

Title	Data Structures and Algorithms Laboratory
Code	CC-213L
Credit Hours	1 (3 hours Laboratory work per week)
Category	<ul style="list-style-type: none"> Computing Core
Aims and Objectives	<ul style="list-style-type: none"> To perceive computer programming as a basic and foremost computer science implementation and verification tool. To express and master the constructs of programming that are unavoidable for any computer science student. To introduce the concepts of Data Structure and Algorithms in a programming paradigm. <ul style="list-style-type: none"> Performance Analysis of Algorithms Abstract Data Types (ADTs) Arrays and Sparse Matrices Linked Lists Recursion Trees Graphs Searching and Sorting Hashing and Indexing Some Advanced concepts Like (Generalized List, Quad-Tree, B-Trees Tries etc.) <p>To be able to program in the C++ programming language using fundamental constructs e.g., data types, operators, control structures, functions, arrays, pointers, memory management, and text-based file I/O.</p>
Course Description	<p>Algorithm Specification: Properties of Algorithm, examples, performance, analysis, measurement, and Big Oh notation. Introduction to ADTs: Array and Polynomial as an ADT, Sparse Matrices, and Representation of Arrays. The Stack ADT: Linked list and array implementations, Expressions, Postfix Notation, and Infix to postfix conversion. The Queue ADT: Linked and array implementations of circular and double ended queue. Recursion: Recursive Definition and Processes, Writing Recursive Programs. Self-Referencing Classes and Dynamic Memory Allocation. Linked List: Singly Linked Lists, Circular Lists, Linked Stacks and Queues (Double Ended List), Doubly Linked Lists. Trees: Introduction to Trees, Logical construction and Traversing of Binary Trees, Implementation of Binary Trees (Insertion and Traversing), Searching and deletion in Binary Trees, Binary Search Tree, Introduction to Balanced and AVL Trees. Heaps: Heaps and Heaps as Priority Queues, Double Ended Priority Queue. Searching: Linear Search, Binary Search, and Types of Indexing. Hashing: Hash Functions: Division; Overflow Handling: Chaining; Introduction to advanced topics: B-Trees, Generalized List, etc. Sorting: Selection, Insertion, Merge, Quick, Bubble, Heap, Shell, Radix, and Bucket sorts. Graphs: Graph terminology, Adjacency List and Adjacency Matrix and Adjacency list representation of Graph; Elementary Graph Operations: Breadth First Search and Depth First Search, Spanning Trees (BFSST, DFSST), topological order, shortest path.</p>
Text Books	<p>A. Ellis Horowitz, Sartaj Sahni, and D. Mehta "Data Structures and Algorithm Analysis in C++", 4th Ed., Computer Science Press, 1995. ISBN 81-7808-792-8</p> <p>B. Adam B. Drozdek "Data Structure and Algorithm in C++" ISBN 0-534-37668-1</p> <p>C. D. Malhotra and N. Malhotra. "Data Structures and Program Design Using C++." ISBN: 978-1-68392- 370-1</p>
Reference Material	<p>R1. VARSHA H. PATIL. " Data Structures Using C++" ISBN-13: 978-0-19-806623-1 ,ISBN-10: 0-19-806623-6</p> <p>R2. D. Samanta. "Classic Data Structures", Prentice Hall, 2001</p> <p>R3. Mark Allen Weiss, "Data Structure and Algorithms in C++", 2nd Ed., Pearson Education, ISBN 81-7758-943-1</p> <p>R4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, "Introduction to Algorithms", 2nd Ed, MIT Press, 2001, ISBN 0-07-013151-1</p>

Week No.	Lab No.	Topic	Source Book-Chapter No. (Sections / Pages)	Recommendations for Learning Activities (Mention Assignments, Test, Quizzes, Practical, Case Study, Projects, Lab Work or Reading Assignments)
1	1	Algorithms <ul style="list-style-type: none"> Performance Analysis Measurement of Step Count Time and Space Complexity Big O notation 	LAB Manual – I A-(2.3,2.4), R3-(1.2), R1-(1.10), C- (1.6,1.8), B-(2.2 ~ 2.5)	Examples given in A – Ch.2 and R1-Ch.1 and C-Ch.2 Programming Challenges these chapters Exercises.
2	2	ADTs <ul style="list-style-type: none"> Array Polynomial Sparse Matrices N-Dimensional Arrays 	LAB Manual – II R1-Chapter No.2 R3-(3.2,3.2.1,3.2.7) R2-Chap No.2 R1-(2.11)	Examples given in R1 – Ch.2, R3-Ch.3 and R2-Ch.2 Programming Challenges in R1 – Ch.2, R3-Ch.3 and R2-Ch.2
3	3	Stack <ul style="list-style-type: none"> Bag and Stack ADTs Expression Evaluation Applications of Stack 	LAB Manual – III A-(3.6.1, 3.6.2) C-(7.1 ~ 7.5) A-(3.6.3), C- (7.6)	Examples given in A – Ch. 3 and C-Ch.7 Programming Challenges A – Ch. 3 and C-Ch.7
4	4	Queue <ul style="list-style-type: none"> Queue ADT Double Ended Queue Applications and Problem-Solving Using Queue 	LAB Manual – IV A-(3.7), C- Chap No.5 R1-(5.4) C -(5.5.1)	Examples given in A – Ch.3 and C-Ch.5 and R1-Ch.5 Programming Challenges in A – Ch.3 and C-Ch.5 and R1-Ch.5
5	5	Recursion <ul style="list-style-type: none"> Backtracking Application of Recursion 	LAB Manual – V B-Chap No.5 R1-(4.2,4.3,4.4), R1-(4.6)	Examples given in B – Ch. 5, R1-Ch.4 Programming Challenges in B – Ch. 5, R1-Ch.4
6	6	Linked List 01 <ul style="list-style-type: none"> Self-Referencing Classes Dynamic Memory Allocation Singly Linked Lists 	LAB Manual – VI B-(3.1), R1 -(6.11 ,6.12)	Examples given in B – Ch. 3 and R1-Ch.6 Programming Challenges in B – Ch. 3 and R1-Ch.6
7	7	Linked List 02 <ul style="list-style-type: none"> Circular Lists Linked Stacks and Queues Double Ended List 	LAB Manual – VII B-(3.2, 3.3), R1-(6.7,6.8)	Examples given in B – Ch. 3 and R1-Ch.6 Programming Challenges in B – Ch. 3 and R1-Ch.6
8	8	Linked List 03 <ul style="list-style-type: none"> Doubly Linked List Doubly Circular Linked List 	LAB Manual – VIII R1-(6.9, 6.13, 6.15) C-(4.5, 4.6), R1-(6.3.1)	Examples given in R1 – Ch. 6 and C-Ch.4 Programming Challenges in R1 – Ch. 6 and C-Ch.4

