Data Structure Lab (KCS351)
List of Programs

NO.	AB No.	S.No.	Program	Domain	*Problem statement/link*
1	Lab1	1	Traversal	**Arrays**	Program for traversing array elements.
2		2	Insertion	**Arrays**	Program to insert the given elements into an array.
3		3	Insertion	**Arrays**	Program for insetion in the sorted array
4		4	Deletion	**Arrays**	Program for delete the given elements into an array.
5		5	Missing Number	**Arrays**	Program for Missing number in an array
6		6	Element analysis	**Arrays**	Program to find which element is repeated in the array and which is not
7		7	Arrangement and de–arranger	**Arrays**	Program for reversal of an array.
8	Lab 2	8	Merging	**Arrays**	Program for merging two sorted arrays
9		9	Set Operation	**Arrays**	Program for Set union
10		10	Set Operation	**Arrays**	Program for Set Intersection
11		11	Set Operation	**Arrays**	Program for Set Difference
12		12	Set Operation	**Arrays**	Program for Set Symmetric Difference
13	In class	13	Address Computation	**Arrays**	Program for computation of address of given element in the one dimensional array and verification with the Physical Address
14	In class	14	Address Computation	**Arrays**	Program for computation of address of given element in the two dimensional array and verification with the Physical Address
15	In class	15	Address Computation	**Arrays**	Program for computation of address of given element in the three dimensional array and verification with the Physical Address
16	Lab 3	16	Matrix	**Arrays**	Program for Matrix Addition
17		17	Matrix	**Arrays**	Program for Matrix Subtraction
18		18	Matrix	**Arrays**	Program for Matrix Multiplication
19		19	Matrix	**Arrays**	Program for Matrix Transpose
20		20	Matrix	**Arrays**	Program for finding Matrix Determinant
21		21	Matrix	**Arrays**	Program for Matrix transposition without second matrix
22	*Lab 4*	1	Linear Search	***Searching***	program for Linear Search
23		2	Binary Search	***Searching***	Program for Binary search
24		3	Variation of Binary Search	***Searching***	Program for Ternary search
25		4	Variation of Binary Search	***Searching***	Program for Jump Search
26		5	Variation of Binary Search	***Searching***	Program for interpolation search
27		6	Index Sequential Search	***Searching***	Program for Index sequential Search
28		7	Variation of Binary Search	***Searching***	Program for Exponential search
29		1	Hash function	***Hashing***	Program for Hash Table Implementation for Basic Hash Function (Without collisions)
30		2	Collision resolution	***Hashing***	Program for Hash Table Implementation for Collision Resoulution using Linear Probing

31	***Lab 5***	3	Collision resolution	***Hashing***	Program for Hash Table Implementation for Collision Resoulution using Quadratic Probing
32		4	Collision resolution	***Hashing***	Program for Hash Table Implementation for Collision Resoulution using Double Hashing/Re-Hashing
33		5	Collision resolution	***Project***	***Program for Hash Table Implementation for Collision Resoulution using Separate Chaining***
34	**Lab 6**	1	Tail Recursion	Recursion	Program for finding factorial of a given number using recursion
35		2	Tail Recursion	Recursion	Program for Computing A raised to power n using Recursion
36		3	Tree Recursion	Recursion	Program for finding nth Fibonacci number using Recursion and improving its run time to save stack operations
37		4	Tail Recursion	Recursion	Program for finding GCD of two numbers using Recursion
38		5	Tail Recursion	Recursion	Binary Search with Recursion
39		6	Mixed Recursion	Recursion	Program for Towers of Hanoi for n disk (user defined)
40		7	Tail Recursion	Recursion	Program to reverse the given number using Recursion
41		8	Tail Recursion	Recursion	Finding sum of the digits of the number
42		9	Decision problem	Recursion	To check if the given string is a palindrome using Recursion
43	**Lab 7**	1	O(N2) Sorting	Sorting	Program for Bubble Sort
44		2	O(N2) Sorting	Sorting	Program for Selection Sort
45		3	O(N2) Sorting	Sorting	Program for Insertion Sort
46		4	O(N2) Sorting	Sorting	Program for Implementation of Shell Sort
47		5	O(NlogN) Sorting	Sorting	Program for Merge Sort
48	**Lab 8**	6	O(NlogN) Sorting	Sorting	Program for Quick Sort
49		7	O(NlogN) Sorting	Sorting	Program for Median Quick Sort
50		8	O(NlogN) Sorting	Sorting	Program for Randomized Quick Sort
51		9	Sorting in linear time	Sorting	Program for Counting Sort
52		10	Sorting in linear time	Sorting	Program for Radix Sort
53		11	O(NlogN) Sorting	Sorting	Program for Heap Sort
54		1		Structure	Get the input of student: Name, Roll No, Marks in 6 subjects in 12th. Find if the student is eligible for admission in Delhi University. A student is eligible for DU if he has scored 95 % or more in Best 4.
