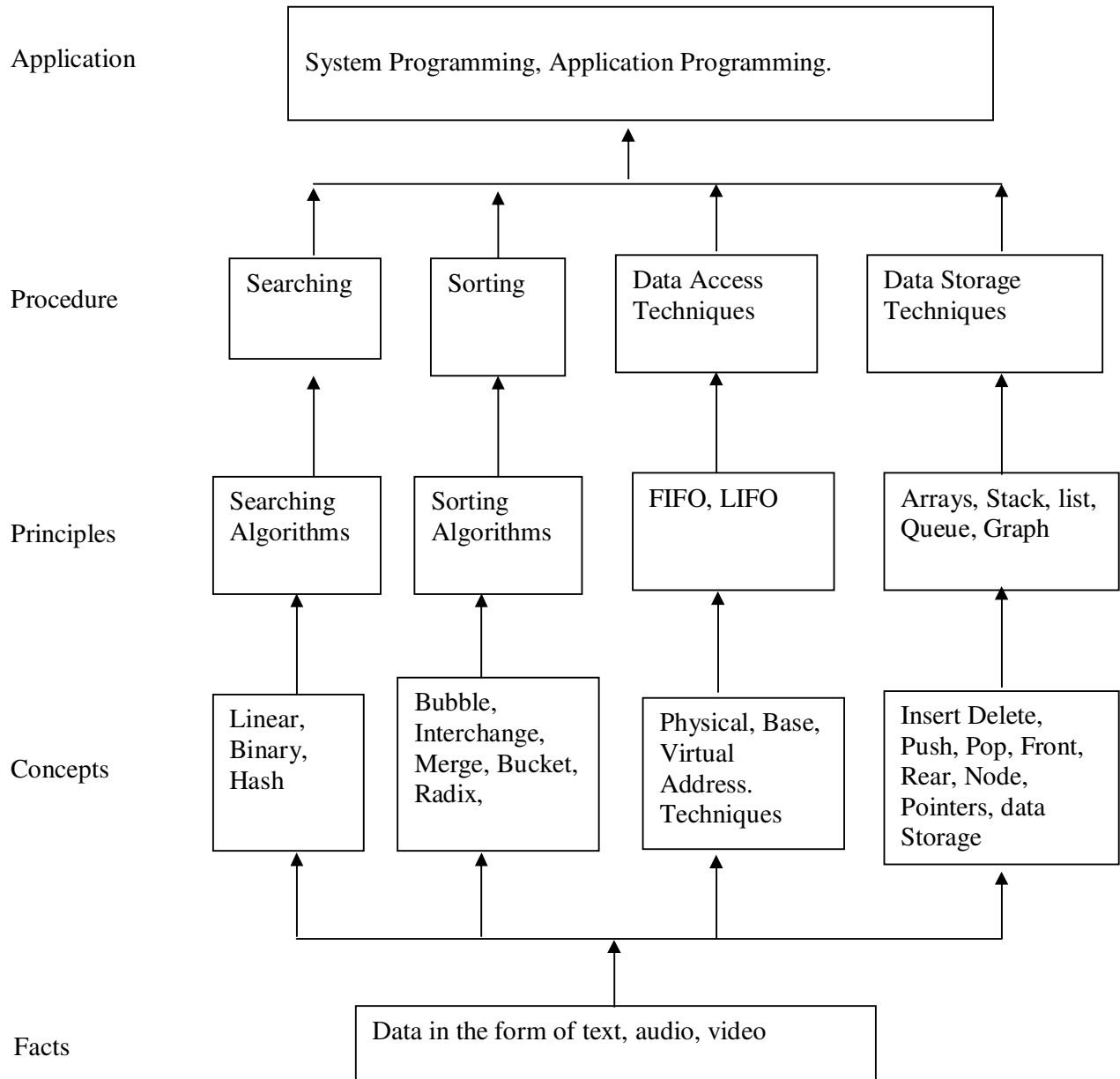
Data Structures is a subject of primary importance in Computer Science. Organizing or structuring data is important for implementation of efficient algorithms and program development. Efficient problem solving needs the application of appropriate data structures during program development.

Objectives:

Students will able to:

1. Understand different types of data structures.
2. Use the data structures stacks, queues, arrays, and lists in application programs.
3. Use the data structures trees, graphs in application programs.
4. Understand the applications of data structures in systems programming, database applications and programming languages.
5. Compare different implementations of data structures and to recognize the advantages and disadvantages of the different implementations.
6. Understand and apply sorting algorithms used in data structures such as bubble sort, insertion sort, selection sort, radix sort, merge sort and quick-sort.
7. Compare the efficiency of various sorting algorithms in terms of both time and space.

