

Course Handout

Institute/School Name	Chitkara University Institute of Engineering and Technology		
Department Name	Department of Computer Science & Engineering		
Programme Name	Bachelor of Engineering (B.E.), Computer Science & Engineering		
Course Name	Data Structures using Object Oriented Programming-II	Session	Jan – June 2026
Course Code	24CSE0212	Semester/Batch	4 th /2024
L-T-P (Per Week)	2-0-6	Course Credits	5
Pre-requisite	Data Structures using Object Oriented Programming-I	NHEQF Level ¹	5
Course Coordinator		SDG Number ⁴	4, 8, 9

1. Objectives of the Course

The scope of the course is to provide the foundation for understanding the key aspects of Java programming and implementation obtaining a theoretical understanding of advanced programming concepts. The objectives of the course are:

- To understand and implement recursion and divide & conquer techniques
- To develop problem-solving skills using data structures
- To apply object-oriented programming concepts
- To implement and analyze linear and non-linear data structures
- To gain hands-on experience with real-world programming problems

2. Course Learning Outcomes (CLOs)

Student should be able to:

	CLOs	Program Outcomes (PO)	NHEQF Level Descriptor ²	No. of Lectures
CLO01	To understand recursion and divide-and-conquer techniques, enabling them to design efficient solutions for complex problems.	PO1, PO2, PO4	Q1, Q2, Q6	15
CLO02	Apply object-oriented programming constructs such as classes, objects, exception handling, polymorphism and inheritance.	PO1, PO8, PO9	Q3, Q5	28
CLO03	Design and implement core linear data structures including stacks, queues and linked lists, and solve standard problems.	PO2, PO3, PO5	Q2, Q4, Q6	28
CLO04	Construct, traverse, and analyze non-linear data structures such as binary trees, binary search trees, AVL trees, and heaps to perform efficient searching, sorting, and hierarchical data processing.	PO2, PO4	Q2, Q3, Q4, Q6	15
CLO05	Ability to work with Interfaces, Abstraction & Generic.	PO1, PO3, PO5	Q1, Q5	14
CLO06 (Only for lab components)	Illustrate the concept of stack, queue, linked list and tree data structures.	PO7, PO8, PO10, PO11	Q6	20
Total Contact Hours				120

CLO-PO Mapping

CLO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	Type of Assessment's ³
CLO01	H	M		L								Formative, Summative
CLO02	M							L	L			Formative, Summative
CLO03		L	M		M							Formative, Summative
CLO04		H		M								Formative, Summative
CLO05	M		M		M					M	L	Formative, Summative
CLO06												Formative, Summative

H=High, M=Medium, L=Low

¹ National Higher Education Qualification Framework Level, Refer to annexure

² NHEQF Level Descriptor, Refer to Annexure & [Learning outcomes descriptors for qualification for all levels on the NHEQF](#)

³Types of Assessments can be referred from Type of Assessments. Refer to Annexure.

⁴For SDG Mapping with Courses, Pl refer [SDG Mapping policy for Courses](#)

3. Recommended Books:

- B01:** Herbert Schildt, Java A Beginner's Guide Ninth Edition, Mc Graw Hill, 2022.
- B02:** Horstmann, C. S., Core Java for the impatient. Pearson Education, 2023.
- B03:** Data Structures and Algorithms in Java, Robert Lafore, Sams Publishing, 2nd edition, 2002.
- B04:** Data structures and algorithms in Java. John, Goodrich MT, Tamassia R, Goldwasser MH, wiley 2014.
- B05:** Introduction to Algorithms by Thomas H. Cormen, The MIT Pressman 3rd Edition, 2001

4. Other readings and relevant websites:

SerialNo	Link of Journals, Magazines, websites and Research Papers
1.	https://nptel.ac.in/courses/106105225
2.	https://cse.iitkgp.ac.in/~dsamanta/javads/index.htm
3.	https://algs4.cs.princeton.edu/home/
4.	https://library.chitkara.edu.in/subscribed-books.php
5.	http://www.nptelvideos.com/java

