

Department of Computer Science Engineering – (Data Science)

COURSE MODULE OF THE SUBJECT TAUGHT FOR THE SESSION 2026-27 (ODD SEM)

Course Syllabi with CO's

Faculty Name: Dr. Neethi M V			Academic Year: 2026 - 2027				
Department: Computer Science & Engineering- Data Science							
Course Code	Course Title	Core / Elective	Prerequisite	Contact Hours			Total Hrs/ Sessions
				L	T	P	
1BCSL306	DATA STRUCTURES LABORATORY	PCCL	C Programming	0	0	2	24
Topics Covered as per Syllabus							
List of Experiments							
1. Develop a Program in C for the following: a. Define a structure Book with the following fields: Book_ID, Title, Author, Price, Availability Status (Available/Issued) b. Dynamically allocate memory to store the details of N books, where N is entered by the user. c. Implement the following functions: i. create() – To allocate memory and read the details of N books and store them in the allocated memory. ii. display() – To display the details of all books available in the library. iii. search() – To search for a book using Book ID and display the book details if found. iv. issueBook() – To issue a book by updating its availability status and display an appropriate message. v. returnBook() – To return an issued book by updating its availability status and display an appropriate message. d. Provide a menu-driven interface with the following options: i. Add Book Records ii. Display All Book Records iii. Search Book by Book ID iv. Issue a Book v. Return a Book vi. Exit							
2. Develop a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)							

Department of Computer Science Engineering – (Data Science)

- a. Push an Element on to Stack
- b. Pop an Element from Stack
- c. Demonstrate how Stack can be used to check Palindrome
- d. Demonstrate Overflow and Underflow situations on Stack
- e. Display the status of Stack
- f. Exit

Support the program with appropriate functions for each of the above operations

3. Develop a program in C for simulating a Printer Queue using Queue data structure with the following

operations:

- a. Add a print job to the Printer Queue.
- b. Process (delete) a print job from the Printer Queue.
- c. Display all pending print jobs in the Queue.
- d. Display the number of print jobs currently waiting in the Queue.
- e. Demonstrate Queue overflow and underflow conditions.
- f. Exit.

4. Develop a menu-driven Program in C for the following operations on a Singly Linked List of Student Records:

- a. Create a linked list using front insertion.
- b. Display all student records and count the number of nodes.
- c. Insert and delete nodes at the beginning and end of the list.
- d. Search for a student record using USN.

5. Develop a Program in C for the following:

- a. Create a Binary Tree by inserting integer elements entered by the user in level order.
- b. Implement the following functions:
 - i. createTree() – Create a binary tree using dynamic memory allocation.
 - ii. preorder() – Display the elements using Preorder Traversal.
 - iii. inorder() – Display the elements using Inorder Traversal.
 - iv. postorder() – Display the elements using Postorder Traversal.
- c. Provide a menu-driven interface with the following options:
 - i. Create Binary Tree
 - ii. Display Preorder Traversal
 - iii. Display Inorder Traversal
 - iv. Display Postorder Traversal
 - v. Exit

6. Develop a Program in C for the following operations on Graph(G) of Cities

- a. Create an Adjacency Matrix representing the graph of N cities

Department of Computer Science Engineering – (Data Science)

- b. Print all the nodes reachable from a given starting node in a digraph using DFS/BFS method.

PART – B

OPEN ENDED EXPERIMENTS

Open-ended experiments are a type of laboratory activity where the outcome is not predetermined and students are given the freedom to explore, design, and conduct the experiment based on the problem statements as per the concepts defined by the course coordinator. It encourages creativity, critical thinking, and inquiry-based learning.

1. Develop a C program to:

- Read two sparse matrices of the same order.
- Convert both into 3-tuple representation.
- Perform matrix addition.
- Display the resultant matrix in both normal matrix form and 3-tuple form.