55		2		Structure	Write a program to store and print the roll no., name, age and marks of a student using structures. Write a program to store the roll no. (starting from 1), name and age of 5 students and then print the details of the student with roll no. 1.
56		3		Structure	3. Write a program to store and print the roll no., name, age, address and marks of 15 students using structure.
57		4		Structure	4. Write a program to add two distances in inch-feet using structure. The values of the distances is to be taken from the user.
58		5		Structure	5. Write a program to add two complex numbers using structure. The values of the complex number is to be taken from the user.

59		6		Structure	6. Write a program to add two time in hour, minute and second using structure. The values of the time is to be taken from the user.
60		7		Structure	7. Enter the marks of 5 students in Chemistry, Mathematics and Physics (each out of 100) using a structure named Marks having elements roll no., name, chem_marks, maths_marks and phy_marks and then display the percentage of each student.
61		8		Structure	8. Write a program to add, subtract and multiply two complex numbers using structures to function.
62 63 64	**Lab 9**	9		Structure	9. Write a structure to store the roll no., name, age (between 11 to 14) and address of students (more than 10). Store the information of the students. - Write a function to print the names of all the students having age 14. - Write another function to print the names of all the students having even roll no. - Write another function to display the details of the student whose roll no is given (i.e. roll no. entered by the user).
65 66		10		Structure	10. Write a structure to store the name, account number and balance of customers (more than 10) and store their information. 1 - Write a function to print the names of all the customers having balance less than $200. 2 - Write a function to add $100 in the balance of all the customers having more than $1000 in their balance and then print the incremented value of their balance.
67		11		Structure	11. Write a program to compare two dates entered by user. Make a structure named Date to store the elements day, month and year to store the dates. If the dates are equal, display "Dates are equal" otherwise display "Dates are not equal".
68 69		12		Structure	12. Write a structure to store the names, salary and hours of work per day of 10 employees in a company. Write a program to increase the salary depending on the number of hours of work per day as follows and then print the name of all the employees along with their final salaries. Hours of work per day 8 Increase in salary $50
70 71 72 73 74 75 76 77		13		Structure	13. Let us work on the menu of a library. Create a structure containing book information like accession number, name of author, book title and flag to know whether book is issued or not. Create a menu in which the following can be done. 1 - Display book information 2 - Add a new book 3 - Display all the books in the library of a particular author 4 - Display the number of books of a particular title 5 - Display the total number of books in the library 6 - Issue a book (If we issue a book, then its number gets decreased by 1 and if we add a book, its number gets increased by 1)
78		1	Primitive Operation	**Stack**	Program for Stack Primitive Operations

79	**Lab 10**	2	Number conversion	***Stack***	*Program for Decimal to Binary Conversion*
80		3	Number conversion	***Stack***	*Program for Decimal to Octal Conversion*
81		4	Number conversion	***Stack***	*Program for Decimal to Hexadecimal Conversion*
82		5	Number conversion	***Stack***	*Program for Decimal to Any Base Conversion*
83		6	Expression validity	***Stack***	*Program to check the validity of Parenthesized Arithmetic Expression using Stack*
84		7	Expression validity	***Stack***	*Program to check the validity of Bracketed Arithmetic Expression using Stack*
85		8	Palidrome check	***Stack***	*Program to check if the given number is a palindrome using stacks*
86		9	string reverse	***Stack***	*Program to Reverse the given String using Stack*
87	**Lab 11**	10	Expression interconversion	**Stack**	Program for Postfix Evaluation
88		11	Expression interconversion	**Stack**	Program for Prefix Evaluation
89		12	Expression interconversion	**Stack**	Program for Infix to Postfix Coversion
90		13	Expression interconversion	**Stack**	Program for Infix to Prefix Coversion
91	**Lab 12**	14	Multi stack	***Stack***	Program for implementation of 2 stacks using a single Array
92		15	Maximum	***Stack***	Program for Finding Minimum in the Stack
93		16	Sorting	***Stack***	Program for Sorting of stack
94		17	Multi stack	***Stack***	Program for implementation of Multiple stack in one Array
95	**Lab 13**	1	Linear Queue	**Queue**	Program of Array Implementaion of Linear Queue
96		2	Circular Queue	**Queue**	Program of Array Implementaion of CircularQueue
97		3	DEQUE	**Queue**	Program for ArrayImplementation of Double Ended Queue
98	**Lab 14**	4	Priority Queue	**Queue**	Program for Array Implementation of Priority Queue (Ascending Array)
99		5	Priority Queue	**Queue**	Program for Array Implementation of Priority Queue (Descending Array)
100		6	Priority Queue	**Queue**	Program for Heap Implementation of Priority Queue
101		7	Stack using Queue	**Queue**	Program for Stack implementation using Queue
102		8	Queue using Stack	**Queue**	Program for Queue implementation using Stack
103	**Lab 15**	1	Linear Linked List	Linked List	Program for Linear Linked List Primitive operations
104		2	Linear Linked List	Linked List	Program for creation of Linked List header file and test of basic functions through that
105		3	Linear Linked List	Linked List	Program for finding count of Nodes in Linked List
106		4	Linear Linked List	Linked List	Program for concatenation of Linear Linked List
107		5	Linear Linked List	Linked List	Program to implement Linear search.
108		6	Linear Linked List	Linked List	Program to insert an item at any given position in the linked List
109		7	Linear Linked List	Linked List	Program for Creation of Copy of the Linked list
110		8	Linear Linked List	Linked List	Program for counting nodes containing even and odd information.
111	**Lab 16**	9	Linear Linked List	Linked List	*Program for Splitting a Linked List(in-place)*
112		10	Linear Linked List	Linked List	Program for Creation of Ascending Order Linear Linked List
113		11	Linear Linked List	Linked List	*Program for Merging two sorted Linked List/unsoted link list*
114		12	Linear Linked List	Linked List	*Program for Union of two sorted Linked List (consider lists as sets)*
115		13	Linear Linked List	Linked List	*Program for Intersection of two sorted Linked List (consider lists as sets)*
116		14	Linear Linked List	Linked List	*Program for finding difference of two linked list (consider lists as sets)*

