

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

COURSE MODULE OF THE SUBJECT TAUGHT FOR THE SESSION 2026-27

Faculty Name: Prof. Bhavana V R	Academic Year: 2026 - 2027
--	-----------------------------------

Department: Cyber Security

Course Code	Course Title	Core / Practical	Prerequisite	Contact Hours				Total Hrs
				L	T	P	S	
1BCSL306	Data structures Laboratory	Practical	-	-	-	28	-	28

Course outcome (Course Skill Set)

At the end of the course, the student will be able to:

1. Develop C programs to implement fundamental data structures and their operations for solving computational problems.
2. Make use of appropriate data structures to organize, store, and manipulate data for solving real-world problems using C.
3. Build a laboratory record consisting of problem statements, algorithm/design, implementation, program outputs, and result with appropriate inferences.

PART – A

FIXED SET 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

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

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)

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.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

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
- 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:

- a. Read two sparse matrices of the same order.
- b. Convert both into 3-tuple representation.
- c. Perform matrix addition.
- d. 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.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

3. Develop a menu-driven Program in C for the following operations on Circular Queue (integer/floating point data):
 - a. Insert an element into the Circular Queue.
 - b. Delete an element from the Circular Queue.
 - c. Display the contents of the Circular Queue.
 - d. Demonstrate overflow and underflow conditions.
 - e. 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.
 - a. Create a DLL of N Employees Data by using end insertion.
 - b. Display the status of DLL and count the number of nodes in it
 - c. Perform Insertion and Deletion at End of DLL
 - d. Perform Insertion and Deletion at Front of DLL
 - e. Demonstrate how this DLL can be used as Double Ended Queue.
 - f. Exit
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.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

Suggested Learning Resources: (Text Book/ Reference Book/ Manuals):

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.

Web links and Video Lectures (e-Resources):

1. GNU-GCC Documentation: <https://gcc.gnu.org/>
2. Virtual Labs on Data Structures-1: <https://dsl-iiith.vlabs.ac.in/List%20of%20experiments.html>
3. DSA Tutorial: <https://www.geeksforgeeks.org/dsa/dsa-tutorial-learn-data-structures-and-algorithms/>
4. NPTEL Course on DSA: <https://nptel.ac.in/courses/106106130>
5. NPTEL Course on Introduction to C Programming: <https://nptel.ac.in/courses/106104128>

Teaching-Learning Process (Innovative Delivery Methods):

The following are sample strategies that educators may adopt to enhance the effectiveness of the teaching-learning process and facilitate the achievement of course outcomes.

1. Engineering tool usage for the conduction of experiment
2. Demonstration through ICT tools
3. Use of virtual labs

Assessment Structure:

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.

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

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

Rubrics for CIE – Continuous assessment:

Criteria/Scale	Superior (5)	Good (4)	Fair (3)	Needs Improvement (2)	Unacceptable (1)
Fundamental Knowledge: Understanding the problem statement and selection of appropriate data structure [CO1, CO2] [PO1, PO2]	Demonstrates in-depth understanding of data structure concepts and correctly identifies the most suitable data structure for the given problem.	Demonstrates good understanding of data structure concepts and selects an appropriate data structure with minor guidance.	Demonstrates satisfactory understanding and selects a suitable data structure for the problem.	Demonstrates partial understanding and selects an inappropriate or partially suitable data structure.	Unable to understand the problem or identify an appropriate data structure.
Design of Algorithm and Data Structure [CO1, CO2] [PO2, PO3]	Develops multiple algorithmic solutions, compares them, and justifies the best design with appropriate data structure selection.	Develops more than one design but provides limited justification for selecting the best solution.	Develops a correct algorithm and data structure with basic explanation.	Develops an incomplete algorithm or partially correct design.	Unable to design a suitable algorithm or data structure.
Implementation (Program Coding) [CO1, CO2] [PO5, PO8]	Implements the selected data structure efficiently using appropriate C constructs with correct operations and modular programming practices.	Implements the selected data structure correctly and explains the implementation clearly.	Implements the required data structure with minor coding issues.	Implements the program with significant errors or incomplete operations.	Unable to implement the required data structure program.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

