Course Information Sheet

Programme: UG	**Degree:** B. Tech (Common for all Branches)	
Course Code:	**Course Title: DATA STRUCTURES**	
Year: I **Sem:** I **A.Y. :2025-26**	**Regulation: IS23** **College: ISTS (an Autonomous Institution)**	
L T/P/D C: 3/0/0/3	**Credits: 3**	**Contact Hrs: 5**
Mid Marks: 30	**External Marks: 70**	**Total Marks: 100**
Teaching Hrs:	**Exam Duration:** 3 hrs.	

Course Information:

Course Code	Course Name	Description	Year-Sem
	Data Structures	❖ This course introduces fundamental data structures such as arrays, linked lists, stacks, queues, trees, graphs and hashing. ❖ It emphasizes storage structures, memory allocation, searching, sorting, and algorithm efficiency. Students learn to choose appropriate data structures for solving computational problems effectively	I B.Tech (Common for Group-B Branches) Semester -1

Course outcomes:

Student will able to

SUBJECT: ENGINEERING PHYSICS		
CO.NO	Course Outcomes	BT Level
After successful completion of this course students will be able to:		
C112.1	CO1: Explain linear data structures and compare searching & sorting algorithms.	Understanding
C112.2	CO2: Design and implement linked lists for dynamic memory usage	Applying
C112.3	CO3: Develop stack-based programs for recursion and expression evaluation.	Applying
C112.4	CO4: Apply queue algorithms for scheduling and BFS traversal	Applying
C112.5	CO5: Apply tree operations to solve computational problems.	Analyzing
C112.6	CO6: Identify hashing techniques and implement hashing solutions.	Analyzing

Course Articulation Matrix:

Mapping of Course Outcomes (CO) with Program Outcomes (PO) and Program Specific Outcomes (PSO's):

Course Outcomes (CO)	Program Outcomes (PO)												Program Specific Outcomes (PSO's)	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO1	PSO2
C112.1	3	3	2	--	2	_	_	_	_	_	_	2	_	_
C112.2	3	2	2	_	_	_	_	_	_	_	_	2	_	_
C112.3	3	3	2	_	_	_	_	_	_	_	_	2	_	_
C112.4	3	2	2	_	_	_	_	_	_	_	_	2	_	_
C112.5	3	2	2	_	_	_	_	_	_	_	_	2	_	_
C112.6	3	2	2	--	--	--	--	--	--	--	--	2	--	--
	3	**2.4**	**2**	--	**2**							**2**		

Level: 3–Strongly linked | 2–Moderately linked | 1–Weakly linked

JUSTIFICATIONS OF CO –PO MAPPING

C121.1		
C112.1-P01	3	Understanding linear data structures improves analytical ability in engineering problem-solving.
C112.1-P02	3	Searching and sorting analysis strengthens algorithmic reasoning.
C112.1-P03	2	Application of structures supports software implementation skills.
C112.1-P04	--	
C112.1-P05	--	
C112.1-P06	--	
C112.1-P07	--	
C112.1-P08	--	
C112.1-P09	--	
C112.1-P010	--	
C112.1-P011	--	.
C112.1-P012	2	Learning DS techniques enables continuous skill development

C112.2		
C112.2-P01	3	Understanding linked list memory structure builds engineering logic.
C112.2-P02	2	Designing linked lists improves analytical comparison skills.
C112.2-P03	2	Implementation strengthens software design abilities.
C112.2-P04	--	
C112.2-P05	--	
C112.2-P06	--	
C112.2-P07	--	
C112.2-P08	--	
C112.2-P09	--	
C112.2-P010	--	
C112.2-P011	--	
C112.2-P012	**2**	Dynamic memory concepts support lifelong learning.

C112.3		
C112.3-P01	3	Stack concepts reflect engineering fundamentals.
C112.3-P02	3	Analysis of recursion improves logic.
C112.3-P03	2	Implementing stack operations enhances coding
C112.3-P04	--	
C112.3-P05	--	
C112.3-P06	--	
C112.3-P07	--	
C112.3-P08	--	
C112.3-P09	--	
C112.3-P010	--	
C112.3-P011	--	
C112.3-P012	2	Deepens interest in advanced algorithms.