Learning Structure:



Scheme

Contents: Theory

Chapter	Name of the Topic	Hours	Marks
01	Introduction to data structure: 1.1 Data Representation <ul style="list-style-type: none"> • Abstract data Types • Data Structures (Linear and Non- Linear) • Atomic Type 1.2 Data Types <ul style="list-style-type: none"> • Primitive data type • Derived data type 1.3 Operations on data structures <ul style="list-style-type: none"> • Traversing, Inserting, Deleting • Searching and sorting 	02	08
02	Principles of programming and Analysis of Algorithms: 2.1 Algorithms <ul style="list-style-type: none"> • Different approaches for designing an algorithm • Complexity in terms of time and space • Big 'O' Notation 	02	06
03	Searching & Sorting: 3.1 Sorting <ul style="list-style-type: none"> • An Introduction • Efficiency of Sorting Algorithms 3.2 Sorting Techniques <ul style="list-style-type: none"> • Bubble Sort • Selection Sort • Insertion Sort • Merge Sort • Radix Sort (only algorithm) • Shell Sort (only algorithm) • Quick Sort (only algorithm) 3.3 Searching <ul style="list-style-type: none"> • An Introduction • Linear search • Binary Search 	08	16
04	Stacks: 4.1 Introduction to Stacks <ul style="list-style-type: none"> • Stacks as an Abstract Data Type • Primitive operations of stacks 4.2 Representation of Stacks through Arrays 4.3 Application of Stacks <ul style="list-style-type: none"> • Stack machines • Recursion • Arithmetic expression: Polish Notation 	06	12
05	Queues: 5.1 Introduction <ul style="list-style-type: none"> • Queue as an Abstract Data Type • Representation of Queues 5.2 Operations on queue : Searching ,Insertion, Deletion. 5.3 Types of queues	06	12

	<ul style="list-style-type: none"> • Circular Queues • Priority Queue • Dequeues 5.4 Application of Queues		
06	Linked List: 6.1 Introduction <ul style="list-style-type: none"> • Terminologies Node, Address, Pointer, Information, Next, Null pointer, Empty list etc. 6.2 Operations on list Searching, Insertion and Deletion 6.3 Types of lists <ul style="list-style-type: none"> • Linear list • Circular list • Doubly list 6.4 Array, stacks, queues, implementation using list.	06	12
07	Trees: 7.1 Introduction to Trees 7.2 Types of Trees <ul style="list-style-type: none"> • General tree • Binary tree • Height balanced • Weight balanced • Binary search tree 7.3 Operations on Binary Search Tree <ul style="list-style-type: none"> • Insertion of node • Deletion of node • Traversal—Inorder, Preorder and Postorder • Searching-- Depth-first search and Breadth-first search 	10	18
08	Graphs: 8.1 Introduction <ul style="list-style-type: none"> • Terminology graph, node (vertices), arcs (edge), directed graph, in-degree, out-degree, adjacent, successor, predecessor, relation, weight, path, length 8.2 Sequential Representation of Graphs 8.3 Linked Representation of Graphs 8.4 Traversal of Graphs <ul style="list-style-type: none"> • Depth-first search • Breadth-first search 8.5 Shortest Path algorithm for graph 8.6 Application of Graph	06	12
09	Hashing 9.1 Hash functions 9.2 Deleting items from hash tables	02	04
Total		48	100

Practical:

Skills to be developed:

Intellectual skills:

- Use of programming language constructs in program implementation.
- To be able to apply different logics to solve given problem.

- To be able to write program using different implementations for the same problem
- Study different types of errors as syntax semantic, fatal, linker & logical
- Debugging of programs
- Understanding different steps to develop program such as
 - Problem definition
 - Analysis
 - Design of logic
 - Coding
 - Testing
 - Maintenance (Modifications, error corrections, making changes etc.)
- Selection of appropriate data structures
- Selection of appropriate sorting and searching techniques

Motor skills:

- Proper handling of Computer System.

List of Practical:

Sr.No.	Practical
01	Programs based on: Array operations, insertion, deletion
02	Programs for implementing various sorting techniques. (Minimum three sorting techniques from topics mentioned in the syllabus))
03	Programs for implementing various sorting and searching techniques. (Minimum two searching techniques from topics mentioned in the syllabus.)
04	Programs based on Stacks Implementation of PUSH & POP operations, Evaluate postfix expressions, Infix to postfix conversions.
05	Recursive programs: factorial, fibonacci, Ackerman function, and tower of Hanoi. (Any two)
06	Programs for demonstrating queue operations (Minimum two).
07	Two programs based on Linked lists
08	Programs based on trees Creating a binary tree, in order, preorder and post order traversal of binary tree, deleting a node from binary tree.
09	Assignments based on graph theory.
10	Program based on hashing

Learning Resources:

Sr. No.	Author	Title	Publisher
01	ISRD Group New Dehli	Data Structure Using C	Tata McGraw Hill
02	Tremblie and Sorrenson	Data Structures	TMH Publications
03	Lafore	Teach Yourself data Structure and Algorithms in 24 Hrs.	BPB Publication
04	Tannenbaum	Data Structures using C++	PHI Publication
05	Seymour Lipschutz	Data Structures	Tata McGraw Hill

Books:

