

DATA STRUCTURES LABORATORY

Lab Hours/ Week	: 3	Credits :	1.5
Sub. Code	: 3CSL01	CIE Marks :	50
		SEE Marks :	50

Note:

80% of total marks for PART-A question.

20% of total marks for PART-B.

The student has to pick one question in PART-A from lots.

PART - A

1. Write a C program to create a sequential file with at least five records, each record having the structure shown below:

EMPLOYEE_ID	NAME	DEPARTMENT	SALARY	AGE
Non-Zero Positive integer	25 Characters	25 Characters	Positive Integer	Positive integer

Write necessary functions to perform the following operations:

- i) to display all the records in the file.
 - ii) to search for a specific record based on
EMPLOYEE_ID/SALARY/DEPARTMENT/AGE. In case if the required record is not found, suitable message should be displayed.
2. Write a C program to implement STACK to perform the PUSH, POP and DISPLAY operations.
3. Write a C program to convert the given infix expression to postfix expression.
4. Write a C program to evaluate the given prefix expression.
5. Write a C program to implement ordinary QUEUE to perform the insertion, deletion and display operations.
6. Write a C program to implement CIRCULAR QUEUE to perform the insertion, deletion and display operations.
7. Write a C program to perform the following operations using singly linked list:
- a) to insert a node at the end of the list.
 - b) to delete the first node in the list.
 - c) to insert a node at the specified position in the list ($1 \leq \text{pos} \leq n+1$ where 'n' is the total number of nodes in the list & 'pos' is the position where data is to be inserted).
 - d) to display the contents of the list.
 - e) to reverse a given list.

[**NOTE:** either of the operations a, b, d & e or operations b, c, d & e must be specified in the examination]

8. Write a C program to construct two ordered singly linked lists with the following operations:

- a) insert into list1.
- b) insert into list2.
- c) to perform UNION(list1,list2)
- d) to perform INTERSECTION(list1,list2)
- e) display the contents of all three lists.

[**NOTE:** either of the operations a, b, c & e or operations a, b, d & e must be specified in the examination]

9. Write a C program for the following using singly linked list.

a)to implement STACK to perform PUSH, POP and DISPLAY operations.

b)to implement ordinary QUEUE for performing insertion, deletion and display operations.

c)to add two given polynomials.

[**NOTE:** either a or b or c must be specified in the examination]

10. Write a C program to perform the following operations using doubly linked list with header node. Header node should maintain the count of number of nodes in the list after each operation:

a)to insert a node next to a node whose information field specified.

b)to delete first node if pointer to the last node is given.

c)to delete a node at the specified position in the list ($1 \leq \text{pos} \leq n$ where 'n' is the total number of nodes in the list & 'pos' is the position where data is to be deleted).

d)to display the contents of the list.

e)to swap n^{th} and m^{th} nodes in the list.

[**NOTE:** either of the operations a, b, d & e or operations a, c, d & e must be specified in the examination]

11. Write a C program to implement DEQUE using doubly linked list to perform the insertion, deletion and display operations.

12. Write a C program to perform the following operations:

a) Construct a binary search tree of integers.

b) Traverse the tree in inorder/ preorder/ postorder.

c) Delete a given node from the BST.

13. Write a C program to construct an expression tree for a given postfix expression and evaluate the expression tree.

PART – B

Additional question will be asked in the Semester End Examination (SEE) for modification on the PART-A question obtained.

Open Ended Problems

These problems are introduced to make the students to apply the knowledge of Data Structures in solving real world problems. Following are the guidelines:

- Each team (3/4 students) from each batch should come up with an application of any of the data structures like files, stacks, queues, linked lists and trees.
- Faculty in-charge of the lab will formulate the problem statement and distribute the same to each team, based on the application submitted.
- Each team has to implement the problem statement given to them within the deadline.
- Implementation will be considered for Continuous Internal Evaluation (CIE) and it will be based on individual contribution of the students in each team.