C112.4		
C112.4-P01	3	Understanding queue structure supports foundational concepts necessary for real-time systems and scheduling processes.
C112.4-P02	2	Analyzing BFS algorithms strengthens students' ability to solve problems related to graph traversal and shortest paths.
C112.4-P03	2	Implementing various queue types (simple, circular, priority queues) enhances engineering design and development skills.
C112.4-P04	--	
C112.4-P05	--	
C112.4-P06	--	
C112.4-P07	--	
C112.4-P08	--	
C112.4-P09	--	
C112.4-P010	--	
C112.4-P011	--	
C112.4-P012	2	Knowledge gained supports lifelong learning in advanced data management, distributed systems, and operating systems.

C112.5		
C112.5-P01	3	Understanding hierarchical structures like trees improves analytical modeling and problem-solving competence.
C112.5-P02	2	Analyzing BST operations (searching, insertion, deletion) strengthens algorithmic reasoning.
C112.5-P03	2	Implementation of tree traversals and balancing concepts enhances software development and system design capabilities.
C112.5-P04	--	
C112.5-P05	--	
C112.5-P06	--	
C112.5-P07	--	
C112.5-P08	--	
C112.5-P09	--	
C112.5-P010	--	
C112.5-P011	--	
C112.5-P012	2	Provides a basis for lifelong learning in advanced areas such as AVL Trees, Red-Black Trees, and AI search trees.

C112.6		
C112.5-P01	3	Understanding hash functions and key–value structures contribute to fundamental engineering knowledge.
C112.5-P02	2	Analyzing collision resolution methods (chaining, open addressing) enhances analytical thinking.
C112.5-P03	2	Designing and implementing hash tables improves skills in software construction and optimization.
C112.5-P04	--	
C112.5-P05	--	
C112.5-P06	--	
C112.5-P07	--	
C112.5-P08	--	
C112.5-P09	--	
C112.5-P010	--	
C112.5-P011	--	
C112.5-P012	2	Encourages lifelong learning in areas like indexing, cryptographic hashing, and big data structures.

Justification for Avg CO-PO Mapping		
Mapping	**Level**	**Justification**
C112.PO1	3.0	Strong understanding of foundational data structures such as arrays, lists, stacks, queues, trees, and graphs equips students with essential engineering knowledge required to design efficient algorithms and computational models.
C112.PO2	2.3	Analysis of searching, sorting, and structural operations enhances students' ability to evaluate performance,
C112.PO3	2	Implementation of linked lists, stacks, queues, trees, and hashing improves students' hands-on engineering skills in developing efficient software solutions through systematic design, coding, testing, and debugging.
C112.PO4	--	
C112.PO5	1.5	Use of software tools, IDEs, and data-visualization techniques introduces students to modern computational tools required for representing and validating data structures and algorithms.
C112.PO6	0.5	Understanding real-time data structure applications in scheduling, routing, file systems, and memory management helps students relate concepts to societal, industrial, and practical problems.
C112.PO7	--	
C112.PO8	--	
C112.PO9	--	
C112.PO10	--	
C112.PO11	--	
C112.PO12	2	Knowledge of fundamental data structures forms a base for lifelong learning, enabling students to upgrade into advanced fields such as advanced data structures, algorithms, machine learning, database indexing, and system design.

Justification for CO-PSO Mapping		
Mapping	**Level**	**Justification**
C112.PO1	--	
C112.PO2	--	
C112.PO3	--	
C112.PO4	--	
C112.PO5	--	
C112.PO6	--	
C112.PO7	--	
C112.PO8	--	
C112.PO9	--	
C112.PO10	--	
C112.PO11	--	
C112.PO12	--	

Topics beyond Syllabus

S.No.	Description	Proposed Actions
1	VL Trees and Red-Black Trees	Assignments
2	Advanced Hashing Techniques	Assignments

TOPICS BEYOND SYLLABUS/ASSIGNMENT/INDUSTRY VISIT/PROJECTS/NPTEL ETC

Topic beyond Syllabus: Mapping with PO and PSO:

Topic beyond syllabus	**Program Outcomes (PO)**												**Program Specific Outcomes (PSO's)**	
1.	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
	3	3	2	-	3	-	-	-	-	-	-	2	-	-

Justification for Topic beyond the Syllabus (TBS) -PO Mapping.