5. Recommended Tools and Platforms

- Visual Studio IDE
- Tespad

6. Course Plan: Theory+ Lab Plan**Theory Plan**

Lect. No.	Topic(s)
1	Recursion – Fundamental Concept, recurrence relations, call stack, factorial, power
2	Advanced Recursion – I: Fibonacci, subsequences, coin toss, board path
3	Advanced Recursion – II: Permutations, Tower of Hanoi, lexicographical counting
4	Divide & Conquer: Merge sort, quick sort
5	Bit Masking – Basics: AND, OR, XOR, NOT, applications, unique number problems, Power Set
6-7	OOP – Basics: Classes, objects, constructors, Classes & Objects: Access modifiers, this keyword, static keyword, heap vs stack
8-9	Exception Handling: Exception hierarchy, try-catch, multiple catch, nested try, throw, throws, finally, custom exceptions
10	Inheritance: Basics, types of inheritance, super keyword, method overriding
11	Polymorphism: Compile-time and runtime, dynamic method dispatch, final keyword
12	Object Class & Memory Model: toString, equals, hashCode, garbage collection
13-14	Stack & Queue – I: Stack and queue implementation, queue operations, deques, circular queues
15	Stack & Queue Problems: Stock span, celebrity problem, next greater element
ST1	
16-17	Monotonic Stack and Queue: Question practices, Sliding Window
18	Linked List – I: Singly linked list, arrays vs linked list, implementation, basic linked list problems
19	Linked List – II: Reverse, middle, kth from end, cycle detection & removal, intersection
20-21	Linked List – III: Doubly linked list, circular linked list, applications, sorting linked list
22-23	Binary Trees – I: Implementation, tree traversals (preorder, inorder, postorder, level order), height, max, find
24-25	Binary Trees – II: Diameter, optimized diameter, max sum, balanced tree, mirror, ancestors
26	Binary Search Trees: Insertion, deletion, range queries
ST2	
27	AVL and Complete Binary Trees: Concepts and question practice
28	Interfaces, Abstraction & Generics: Interface, Comparator, Comparable, generic classes
29-30	Heap & Priority Queues – Fundamentals: Min heap, max heap, properties, applications (merge k sorted lists, running median)
End Term Exam	

Lab Plan

Lab No.	Topic(s)
1-4	Implement simple problems of recursion like factorial, Fibonacci
5-8	Implement complex problems like coin toss, board path using recursion.
9-14	Write a program to implement permutations, lexicographical counting, tower of Hanoi using recursion.
15-20	Implement quick and merge sort with divide and conquer strategy.
21-24	Write a program in java to demonstrate various concepts of object-oriented programming.
25-26	Write a program to show the working of static member functions.
27-30	Implement Exception handling using various examples.
31-34	Implement the concept of inheritance by creating class hierarchies.
35-38	Implement the concept of polymorphism by demonstrating compile-time and runtime polymorphism.
39-42	implement the methods of the Object class—toString(), equals(), and hashCode() with respect to object creation, memory allocation, and garbage collection.
43-44	Create a stack and perform Pop, Push, Traverse operations on the stack
45-48	Implement various operations on queue, deque and circular queue.
49-58	Implement in java using stack and queue- Stock span, celebrity problem, next greater element

59-62	Write a program to insert and delete node from the linked list.
63-66	Implement Linked list various operation reverse, middle, kth from end, cycle detection in Java.
67-70	Implement various operations on doubly and circular linked list.
71-74	Write programs in java for binary tree traversal.
75-78	Write a program to calculate height, find maximum element in a binary tree.
79-85	Implement tree traversals (preorder, inorder, postorder, level order) using linked list.
86-90	Implement a Binary Search Tree to perform insertion and deletion of nodes, and to process range queries to retrieve all elements within a specified range.

7. Delivery/Instructional Resources Theory

Plan:

Lect . No.	Topics	CLO	Book No, CH No, Page No	TLM ³	ALM ⁴	Web References	Audio-Video
1	Recursion – Fundamental Concept, recurrence relations, call stack, factorial, power	1	B02, CH 6	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://javascript.info/recursion	https://www.youtube.com/watch?v=4tMG_sBKoIA
2	Advanced Recursion – I: Fibonacci, subsequences, coin toss, board path	1	B02, CH 6	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://arxiv.org/html/2501_07463v1 https://www.geeksforgeeks.org/dsa/tail-recursion-fibonacci/	https://www.youtube.com/watch?v=k4y5Pr0YVhg
3	Advanced Recursion – II: Permutations, Tower of Hanoi, lexicographical counting	1	B02, CH 6	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.youtube.com/watch?v=rf6uf3jNjbo	https://www.geeksforgeeks.org/dsa/c-program-for-tower-of-hanoi/
4	Divide & Conquer: Merge sort, quick sort	1	B02, CH 7	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://workat.tech/problem-solving/tutorial/sorting-algorithms-quick-sort-merge-sort-dsa-tutorials-gj3h98lk6j2w https://www.geeksforgeeks.org/quick-sort-algorithm/	https://www.youtube.com/watch?v=tWCaFVJM_Ui8 https://www.youtube.com/watch?v=tn9hxD8gx2M
5	Bit Masking – Basics: AND, OR, XOR, NOT, applications, unique number problems, Power Set	1	B01, CH 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeeks.org/competitive-programming/bit-manipulation-for-competitive-programming/	https://www.youtube.com/watch?v=LqKaUv1G3_I
6-7	OOP – Basics: Classes, objects, constructors, Classes & Objects: Access modifiers, this keyword, static keyword, heap vs stack	2	B01, CH 6	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeeks.org/java/classes-objects-java/	https://www.youtube.com/watch?v=vqZMgufPwzE
8-9	Exception Handling: Exception hierarchy, try-catch, multiple catch, nested try, throw, throws, finally, custom exceptions	2	B01, CH 10	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeeks.org/java/exceptions-in-java/	https://www.youtube.com/watch?v=ggdv1RhcK_4
10-11	Inheritance: Basics, types of inheritance, super keyword, method overriding	2	B01, CH 8	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeeks.org/java/inheritance-in-java/ https://www.geeksforgeeks.org/java/overriding-in-java/	https://www.youtube.com/watch?v=hxDRaNBRM
12-13	Polymorphism: Compile-time and runtime, dynamic method dispatch, final keyword	2	B01, CH 8	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeeks.org/java/dynamic-method-dispatch-runtime-polymorphism-java/	https://www.youtube.com/watch?v=TRlz0X5hD7E

³ Teaching Learning Methods, Refer to Annexure

⁴ Active Learning Methods

				tool			
14	Object Class & Memory Model: <code>toString</code> , <code>equals</code> , <code>hashCode</code> , <code>garbage collection</code>	3	B01, CH 16	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/java/object-class-in-java/	https://www.youtube.com/watch?v=XxMxCYMygx8
15-16	Stack & Queue – I: Stack and queue implementation, queue operations, deques, circular queues	1, 3	B02, CH 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/deque-set-1-introduction-applications/	https://www.youtube.com/watch?v=tqQ5fTamiN4
17-18	Stack & Queue Problems: Stock span, celebrity problem, next greater element	1, 3	B02, CH 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/top-50-problems-on-stack-data-structure-asked-in-interviews/	https://www.youtube.com/watch?v=eayzoSRkVc
19	Monotonic Stack and Queue: Question practices, Sliding Window	1, 3	B02, CH 4	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/introduction-to-monotonic-queues/	https://www.youtube.com/watch?v=DfljaUwZsk
20	Linked List – I: Singly linked list, arrays vs linked list, implementation, basic linked list problems	2, 3	B02, CH 5	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/linked-list-vs-array/	https://www.youtube.com/watch?v=HKfj0l7ndbc
21	Linked List – II: Reverse, middle, kth from end, cycle detection & removal, intersection	2, 3	B02, CH 5	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/remove-every-k-th-node-linked-list/	https://www.youtube.com/watch?v=qEJrlcc-hAY
22	Linked List – III: Doubly linked list, circular linked list, applications, sorting linked list	2, 3	B02, CH 5	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.w3schools.com/dsa/data_linkedlists_types.php	https://www.youtube.com/watch?v=7heYkXsi5BU
23-24	Binary Trees – I: Implementation, tree traversals (preorder, inorder, postorder, level order), height, max, find	2, 4	B02, CH 8	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/tree-traversals-inorder-preorder-and-postorder/	https://www.youtube.com/watch?v=eapz1rvKDCM
25	Binary Trees – II: Diameter, optimized diameter, max sum, balanced tree, mirror, ancestors	2, 4	B02, CH 8	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/diameter-of-a-binary-tree/	https://www.youtube.com/watch?v=p3YUIEZr2vM
26	Binary Search Trees: Insertion, deletion, range queries	2, 4	B02, CH 8	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/binary-search-tree-data-structure/	https://www.youtube.com/watch?v=4s1Tcvm00pA
27	AVL and Complete Binary Trees: Concepts and question practice	2, 4	B02, CH 9	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/dsa/introduction-to-avl-tree/	https://www.youtube.com/watch?v=CVA85JuJen0
28	Interfaces, Abstraction & Generics: Interface, Comparator, Comparable, generic classes	5	B01, CH 9, B01, CH 14	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeek.org/java/generic-constructors-and-interfaces-in-java/	https://www.youtube.com/watch?v=PHmeJoxYpg

