Name of the college: Government College of Arts, Science and Commerce, Sanquelim - Goa

Name of Faculty: Bhakti Gawas	Subject: Computer Science	

Paper code: CSC-202 Data Structures and Algorithms	Program/Course: S.Y.B.Sc.	Division:

Academic year: 2025 - 2026	Semester: IV	Total Lectures: 45

Course Objectives:

1. To familiarize with basic data structures.
2. To develop the ability to choose the appropriate data structure for designing efficient algorithms.

Course Learning Outcome:

   On completion of the course, students will be able to:

CO1. Understand basic data structures, their implementation and some of their standard applications.
CO2. Analyze space-time complexity of basic algorithms.
CO3. Design and analyze basic algorithms using appropriate data structures.
CO4. Code, debug and analyze programs using suitable data structures.

Month	Lectures From:	To:	No. of lectures allotted	Topic, Subtopic to be covered	Learning outcome	ICT Tools	Reference books
December	01/12/2025	06/12/2025	3	**Unit 1:** Algorithm Basics – Algorithms and Data Structures, Pseudocode, Algorithm Features.	1. Define what an algorithm is and explain its importance in problem solving. 2. Identify different types of data structures 3. Explain how algorithms and data structures work together to process data efficiently. 4. Recognize real-life examples where algorithms and data structures are applied.	Laptop, LCD Projector, PowerPoint Presentation	Rod Stephens (2013). A Practical Approach to Computer Algorithms. Wiley.
	08/12/2025	13/12/2025	3	Arrays – Basic Concepts, One-dimensional Arrays, Two-dimensional Arrays, Higher-dimensional Arrays.	1. Understand the basic concept of arrays and how data is stored using index positions. 2. create and use one-dimensional and two-dimensional arrays. 3. Basic understanding of higher-dimensional arrays.		
	15/12/2025	23/12/2025	5	Linked Lists – Basic Concept, Singly Linked List, Doubly Linked List, Sorted Linked Lists	1. Understand the basic concept of linked lists and how nodes are connected dynamically in memory. 2. create and perform operations on singly and doubly linked lists such as insertion, deletion, and traversal.		
December Christmas Vacation	24/12/2025	01/01/2026		Christmas Vacation	Christmas Vacation		

Month	Start Date	End Date	Hours	Topics	Learning Outcomes	Resources	Textbook
January	02/01/2026	10/01/2026	3	Linked List Algorithms, Linked List Selection-sort, Multithreaded Linked List, Linked List with loops.	1. Understand and apply linked list algorithms to perform operations such as traversal, insertion, deletion, and searching. 2. Students will be able to implement selection sort on linked lists and analyse its working and efficiency. 3. Understand of multithreaded linked lists and linked lists with loops, including their structure.		
January	12/01/2026	17/01/2026	3	**Unit-2:** Stacks – Linked-List Stacks, Array Stacks	1. Understand the concept of stacks and the LIFO (Last In, First Out) principle. 2. Understand stack operations (push, pop, peek) using both arrays and linked lists. 3. Compare array-based and linked-list-based stacks.	Laptop, LCD Projector, PowerPoint Presentation	Donald F. Stanat and David F. McAllister, Discrete mathematics in Computer Science.
January	19/01/2026	24/01/2026	3	Double Stacks, Stack Algorithms-Applications of stacks.	1. Understand the concept of double stacks and how two stacks can be implemented efficiently within a single array. 2. Apply stack algorithms to perform standard operations and solve problems such as expression evaluation and conversion.		
January	26/01/2026	31/01/2026	2	Queues – Linked-List Queues	1. Understand the concept of queues and the FIFO (First In, First Out) principle. 2. Implement queue operations (enqueue, dequeue) using linked lists.		
February	02/02/2026	07/02/2026	3	Array Queues, Specialized Queues.	1. Understand the implementation of queues using arrays and perform basic operations such as enqueue and dequeue. 2. They will be able to differentiate and use specialized queues such as circular queues, priority queues, and dequeues.		

February	09/02/2026	14/02/2026	3	Sorting Algorithms, Searching Algorithms, Recursion.	1. Understand the working of basic sorting algorithms and compare them based on time and space complexity. 2. Understand searching algorithms. 3. Understand the concept of recursion.
February	16/02/2026	21/02/2026	3	**Unit-3:** Trees – Tree Terminology, Binary Search Tree, Tree Representations, Tree Traversal	1. Understand basic tree terminology and the hierarchical structure of trees. 2. Construct and use binary search trees (BSTs) and represent trees using suitable data structures. 3. Perform and compare tree traversal techniques such as inorder, preorder, and postorder.
February	23/02/2026	28/02/2026	3	Threaded Trees, Specialized Tree Algorithms, Balanced Trees.	1. Understand the concept of threaded trees. 2. Apply specialized tree algorithms for searching, insertion, and deletion. 3. Understand balanced trees, their importance.
March	02/03/2026	07/03/2026	3	Graphs – Graph Terminology, Representation, Traversals, Finding Paths.	1. Understand basic graph terminology and different ways of representing graphs. 2. Perform graph traversals such as BFS and DFS. 3. Learners will also apply algorithms to find paths between vertices in a graph.
March	09/03/2026	14/03/2026	3	Heaps	1. Understand the concept and properties of heaps as a specialized tree-based data structure. 2. Understand heap operations such as insertion, deletion. 3. Understand heaps in priority queues and heap sort.
March	16/03/2026	21/03/2026	3	Hashing	1. Understand the concept of hashing and the role of hash functions in fast data retrieval. 2. They will be able to implement hashing techniques and handle collisions using appropriate methods. 3. Analyse the efficiency and applications of hashing in databases and indexing systems.

Practical Work:

Week 1	Multi-dimensional arrays.
Week 2	Singly linked list.
Week 3	Doubly linked list, sorted list.
Week 4	multithreaded and circular linked list.
Week 5	Stack using arrays.
Week 6	Stacks using linked list and applications of stacks.
Week 7	Queues using arrays.
Week 8	Queues using Linked List.
Week 9	Sorting Algorithms.
Week 10	Searching algorithms.
Week 11	Recursion.
Week 12	Trees.
Week 13	Tree traversals.
Week 14	Heaps.
Week 15	graphs with traversals.