|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Course Code | | Course Title | L | T | P | Credits | QP |
|  | | **DATA STRUCTURES & ALGORITHMS LABORATORY**  **LAB EXPERIMENTS** | **0** | **0** | **4** | 2 |  |
| Pre -Requisite: Basic knowledge of Algorithms | | | | | | | |
| Course Outcome: student can able to | | | | | | | |
| CO1 | Understand the basic concepts of data structures, remember the fundamental concepts. | | | | | | |
| CO2 | Understand the programming by comparing the types of data structures concepts. | | | | | | |
| CO3 | Develop the codes for implementing different operations on data structures. | | | | | | |
| CO4 | Analyses of the codes regarding different operations on data structures. | | | | | | |
| CO5 | Solve and evaluate complex problems by coding on linear and non-linear data structures. | | | | | | |
| CO6 | Design the codes on advanced concepts of data structure by implementation in different application of data structure. | | | | | | |
| Experiment-1(functions and arrays)  Q1) Write a program to create methods for performing addition, subtraction, multiplication and division on 2 integers.  Q2) Write a program to create an UDF for input 10 numbers into a 1D array. Create two functions MAX() and MIN(). MAX() is used to return the largest element and MIN() is used to return the smallest number in array.  Q3) Write a C program to create methods for operations insertion and display on 1D array of elements using UDF.  Q4) Write a C program to create methods for operations deletion, and display on 1D array of elements using UDF. | | | | | | | |
| Experiment 2: (concepts of matrix and sparse matrix)  Q1) Write a C program to create function for performing matrix multiplication using UDF  Q2) Write a C program to input elements into a square matrix and display the transpose of it using UDF.  Q3) Write a program to input elements into a 4X4 matrix and display the sum of individual row elements using UDF.  Q4) Write a program to input elements into a 4X4 matrix, check it for sparse or not. If sparse then store the non-zero elements into an alternate matrix and then display it using UDF. | | | | | | | |
| Experiment 3: (stack and queue)  Q1) Write a program using C to create a stack of numbers and perform using UDF:  (i) push operation (ii) pop operation (iii) display operation  Q2) Write a C program to create a linear queue and perform the following operations using UDF: (i) insertion ii) deletion and iii) Traversal  Q3) Write a C program to create a circular queue and perform the following operations using UDF: (i) insertion ii) deletion and iii) Traversal | | | | | | | |
| Experiment 4: (searching and sorting)  Q1) Write a program to implement linear search on array elements using UDF  Q2) Write a program to implement binary search on array elements using UDF  Q3) write a program to implement selection sort on a given list of array elements.  Q4) Write a program to input a string and sort the alphabets in ascending order. | | | | | | | |
| Experiment 5: (sorting)  Q1) Write a program to input elements into two arrays A[5] and B[5]. Input the elements in ascending order and then merge their values into a resultant array C[10] in sorted manner using UDF.  Q2) Write a program to implement insertion sort on a given list of array elements.  Q3) Write a C program to implement quick sort to a given list of integers to sort in ascending order.  Q4) write a program to implement bubble sort on a given list of array elements. | | | | | | | |
| Experiment 6:( Pointer, structure and DMA)  Q1) Write a program to store N numbers using dynamic memory allocation and then find the largest element using UDF.  Q2) Write a C program to create a structure called student to store your rollno, name, age. Create an array to input 5 students data and then create an UDF to display details where age>=20.  Q3) Write a program to create a structure for products of a super market. Store product no, name and cost for N products using dynamic memory allocation. Display the products whose cost is in between 100 rupees to 1000 rupees. | | | | | | | |

|  |
| --- |
| Experiment 7 : (single linked list)  Q1) Write a C program to perform the operations on a single linked list:  i) Insertion at beginning, ii) Deletion of 1st node iii) display all nodes  Q2) Write a C program to perform the operations on a single linked list:  i) insertion at end, i) deletion of last node iii) display all the nodes  Q3) Write a C program to perform the operations on a single linked list:  i) insertion of a new node after a node item ii) searching for a node item iii) deletion of a node item  iv) display all the nodes. |
| Experiment-8  Q1) Write a C program that uses functions to implement linked stack on single linked list.  Q2) Write a C program that uses functions to implement linked queue on single linked list. |
| Experiment-9 (double linked list)  Q1) Write a C program to perform the operations on a single linked list:  i) Insertion at beginning, ii) Deletion of 1st node iii) display all nodes  Q2) Write a C program to perform the operations on a single linked list:  i) insertion at end, i) deletion of last node iii) display all the nodes  Q3) Write a C program to perform the operations on a single linked list:  i) insertion of a new node after a node item ii) searching for a node item iii) deletion of a node item iv) display all the nodes. |
| Experiment 10: (linked stack, linked queue)  Q1) Write a program to create a single linked list for storing the N cricket player details having members player name, team name and bating average. Display only those players information whose batting average>=50  Q2) Write a program to create a double linked list for storing account details of bank customers such as AC no, name, balance. Store details for N bank account holders and find the total balance for all account holders. |
| Topic Beyond Syllabus:  Q1) Write a program to construction a Binary Search Tree for storing N unique numbers. Apply insertion(), deletion() and display() operation on it using UDF.  Q2) Write a program to create a connected graph when its adjacency list is given and display it. |
| CO-PO Mapping: |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |  | | CO1 | 3 | 2 |  |  |  |  |  |  |  |  |  |  |  | | CO2 | 2 | 2 | 2 |  |  |  |  |  |  |  |  |  |  | | CO3 | 2 | 3 | 3 | 1 |  |  |  |  |  |  |  |  |  | | CO4 | 2 | 2 | 3 | 2 | 1 |  |  |  | 1 |  |  |  |  | | CO5 | 2 | 3 | 2 |  |  |  |  |  | 1 | 1 |  |  |  | | CO6 | 1 | 2 |  |  |  |  |  |  |  |  |  |  |  | |  | 2.00 | 2.33 | 1.67 | 0.50 | 0.17 |  |  |  | 0.33 | 0.17 |  |  | 1.02 | |

Signature of Faculty Signature of HOD