Mapping	Level	Justification
TBS-PO1	3	The topic extends understanding of fundamental principles and integrates interdisciplinary concepts, deepening the core engineering knowledge
TBS-PO2	4	Students analyze complex, open-ended problems related to the TBS topic, identifying constraints, and applying critical thinking
TBS-PO3	5	Enables students to design innovative solutions or models using advanced techniques beyond syllabus.
TBS -PO4	4	Encourages experimental and research-oriented activities, improving investigative and analytical skills.
TBS -PO5	5	Involves use of modern computational tools, simulation software, or emerging technologies for practical application.
TBS -PO6	3	Highlights ethical considerations in implementing advanced technologies or research outcome
TBS -PO7	4	Motivates self-directed learning beyond the syllabus to stay updated with current industry and research trends.

Justification for Topic Beyond the Syllabus (TBS) -PSO Mapping.

Mapping	Level	Justification
TBS-PSO1	4	The topic enhances understanding of advanced physics concepts beyond the syllabus, encouraging analytical thinking and deeper domain knowledge.
TBS-PSO2	5	Students apply modern tools, simulations, or experiments to solve complex physics problems, promoting research-oriented and innovative learning.

WEB SOURCE REFERENCES:

1	https://www.geeksforgeeks.org/data-structures/
2	https://www.tutorialspoint.com/data_structures_algorithms/index.htm

SYLLABUS / LESSON PLAN:

Sl.No	Unit	Topic	No. of Periods	Text Book / Reference Book	Teaching Aid Used / Methodology	Hours
INTRODUCTION						
1	**1.1**	Introduction to Linear Data Structures	**1**	TB.1	CHALK/TALK	**4**
2		Abstract data types (ADTs) and their implementation	**2**	TB.1	CHALK/TALK	
3		Overview of time and space complexity analysis for linear data structures	**1**	TB.1	CHALK/TALK	
SEARCHING & SORTING TECHNIQUES						
4	**1.2**	Linear Search	**1**	TB.1	CHALK/TALK	**7**
5		Binary Search	**1**			
6		Bubble sort	**1**	TB.1	CHALK/TALK	
7		Selection sort	**1**	TB.1	CHALK/TALK	
8		Insertion sort	**1**	TB.1	CHALK/TALK	
9		Quick Sort (Concept & Partitioning)	**1**	TB.1	CHALK/TALK	
10		Merge Sort (Divide & Conquer)	**1**	TB.1	CHALK/TALK	
LINKED LISTS						
11	**2**	Introduction to Linked Lists	**1**	TB.1	CHALK/TALK	**9**
12		Singly Linked List – Representation	**1**	TB.1	CHALK/TALK	
13		Singly Linked List – Operations (Insert, Delete)	**2**	TB.1	CHALK/TALK	
14		Doubly Linked List – Representation	**1**	TB.1	CHALK/TALK	
15		Circular Linked List – Concepts & Operations	**1**	TB.2	CHALK/TALK	
16		Applications of Linked Lists	**1**	TB.1	PPT	
17		Comparing Arrays vs Linked Lists	**1**	TB.1	CHALK/TALK	
18		Case Studies / Real-Time Use Cases	**1**	--		
STACKS						
19	**3.1**	Introduction to Stacks	**1**	TB.1	CHALK/TALK	**5**
20		Stack Implementation (Array)	**1**	TB.1	CHALK/TALK	
21		Stack Implementation (Linked List)	**1**	TB.1, TB.2	PPT	
22		Applications of Stacks in expression evaluation.	**1**	TB.1, TB.2	CHALK/TALK	
23		Applications of Stacks in backtracking, reversing list etc.	**1**	TB.1, TB.2	CHALK/TALK	
QUEUES						
24	**3.2**	Introduction to Queues	**1**	TB.1	CHALK/TALK	
25		Queue Implementation (Array)	**1**	TB.1	CHALK/TALK	
26		Queue Implementation (Linked List)	**1**	TB.1	CHALK/TALK	