MID TERM EXAM				
9	9	Trees 01 <ul style="list-style-type: none"> Binary Trees Binary Search Trees(BST) 	LAB Manual – IX A-(4.1),R1-(7.1,7.2) A-(4.2, 4.6), R1-(7.3, 7.7), C-(8.3) A-(4.3), R1-(7.10)	Examples given in A – Ch.4 , R1-Ch.7 and C-Ch.8 Programming Challenges A – Ch.4 , R1-Ch.7 and C-Ch.8
10	10	Trees 02 <ul style="list-style-type: none"> Height Balanced Tree AVL Trees 	LAB Manual – X B-(6.7) A-(4.4) R1-(10.3.1,10.3.2) C-(8.5.1, 8.5.2)	Examples given in A – Ch.4 ,B-Ch.6, C-Ch.8 and R1-Ch.10 Programming Challenges in A – Ch.4 ,B-Ch.6, C-Ch.8 and R1-Ch.10
11	11	Trees 03 <ul style="list-style-type: none"> Heaps Heaps as Priority Queue Double Ended Priority Queue 	LAB Manual – XI A-Chap No.6 R3-Chap No.6	Examples given in A – Ch.6 and R3-Ch.6 Programming Challenges in A – Ch.6 and R3-Ch.6
12	12	Searching And Hashing <ul style="list-style-type: none"> Linear and Binary Search Indexing, Hash Functions and Collision Resolution Open Hashing Chaining 	LAB Manual – XII R1-(9.1,9.2) A-(5.1~5.5) R1-(11.3,11.5)	Examples given in A – Ch.5 , R1-Ch.9 and Ch.11 Programming Challenges in A – Ch.5 , R1-Ch.9 and Ch.11
13	13	Sorting <ul style="list-style-type: none"> Logical and Algorithmic implementation of Selection Sort, Insertion Sort, Merge Sort, Quick Sort, Bubble Sort, Heap Sort, Shell Sort, Radix Sort, Bucket Sort 	LAB Manual – XIII R1-(9.3) A-(7.1 ~ 7.6)	Examples given in A – Ch.7 ,R1-Ch.9 Programming Challenges in A – Ch.7 ,R1-Ch.9
14	14	Graphs 01 <ul style="list-style-type: none"> Graph Terminology Adjacency List Representation Adjacency Matrix Representation 	LAB Manual – XIV R1-(8.1, 8.2, 8.3.1) R1-(8.3.2,8.3.2,8.3.4) C-(12.3.2)	Examples given in R1-Ch.7 and Ch.8, C-Ch.12 Programming Challenges in R1-Ch.7 and Ch.8, C-Ch.12
15	15	Graphs 02 <ul style="list-style-type: none"> Elementary Graph Operations Breadth-First Search (BFS), Depth-First Search (DFS) Spanning Tree (BFSST, DFSST) Topological Sort, Shortest Path 	LAB Manual – XV R1-(8.4) C-(12.4) B-(8.5), R1-(8.5) C-(12.6) R1-(8.6),A-(9.3)	Examples given in R1 – Ch.8, C-Ch.12, B-Ch.8 , A-Ch.9 Programming Challenges in R1 – Ch.8, C-Ch.12, B-Ch.8 , A-Ch.9
16	16	Advanced Topics <ul style="list-style-type: none"> Generalized List Quad Tree Tries B-Tree 	LAB Manual - XVI R1-(13.3) R1-(6.13) C-Chap No.9	Examples given in R1- Ch.13 and Ch.6(6.13) Programming Challenges in R1- Ch.13 and Ch.6
FINAL TERM EXAM				