C. Syllabus Detailing and Learning objectives

Module	Chapter	Detailed Content	Syllabus Detailing	Learning Objectives
Module 1	**CH 1** Introduction to Data Structures	Introduction, Types of Data Structures – Linear and Nonlinear, Operations on Data Structures, Concept of ADT, Arrays.	**Purpose**: To make students understand basics of Data Structures and their types also various operations on data structures. **Scope –** **1. Academic Aspects-** Understanding Data Structure ,their operations and types. **2. Technology Aspect-** Understand concept of Abstract Data Type and arrays. **3. Application Aspect-** Application of data structures in computer system and real world. **Students Evaluation** – 1. Theory Questions to be asked on various data structures and ADT 2. Case study to be given to differentiate between various data structures and their application.	1. To develop the understanding of fundamentals and technological aspects of Data Structure**(R)** 2.To understand and explain various operations on Data Structures**(U)** 3. To Distinguish Between Linear and Non Linear Data Structures **(A)** 4. To understand different types data structures and list them**(AN)** 5. To describe array and its various types . (R)

Module 2	**CH 2** Stack and Queues	Introduction, ADT of Stack, Operations on Stack, Array Implementation of Stack, Applications of Stack – Well form-ness of Parenthesis, Infix to Postfix Conversion and Postfix Evaluation, Recursion. ADT of Queue, Operations on Queue, Array Implementation of Queue, Circular Queue, Priority Queue, Double Ended Queue, Applications of Queue.	**Purpose** – This chapter gives detailed insight of Stack and Queue, implementation of Stack and Queue and their applications.. **Scope –** **1. Academic Aspects-** Learning the insights of stack and queue and their representation in computer system. **2. Technology Aspect-**Understand concept of stack and queue in their implementation using programming concepts. **3. Application Aspect-** Application of stack and queue in computer system to provide solution to various problems. **Students Evaluation** 1. Questions on stack and queue concepts can be asked. 2. Stack and Queues can be implemented using various programming languages. 3. Students can apply concepts of stack and queue to solve computer related problems.	1. To explain fundamentals of stack and queues along with its operations.**(R)** 2. To compare stack, queue in data structures. **(U)** 3. Constructing polish expressios from one form **(A)** 4. Differentiate between various types of queues **(AN)** 5. To select a data structures for particular problem statement **(E)**
Module 3	**CH 3** - Linked Lists	Introduction, Representation of Linked List, Linked List v/s Array,	**Purpose –** Student will study the linked list concepts, implementation of linked list also various types of linked lists.	1. To explain concept of linked list**.(U**) 2. To list various types of linked lists in data structure. **(R)**

		Implementation of Linked List, Linked Implementation of Stack and Queue, Circular Linked List, Doubly Linked List, Application – Polynomial Representation and Addition.	**Scope –** **1. Academic Aspects-** Learning basics of linked lists and its implementation in computer system. **2. Technology Aspect-** Understanding concept of linked list and their implementation using programming langauage **3. Application Aspect-** Application of linked lists in computer system and its representation.	3. To compare stack representation and queue representation of linked list. **(U)** 4. To describe various types of linked list . **(R)** 5. Constructing linked list for given polynomial **.(A)**
			Student Evaluation - 1. Theory Questions to be asked on linked list representation of stack and queue. 2. Lab experiments on linked list . 3. Corresponding viva questions can be asked for polynomial representation.	
			Student Evaluation - 1. This chapter is mainly theoretical, question on the cloud security fundamentals, data security, cloud host security is possible. 2. Students can be asked to use and configure the Amazon Cloud Storage Gateway. 3. Explanation of the encryption techniques used at various avenues.	

Module 4	CH 4 Trees (Hours -12)	Introduction, Tree Terminologies, Binary Tree, Representation, Types of Binary Tree, Binary Tree Traversals, Binary Search Tree, Implementation of Binary Search Tree, Applications – Expression Tree, Huffman Encoding. Search Trees – AVL, B Tree, B+ Tree, Splay Tree and Trie.	**Purpose**: To make students understand concept of tree data structure. Discuss different terminologies, types of tree data structure and the need and benefits of tree data structures. **Scope –** **1. Academic Aspects**- Understanding tree data structures, its various types and terminologies related to it. **2. Technology Aspect-** different ways and languages/tools for implementation of trees. **3. Application Aspect**- Application of trees like Huffman encoding and expression tree in practical world. **Students Evaluation –** 1. Theory Questions on search trees like AVL trees, B trees, B+ trees etc. 2. Lab experiment for Binary search tree implementation using c language. 3. Corresponding viva questions can be asked for applications of tree data structures and different types of trees.	1. To explain idea of tree data structures**.(U)** 2. To list terminologies related to tree and various types of tree data structure. **(R)** 3. To compare stack, queue, trees and linked list data structures. **(U)** 4. To describe tree traversing techniques. **(R)** 5. Constructing binary search tree, AVL tree, B tree, B+ trees, splay trees and tries for the given data**.(A)**
Module 5	CH 5 Graphs (Hours -6)	Introduction, Graph Terminologies, Representation, Graph Traversals – Depth First	**Purpose** – To make students understand concept of graph data structure. Discuss different terminologies, memory representation of graph and types of traversing techniques of graph data structure as well as the need and benefits of graph data structures.	1. To explain fundamentals of graph data structures along with its terminologies**.(R)** 2. To compare stack, queue, trees, linked

		Search (DFS) and Breadth First Search (BFS), Application – Topological Sorting	Scope – **1. Academic Aspects**- Understanding graph data structures and its importance. **2. Technology Aspect**- different ways and languages/tools for implementation of BFS and DFS. **3. Application Aspect**- Application of graph like topological sorting in practical world. **Students Evaluation –** 1. Theory Questions to be asked on memory representation of the graph and applications of graph. 2. Lab experiments on BFS and DFS traversing techniques. 3. Corresponding viva questions can be asked for applications of graphs, traversing techniques of graph and terminologies related to it.	list and graph data structures. (U) 3. Constructing breadth first search and depth first search for the given data.**(A)** 4. Differentiate between graph traversing techniques. (AN) 5. To select a data structures for implementation of graph traversing technique for any given problem. **(E)**
Module 6	Chapter 6 Sorting and Searching (Hours- 7)	Introduction, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort. Linear Search, Binary Search, Hashing – Concept, Hash Functions, Collision Handling Techniques.	**Purpose**: To make students understand the concepts of searching and sorting for any given data. Discuss different types of searching-sorting techniques along with their need and benefits. Also to introduce the concept of hashing, collision and its different types. Scope – **1. Academic Aspects-** Understanding different searching-sorting techniques and hashing technique. **2. Technology Aspect-** different ways and languages/tools for implementation of different searching-sorting techniques. **3. Application Aspect-** Applications of different searching-sorting techniques in practical world. **Students Evaluation –** 1. Theory Questions to be asked on any of the searching-sorting techniques as well as hashing technique for a given data. 2. Lab experiments on n-queen problem, sum of subset problem. 3. Corresponding viva questions can be asked for applications of backtracking.	1. To explain fundamentals of searching sorting and hashing concepts.**(R)** 2. To compare different sorting techniques as per their efficiency and time complexity. **(U)** 3. To compare different searching techniques as per their efficiency and time complexity. **(U)** 4. To select an efficient searching/sorting technique out of the learnt techniques for a given problem. **(E)** 5. To construct a hash table using hashing and collision resolution technique for any given data. **(A)**