2. Develop a C-based Expression Conversion tool that uses stack data structure to convert arithmetic expressions from infix notation to postfix notation. The tool should support alphanumeric operands, parentheses, and the operators (+, -, *, /, %, ^). Analyze the correctness of the conversion by testing various valid expressions.

3. Develop a menu-driven Program in C for the following operations on Circular Queue (integer/floating point data):

- Insert an element into the Circular Queue.
- Delete an element from the Circular Queue.
- Display the contents of the Circular Queue.
- Demonstrate overflow and underflow conditions.
- Exit.

4. Develop a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee

Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo etc.

- Create a DLL of N Employees Data by using end insertion.
- Display the status of DLL and count the number of nodes in it
- Perform Insertion and Deletion at End of DLL
- Perform Insertion and Deletion at Front of DLL
- Demonstrate how this DLL can be used as Double Ended Queue.
- Exit

Department of Computer Science Engineering – (Data Science)

5. Develop a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers.
 - a. Create a BST of N Integers
 - b. Traverse the BST in Inorder, Preorder and Post Order
 - c. Search the BST for a given element (KEY) and report the appropriate message
 - d. Exit
6. Develop a Program in C for an Employee Record Management System using Hashing with the following operations:
 - a. Create a hash table to store employee records using Employee ID as the key.
 - b. Insert employee records into the hash table using the Hash function $H: K \rightarrow L$ as $H(K)=K \bmod m$
 - c. Resolve collisions using Linear Probing.
 - d. Search for an employee record using a given Employee ID and display the details.
 - e. Display the contents of the hash table along with the corresponding memory locations.

List of Textbook and URLs

Text books:

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2023.

Reference books / Manuals:

1. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2023.
2. A M Tenenbaum, Data Structures using C, PHI, 1989
3. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

Course Outcomes	At the end of the course, the student will be able to: CO1- Develop C programs to implement fundamental data structures and their operations for solving computational problems. CO2- Make use of appropriate data structures to organize, store, and manipulate data for solving real-world problems using C. CO3- Build a laboratory record consisting of problem statements, algorithm/design, mplementation, program outputs, and result with appropriate inferences.
------------------------	--

Assessment Detail: The assessment for each course is equally divided between Continuous Internal Evaluation (CIE) and the Semester End Examination (SEE), with each component carrying 50% weightage (i.e., 50 marks each).

The CIE marks awarded shall be based on the continuous evaluation of the laboratory report using a defined set of rubrics. Each experiment report can be evaluated for 30 marks. Total marks for the all report shall be scaled down to 30 marks. The laboratory test (duration 03 hours) at the end of the last week of the semester /after completion of all the experiments (whichever is early) shall be conducted for 50 marks and Marks scored shall be scaled down to 20 marks. Final average of marks for report and test shall be out of 50 marks.

Department of Computer Science Engineering – (Data Science)

For both CIE and SEE, the student is required to conduct one experiment each from both Part A and Part B.

The Correlation of Course Outcomes (CO's) and Program Outcomes (PO's)

Course	1BDSL306		TITLE: Data Structures laboratory					Faculty		Dr.Neethi M V	
Code:								Name:			
List of Course Outcomes	Program Outcomes										
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11
CO-1	3	3	3	3	3	-	-		1	2	2
CO-2	3	3	3	2	3	1	-	-	1	2	2
CO-3	2	2	2	2	3	-	-	1	2	3	2

Note: 3 = Strong Contribution, 2 = Average Contribution, 1 = Weak Contribution, - = No Contribution

The Correlation of Course Outcomes (CO's) and Program Specific Outcomes (PSO's)

Course Code	BDSL456D	Title: MongoDB	Faculty name: Dr. Neethi M V
List of Course Outcomes	Program Specific Outcomes		
	PSO1	PSO2	PSO3
CO1	2	-	-
CO2	2	-	-
CO3	2	-	-
CO4	2	-	-

Note: 3 = Strong Contribution, 2 = Average Contribution, 1 = Weak Contribution, - = No Contribution