Program Debugging and Testing [CO1, CO2] [PO5, PO8]	Successfully compiles, debugs, and validates the program for all test cases without syntax, semantic, or logical errors.	Identifies and rectifies errors independently with good understanding of debugging.	Corrects most errors with partial understanding of debugging.	Corrects errors only with assistance and limited understanding.	Unable to debug or execute the program successfully.
Results and Analysis [CO1, CO2] [PO4]	Executes the program using various test cases, validates all data structure operations, and provides detailed analysis with appropriate conclusions.	Executes all test cases and explains the results correctly.	Executes most test cases and provides basic analysis.	Executes the program but provides limited or incorrect analysis.	Unable to verify or interpret the program output correctly.
Demonstration and Documentation [CO3] [PO8, PO9, PO11]	Demonstration is excellent. Lab record is well-organized with problem statement, algorithm, data structure representation, program, output, observations, and conclusions presented using proper formatting.	Demonstration is good. Lab record is organized with all major sections and appropriate formatting.	Demonstration is satisfactory. Lab record contains most required sections with acceptable formatting.	Demonstration is incomplete. Lab record has missing sections or inadequate formatting.	Demonstration is poor or absent. Lab record is incomplete, improperly formatted, or not submitted on time.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

Rubrics for SEE / CIE Test:

Criteria /Scale	Superior (5)	Good (4)	Fair (3)	Needs Improvement (2)	Unacceptable (1)
Fundamental Knowledge (2) [CO1, CO2] [PO1]	Demonstrates excellent knowledge of data structures, their operations, and applications.	Demonstrates good knowledge of major data structure concepts.	Demonstrates average understanding of basic concepts.	Demonstrates limited conceptual understanding.	Does not understand the fundamental concepts.
Understanding of Problem Definition (1) [CO1, CO2] [PO2]	Clearly understands the problem and identifies the most appropriate data structure.	Understands the problem with minor clarification.	Demonstrates partial understanding of the problem.	Demonstrates minimal understanding of the problem.	Unable to understand the problem statement.
Design and Implementation (3) [CO1, CO2] [PO3]	Designs an efficient algorithm and correctly implements the selected data structure using appropriate C constructs.	Designs and implements the required solution with minor issues.	Implements the core algorithm with partial correctness.	Partially implements the solution with significant errors.	Unable to design or implement the solution.
Results and Analysis (2) [CO1, CO2] [PO4]	Successfully executes the program using multiple test cases, validates all operations, and provides accurate analysis.	Executes the program correctly and provides good analysis.	Executes the program with limited test cases and basic analysis.	Executes the program but cannot properly analyze the results.	Unable to execute or analyze the program.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

Communication (Viva Voce) (2) [CO3] [PO8, PO9]	Clearly explains the working of the selected data structure, algorithm, implementation, and application using correct technical terminology.	Explains the concepts with good communication and correct terminology.	Explains basic concepts with average communication skills.	Provides incomplete explanations with incorrect terminology.	Unable to explain the implementation or answer questions satisfactorily.
--	--	--	--	--	--

- To qualify and become eligible to appear for SEE, in the **CIE component**, a student must secure a **minimum of 40% of 50 marks, i.e., 20 marks.**
- To pass the **SEE component**, a student must secure a **minimum of 35% of 50 marks, i.e., 18 marks.**
- A student is deemed to have **successfully completed the course** if the **combined total of CIE and SEE is at least 40 out of 100 marks.**

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

Subject Code:	1BCSL306		TITLE: DATA STRUCTURES AND APPLICATIONS LAB								Faculty Name:	Ms. Bhavana V R	
List of	Program Outcomes											Total	
Course Outcome s	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	2	3	-	-	-	-	-	3	17	
CO-2	3	3	3	2	3	-	-	-	-	-	3	17	
CO-3	3	3	3	2	3	-	-	-	-	-	3	17	
CO-4	3	2	2	2	3	-	-	-	-	-	3	15	
Total	12	11	11	08	12	-	-	-	-	-	12	66	

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING - CYBER SECURITY

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

Course Code:	1BCSL306	TITLE: DATA STRUCTURES AND APPLICATIONS LAB	Faculty Name: Ms. Bhavana V R
List of Course Outcomes	Program Specific Outcomes		Total
	PSO-1	PSO-2	
CO-1	3	-	3
CO-2	3	-	3
CO-3	3	-	3
CO-4	3	-	3
Total	12	-	12

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

COURSE COORDINATOR

HOD