

A9503 – Data Structures

Teaching and Learning Scheme				Hours	Credits	Assessment Marks				
CI		LI	TW+SL	H	C	CIE	SEE	Total		
L	T	P	SL			45	90	3	40	60
45	0	0	45							

Course Description

Course Overview

This course introduces C++ programming with a focus on designing and implementing data structures. It covers object-oriented concepts such as classes, encapsulation, and abstraction for modular programming. Students learn to use pointers and dynamic memory for efficient data management. Templates and the Standard Template Library (STL) are taught for generic and reusable code. Linked lists are explored for dynamic data storage and sequential operations. Stacks and queues are implemented for practical applications like expression evaluation and task management. Tree structures are studied to represent hierarchical data with traversal and search methods. Binary search trees and AVL trees are covered to ensure organized and balanced data storage. Graph concepts and algorithms are introduced to model and navigate complex relationships. Hashing and dictionary structures provide fast data retrieval and efficient storage solutions. The course emphasizes hands-on programming to reinforce theoretical concepts. By the end, students can design, implement, and analyze core data structures effectively in C++.

Course Pre/Co-requisites

A9501 – Programming for Problem Solving

A9502 – Programming for Problem Solving Laboratory

Relevant Sustainable Development Goals (SDGs)

SDG 4: Quality Education

SDG 9: Industry, Innovation, and Infrastructure

Course Outcomes

After the completion of the course, the student will be able to:

- A9503.1. Make use of programming constructs, Templates and STL components to implement data structures and solve computational problems effectively.
- A9503.2. Implement linked list data structures using abstract data types to perform various operations.
- A9503.3. Implement stack and queue abstract data types for solving applications on linear data.
- A9503.4. Choose an appropriate nonlinear data structure for representing and solving real world problems.
- A9503.5. Examine hashing and dictionary structures for insertion, deletion, and searching in computational problems.

Course Syllabus

Unit-I:

Introduction to C++ and Data Structures: Structure of a C++ Program, Classes and objects, Dynamic memory allocation (new and delete), Constructors and destructors, Access specifiers (public, private), Encapsulation and abstraction, Pointers and references, Functions, Inline functions, and function overloading. Template Definition, Function Templates, class Templates, Generic Function, Template function Overloading. Standard Template Library (STL) Overview- Containers: vector, list, map, set. Iterators and algorithms, Using STL templates in practical applications.

Unit-II:

Linked Lists: Classification of Data Structures, Abstract Data Types, Introduction to Linked Lists, Applications of Linked Lists, Node Structure and Memory Allocation. Linked List ADT, Singly Linked List Operations- Insert, Delete, Search and Traverse. Doubly Linked List Operations – Insert, Delete, Search and Traverse and Circular Linked Lists.

Unit-III:

Stacks and Queues: Stack ADT – Concepts and Applications. Array and Linked List implementation of Stacks. Applications – Conversion of Expression from infix to postfix, Evaluation of Postfix Expression. Queue ADT – Concepts & Applications. Array and Linked List implementation of Linear Queues, Array Implementation of Circular Queue and Double Ended Queue.

Unit-IV:

Trees and Graphs: Tree ADT, Binary Tree Terminology and Properties, Binary Tree Traversals – Inorder, Preorder and Postorder. Binary Search Tree Operations – Insertion, Deletion and Searching. AVL Trees, Definition, Height of an AVL Tree, Operations – Insertion, Deletion and Searching. Graph Definitions and Terminology. Graph Representations – Adjacency Matrix and Adjacency List. Graph Traversals – Depth-First Search (DFS) and Breadth-First Search (BFS).

Unit-V:

Dictionaries and Hashing: Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing, linear probing, quadratic probing, double hashing, rehashing, extendible hashing. Dictionaries- linear list representation, skip list representation, operations - insertion, deletion and searching.

Books and Materials

Text Books:

1. Horowitz, Ellis, Sartaj Sahni, and Dinesh Mehta. *Fundamentals of Data Structures in C++*, 2nd ed., Universities Press, 2019.
2. Horowitz, *Data Structures Using C++*, 3rd edition, Course Technology, 2010.

Reference Books:

1. Drozdek, Adam. *Data Structures and Algorithms in C++*, 5th ed., Cengage Learning, 2025.
2. Dale, Nell, Chip Weems, and Tim Richards. *C++ Plus Data Structures*, 6th ed., Jones & Bartlett Learning, 2018.