

<b>Course Title: Data Structures Laboratory</b>	<b>Course Code: 20CS36L</b>
<b>Credits (L: T:P): 0:0:1.5</b>	<b>Contact Hours(L: T: P):0:0:39</b>
<b>Type of Course: Practical</b>	<b>Category: Professional Core Course</b>
<b>CIE Marks: 50</b>	<b>SEE Marks: 50</b>

**Pre-requisite:** Knowledge of C programming

SL NO	Course Objectives:
1.	To gain knowledge in practical applications of data structures and to introduce various techniques for representation of the data in the real world.
2.	Strengthening the ability of the students to identify and apply the suitable data structure for the given real world problem.
3.	To develop skills to design and analyze simple linear and non linear data structures.

Lab Session No.	Course Content
1	<p>Define a structure called Student with the members: name, reg_no, marks in 3 tests and average_marks.</p> <p>Develop a menu driven program to perform the following by writing separate function for each operation: a) read information of N students b) display student's information c) to calculate the average of best two test marks of each student.</p> <p>Note: Illustrate the use of pointer to an array of structure and allocate memory dynamically using malloc () /calloc()/realloc().</p>
2	<p>Define a structure called Time containing 3 integer members (hour,minute,second). Develop a menu driven program to perform the following by writing separate function for each operation.</p> <p>a) To read time b) To display time c) To Update time d) Add two times by writing Add (T1, T2) which returns the new Time.</p> <p>Update function increments the time by one second and returns the new time (if the increment results in 60 seconds, then the second member is set to zero and minute member is incremented by one. If the result is 60 minutes, the minute member is set to zero and the hour member is incremented by one. Finally, when the hour becomes 24, Time should be reset to zero)</p> <p>Note: Illustrate the use of pointer to a structure variable and passing and returning of structure type to and from the function (both by value and reference).</p>
3	Develop a menu driven program to implement various operations on array storage representation with static and dynamic memory allocation.

4	Develop a menu driven program to implement singly linked list with various operations such as i) Insertion and Deletion at front/rear ii) Insertion and Deletion at the specified position iii) Delete by Key iv) Search by key v) Create an ordered list vi) Reverse a list vii) Creating a copy of the list
5	Develop a menu driven program to implement Circular singly linked list with various operations such as i) Insertion and Deletion at front/rear ii) Insertion and Deletion at the specified position iii) Delete by Key iv) Search by key v) Create an ordered list vi) Reverse a list vii) Creating a copy of the list
6	Develop a menu driven program to implement Double linked list with various operations such as i) Insertion and Deletion at front/rear ii) Insertion and Deletion at the specified position iii) Delete by Key iv) Search by key v) Create an ordered list vi) Reverse a list vii) Creating a copy of the list
7	Develop a menu driven program to implement Circular Double linked list with Header node to perform various operations such as i) Insertion and Deletion at front/rear ii) Insertion and Deletion at the specified position iii) Delete by Key iv) Search by key v) Create an ordered list vi) Reverse a list vii) Creating a copy of the list
8	Develop a menu driven program to implement Stack with static and dynamic memory allocation mechanisms using array storage representation. (Represent Stack using structure)
9	Convert infix expression to postfix expression. Convert infix expression to prefix Evaluate the postfix expression.
10	Implement ordinary Queue with static and dynamic memory allocation mechanisms using array storage representation.(Represent Queue using structure) Implement Circular Queue with static and dynamic memory allocation mechanisms using array storage representation.(Represent Queue using structure)

11	Implement Double Ended Queue with static and dynamic memory allocation mechanisms using array storage representation. (Represent Queue using structure) Implement Priority Queue with static and dynamic memory allocation mechanisms using array storage representation. (Represent Queue using structure)
12	Develop a menu driven program to implement binary search tree and traversal techniques. Develop a menu driven program to implement Graph traversal techniques.
13	<b>Lab Test/Event</b>

### Reference Books:

Sl. No.	Author/s	Title	Publisher Details
1	Ellis Horowitz and Sartaj Sahni	Fundamentals of Data Structures in C	2 <sup>nd</sup> edition, Universities Press, 2014
2	Richard F. Gilberg and Behrouz A. Forouzan	Data structures, A Pseudo code Approach with C	2 <sup>nd</sup> Edition, Universities Press, 2014
3	Seymour Lipschutz	Data Structures	Schaum's Outlines, Revised 1 <sup>st</sup> edition, McGraw Hill, 2014
4	Yedidyah, Augenstein, Tannenbaum	Data Structures Using C	First edition, Pearson Education, 2019.

### Web References:

Sl. No.	Web link
1	<a href="https://nptel.ac.in/courses/106/105/106105085/">https://nptel.ac.in/courses/106/105/106105085/</a>
2	<a href="https://nptel.ac.in/courses/106/103/106103069/">https://nptel.ac.in/courses/106/103/106103069/</a>

### Course Outcomes: After completing this course, students should be able to:

CO-1	Apply advance C programming techniques such as pointers, dynamic memory allocation, structures to develop solutions for various problems.
CO-2	Design and implement different data structures such as, stack, queue, linked list, graph and tree using C as the programming language with static or dynamic allocation.
CO-3	Analyze, choose and apply appropriate data structure for implementing solutions for various practical problems.

Course Outcomes	Program Outcomes												PSO's			
	PO 1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO-1	3	3	3	3	1	1	1	1	2	2	1	3	3	3	3	3
CO-2	3	3	3	3	1	1	1	1	2	2	1	3	3	3	3	3
CO-3	3	3	3	3	1	1	1	1	2	2	1	3	3	3	3	3

*Educate Elevate Enlighten*

**JSS Mahavidyapeetha**  
**JSS Science And Technology University**  
(Established Under JSS Science and Technology University Act No. 43 of 2013)



*Educate Elevate Enlighten*

**JSS Mahavidyapeetha**  
**JSS Science And Technology University**  
(Established Under JSS Science and Technology University Act No. 43 of 2013)