29-30	Heap & Priority Queues — Fundamentals: Min heap, max heap, properties, applications (merge k sorted lists, running median)	2, 3, 4	B02, CH 12	Lecture Discussion Demonstration method using a simulation or a tool	Quiz Lab Challenge Assignment	https://www.geeksforgeeks.org/dsa/heap-data-structure/	https://www.youtube.com/watch?v=Qf-TDPr0nYw
-------	---	---------	------------	--	-------------------------------	---	---

Lab Plan:

Lab No.	Experiment	CLO	TLM	ALM	Web References	Audio-Video
1-4	Implement simple problems of recursion like factorial, Fibonacci	1,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=4tMG_sBKoIA
5-8	Implement complex problems like coin toss, board path using recursion.	1,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=k4y5Pr0YVhg
9-14	Write a program to implement permutations, lexicographical counting, tower of Hanoi using recursion.	1,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.geeksforgeeks.org/dsa/c-program-for-tower-of-hanoi/
15-20	Implement quick and merge sort with divide and conquer strategy.	1,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=tWCaFVJMUi8 https://www.youtube.com/watch?v=tn9hxD8gx2M
21-24	Write a program in java to demonstrate various concepts of object-oriented programming.	2,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=vqZMgufPwzE
25-26	Write a program to show the working of static member functions.	2,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=vqZMgufPwzE
27-30	Implement Exception handling using various examples.	2,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=ggdv1RhCK_4
31-36	Implement the concept of inheritance by creating class hierarchies.	2,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=hxDranBRM
37-40	Implement the concept of polymorphism by demonstrating compile-time and runtime polymorphism.	2,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=TRlZ0X5hD7E
41-44	Implement the methods of the Object class— <code>toString()</code> , <code>equals()</code> , and <code>hashCode()</code> with respect to object creation, memory allocation, and garbage collection.	2,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=XxMxCYMqx8
45-52	Create a stack and perform Pop, Push, Traverse operations on the stack	2,3	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=tqO5fTamiN4
53-58	Implement various operations on queue, deque and circular queue.	2,3,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=DfljaUwZsOk
59-62	Implement in java using stack and queue- Stock span, celebrity problem, next greater element	2,3,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=eayzoSRkVc
63-64	Write a program to insert and delete node from the linked list.	2,3,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=HKfj0l7ndbc
65-66	Implement Linked list various operation reverse, middle, kth from end, cycle detection in Java.	2,3,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=qEJrlcc-hAY
67-70	Implement various operations on doubly and circular linked list.	2,3,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=7heYkXsi5BU
71-72	Write programs in java for binary tree traversal.	2,4,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=eaxz1rvKDCM
73-74	Write a program to calculate height, find maximum element in a binary tree.	2,4,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=p3YUIEZr2vM
75-78	Implement tree traversals (preorder, inorder, postorder, level order)	2,4,6	Demonstration method using a	Lab Challenge	NA	https://www.youtube.com/watch?v=p3YUIEZr2vM

	using linked list.		simulation or a tool			
79-82	Implement a Binary Search Tree to perform insertion and deletion of nodes, and to process range queries to retrieve all elements within a specified range.	2,4,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=4s1Tcvm00pA
83-86	Implement an AVL Tree by performing insertion operations while maintaining height balance using appropriate rotations.	2,4,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=CVA85JuJEn0
87-88	Implement the concepts of interfaces, abstraction, Comparable and Comparator interfaces, and generic classes in Java	2,3,4,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=PHmeJoxY_pg
89-90	Implement the Heap data structure and perform fundamental operations such as insertion, deletion, and heapify operations	2,3,4,6	Demonstration method using a simulation or a tool	Lab Challenge	NA	https://www.youtube.com/watch?v=Of-TDPr0nYw

8. Remedial Classes⁵

After every Sessional Test, identify weak learners, and prepare the student lists. The following methods would be used to improve the performance of students.

- Remedial Classes on Saturdays
- Offer supplementary materials or activities to reinforce concepts outside regular class hours.
- Special discussions would be planned and scheduled accordingly.

9. Self-Learning⁶

Assignments to promote self-learning, survey of contents from multiple sources.