27		Circular Queue / Priority Queue Concepts	**1**	TB.1	CHALK/TALK	**5**
28		Applications of queues in breadth-first search etc.	**1**	TB.1	PPT	
TREES						
29	**4**	Introduction to Trees	**1**	TB.1	CHALK/TALK	**10**
30		Binary Tree	**1**	TB.1	PPT	
31		Representations of Binary Tree	**1**	TB.1	CHALK/TALK	
32		Binary Search Tree (BST)	**1**	TB.1	CHALK/TALK	
33		Properties of Binary Search Tree	**1**	TB.1	CHALK/TALK	
34		Binary Search Tree operations - Insertion, Deletion	**2**	TB.1	CHALK/TALK	
35		Tree Traversal – in order, preorder, post order traversing techniques	**2**	TB.1		
36		Case Studies / Problems	**1**			
INTRODUCTION TO GRAPHS						
37	**5**	Introduction to Graphs	**1**	TB.1	CHALK/TALK	**8**
38		Types of Graphs	**1**	TB.1	CHALK/TALK	
39		Representation of Graphs	**1**	TB.1, TB.2	PPT	
40		Graph Traversals (BFS Algorithm)	**1**	TB.1, TB.2	CHALK/TALK	
41		Graph Traversals (DFS Algorithm)	**1**	TB.1	CHALK/TALK	
42		Graph Applications	**1**	TB.1	CHALK/TALK	
43		Case Studies (Routing, Networking)	**1**		CHALK/TALK	
44		Summary & Revision	**1**			

Topic Beyond Syllabus:

Sl.No.	Topic Beyond Syllabus Planning	PERIODS	Methodology	Text book / references/ web references and additional text book reference
1	Real-Time Applications of Data Structures (OS Scheduling, File Systems, Memory Management)	**2**	**Assignment**	**T1, T3**
2	Introduction to Advanced Trees (AVL, Red-Black Trees)	**1**	**PPT**	**T1, Web Ref-1**
3	Hashing in Cryptography & Databases	**1**	**Seminar**	**T2, Web Ref-2**
4	Graph Applications in Networking (Routing Algorithms)	**1**	**PPT / Chalk-Talk**	**T1, T2**
5	Introduction to Priority Queues & Heaps	**1**	**Assignment**	**T1, Web Ref-2**

Note: Bloom's Taxonomy Levels

BTL1-Remember	**BTL2 – Understand**	**BTL3 –Apply**
BTL4-Analyze	**BTL5 –Evaluate**	**BTL6–Create**

BTL1 – Remember	Recall basic facts, definitions, and foundational concepts of data structures.
BTL2 – Understand	Explain ideas, interpret concepts, and describe the functioning of data structures.
BTL3 –Apply	Use learned concepts to implement algorithms, write programs, and solve problems.
BTL4-Analyze	Break problems into components, compare algorithmic approaches, and examine operations such as time–space tradeoffs.
BTL5 –Evaluate	Justify the choice of algorithms, assess efficiency, and critique different data structure strategies.
BTL6–Create	Design and develop new algorithms, construct optimized solutions, and propose improvements to existing implementations.

Text books (T) / Reference books (R)/Additional text books (A):

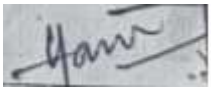
T/R/A	Book Title/Author/Publication
T1	**Fundamentals of Data Structures in C** – Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Universities Press
T2	**Data Structures Using C** – Reema Thareja, Oxford University Press
R1	**Data Structures and Algorithm Analysis in C** – Mark Allen Weiss, Pearson
R2	**Data Structures: A Pseudocode Approach with C** – Richard F. Gilberg, Behrouz A. Forouzan, Cengage Learning
R3	**Classic Data Structures** – Debasis Samanta, Prentice Hall of India
R4	**Data Structures and Algorithms Made Easy** – Narasimha Karumanchi, CareerMonk Publications
A1	**Introduction to Algorithms (CLRS) – Cormen, Leiserson, Rivest & Stein, MIT Press**
A2	**Mastering Data Structures & Algorithms in C and C++ – Abdul Bari (Web Resource / Book)**

ASSESSMENT METHODOLOGIES-DIRECT

☑ASSIGNMENTS	☑ STUD. SEMINARS	☑TESTS/MODEL EXAMS	☑UNIV. EXAMINATION
☐ STUD. LAB PRACTICES	☐ STUD. VIVA	☐ MINI/MAJOR PROJECTS	☐ CERTIFICATIONS
☐ ADD-ON COURSES	☐ OTHERS		

ASSESSMENT METHODOLOGIES-INDIRECT

☑ ASSESSMENT OF COURSE OUTCOMES (BY FEEDBACK, ONCE)	☑ STUDENT FEEDBACK ON FACULTY (TWICE)
☐ ASSESSMENT OF MINI/MAJOR PROJECTS BY EXT. EXPERTS	☐ OTHERS



Prepared by

Approved by