S.No	Topics	CLO	ALM	References/MOOCS
1	Creating a new file and read the content of file	3	Think – Pair- Share	https://www.geeksforgeeks.org/c/basics-file-handling-c/
2	Operations on Circular linked List and its applications	6	Think – Pair- Share	https://archive.nptel.ac.in/courses/106/102/106102064/
3	Dynamic programming to find the nth Fibonacci Number	5,6	Think – Pair- Share	https://www.w3schools.com/dsa/dsa_ref_dynamic_programming.php

10. Delivery Details of Content Beyond Syllabus⁷

Content beyond syllabus covered (if any) should be delivered to all students that would be planned, and schedule notified accordingly.

S.No	Advanced Topics, Additional Reading, Research papers and any	CLO	POs	ALM	References/MOOCS
1	Dijkstra algorithm	7	1, 2, 3, 4, 5, 10	Brainstorming session	https://www.w3schools.com/dsa/dsa_algo_graphs_dijkstra.php https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/

11. Evaluation Scheme & Components:

Assessment Type ⁸	Evaluation Component ⁹	Type of Component ¹⁰	No. of Assessments ¹¹	% Weightage of Component	Max. Marks	Mode of Assessment	CLO
Formative	Component1	Formative Assessments	--	10%	10	--	1-5
Summative	Component2	Sessional Tests(STs)	02*	40%	40	Computer based online in Campus	1,2,3
Summative	Component3	End Term Examination	01**	50%	50	Computer based online in Campus	1-6
Total			100%				

⁵ Refer to Annexure

⁶ Refer to Annexure

⁷ Refer to Annexure

⁸ Refer to [Annexure 2 of NCrF](#)

⁹ Refer to Annexure

¹⁰ Refer to Annexure

¹¹ Refer to Annexure

*Students will have to appear in all Sessional Tests.

**As per Academic Guidelines, a minimum of 75% attendance is required to become eligible for appearing in the End Semester Examination.

12. Syllabus of the Course:

Subject:			
S.No.	Topic(s)	No. of Lectures	Weightage %
1	Recursion – Fundamental Concept, recurrence relations, call stack, factorial, power, Advanced Recursion – I: Fibonacci, subsequences, coin toss, board path, Advanced Recursion – II: Permutations, Tower of Hanoi, lexicographical counting, Divide & Conquer: Merge sort, quick sort, Bit Masking – Basics: AND, OR, XOR, NOT, applications, unique number problems, Power Set, OOP – Basics: Classes, objects, constructors, Classes & Objects: Access modifiers, this keyword, static keyword, heap vs stack, Exception Handling: Exception hierarchy, try-catch, multiple catch, nested try, throw, throws, finally, custom exceptions, Inheritance: Basics, types of inheritance, super keyword, method overriding, Polymorphism: Compile-time and runtime, dynamic method dispatch, final keyword, Object Class & Memory Model: toString, equals, hashCode, garbage collection, Stack & Queue – I: Stack and queue implementation, queue operations, dequeues, circular queues, Stack & Queue Problems: Stock span, celebrity problem, next greater element	62	50%
ST-1			
2	Monotonic Stack and Queue: Question practices, Sliding Window, Linked List – I: Singly linked list, arrays vs linked list, implementation, basic linked list problems, Linked List – II: Reverse, middle, kth from end, cycle detection & removal, intersection, Linked List – III: Doubly linked list, circular linked list, applications, sorting linked list, Binary Trees – I: Implementation, tree traversals (preorder, inorder, postorder, level order), height, max, find, Binary Trees – II: Diameter, optimized diameter, max sum, balanced tree, mirror, ancestors, Binary Search Trees: Insertion, deletion, range queries.	110	92%
ST-2 (ST1 Syllabus also included)			
3	AVL and Complete Binary Trees: Concepts and question practice, Interfaces, Abstraction & Generics: Interface, Comparator, Comparable, generic classes, Heap & Priority Queues – Fundamentals: Min heap, max heap, properties, applications (merge k sorted lists, running median)	120	100%
End Term Exam			

13. Academic Integrity Policy:

Education at Chitkara University builds on the principle that excellence requires freedom where Honesty and integrity are its prerequisites. Academic honesty in the advancement of knowledge requires that all students and Faculty respect the integrity of one another's work and recognize the importance of acknowledging and safeguarding intellectual property. Any breach of the same will be tantamount to severe academic penalties.

This Document is approved by:

Designation	Name	Signature
Course Coordinator	Dr. Shikha	
Head-Academic Delivery	Dr. Mrinal Paliwal	
Dean	Dr. Rishu Chhabra	
Date (DD/MM/YYYY)	17.1.2026	