117		15	Linear Linked List	Linked List	Program for Symmetric difference of two sorted Linked List (consider lists as sets)
118		16	Linear Linked List	Linked List	Program for Finding the Middle element of a singly linked list in one pass
119		17	Linear Linked List	Linked List	Program to perform Binary Search on the Linked List
120		18	Linear Linked List	Linked List	Program for Reversing the Linear Linked List
121	***Lab 17***	19	Linear Linked List	Linked List	Program to print Linked List contents in reverse order
122		20	Linear Linked List	Linked List	Program for Pair wise swap of elements in linked list
123		21	Linear Linked List	Linked List	Program to find kth node from the last in a single link list
124		22	Linear Linked List	Linked List	*Program for Sorting the Linear Linked List*
125		23	Linear Linked List	Linked List	Program for finding if the given link list is palindrome or not
126		24	Linear Linked List	Linked List	Program to Detect if there is ay cycle in the linked list, starting point of cycle, length of cycle
127		25	Linear Linked List	Linked List	Program for Delete duplicate nodes in the Linked List
128		26	Linear Linked List	Linked List	Program to find the Merging point in the linked list
129	***Lab 18***	27	Linear Linked List	Linked List	*Program for Linked List Implementaion of Priority Queue*
130		28	Linear Linked List	Linked List	Program to arrange the consonats ad vowel nodes of the linked list it in such a way that all the vowels nodes come before the consonats while maintaining the order of their arrival
131		29	Linear Linked List	Linked List	Program for Deletion of all occuraces of x from Linked List
132		30	Linear Linked List	Linked List	Program to Delete kth node from end of a linked list in a single scan and O(n) time
133		31	Linear Linked List	Linked List	Program to find out the addition of two given link list 125+85 =210 1->2->5 8->5
134		***32***	***Linear Linked List***	***Project***	***Program for addition very long numbers using Linked List***
135	**Lab 19**	33	Linear Linked List	Linked List	Program to find out the substraction of two given link list
136		34	Linear Linked List	Linked List	Program for Polynomial Addition using Linked List
137		35	Header Linked List	Linked List	Program for Polynomial subtraction using Linked List
138		36	Linear Linked List	Linked List	Program for Polynomial Multiplication using Linked List
139		37	Linear Linked List	Linked List	Program for Circular Linked List Primitive Operations
140		38	Linear Linked List	Linked List	Program for concatenation of Circular Linked List
141		39	Linear Linked List	Linked List	*Program for reversing the Circular Linked List*
142	**Lab 20**	40	Linear Linked List	Linked List	Program to Modify a Circular Linked List such that each node stores the sum of all nodes except itself
143		41	Linear Linked List	Linked List	Program to remove all Fibonacci Nodes from a Circular Singly Linked List
144		42	Circular Linked List	Linked List	Program for implementation of Josephus Problem
145		43	Circular Linked List	Linked list	Program for Doubly linked list Primitive operations
146		44	LInear Linked List	Linked List	Program for Circular Doubly Linked List Primitive Operations
147	**Lab 21**	45	Linear Linked List	Linked List	Program for Linked List Implementation of Stacks
148		46	Linear Linked List	Linked List	Program for Linked List Implementaion of Queue
149		47	Linear Linked List	Linked List	Program for Linked List implementation of Double Ended Queue
150		48	Linear Linked List	Linked List	Program for implementation of Header Linked List

151	**Lab 22**	1	Binary Tree	Tree	Program for Pre-Order, In-Order, Post-Order Traversal
152		2	Binary Tree	Tree	Recursive Creation of Binary Tree
153		3	Binary Tree	Tree	Program to find Node Count in the Binary Tree
154		4	Binary Tree	Tree	Program to find leaf node Count in the Binary Tree
155		5	Binary Tree	Tree	Program to find count of nodes having 1 child
156		6	Binary Tree	Tree	Program to find count of nodes having 2 children
157		7	Binary Tree	Tree	Program to Find the height of the Binary Tree
158		8	Binary Tree	Tree	write a program or function to find the sum all nodes in a given binary tree.
159		9	Binary Tree	Tree	Program to Find if the given Binary Tree is complete
160		10	Binary Tree	Tree	Program to find if the given Binary Tree is strictly
161	**Lab 23**	11	Binary Tree	Tree	Program for Level Order Traversal
162		12	Binary Tree	Tree	Program for Vertical Traversal
163		13	Binary Tree	Tree	Program for Top View Traversal
164		14	Binary Tree	Tree	Program for Bottom view Traversal
165		15	Binary Tree	Tree	Program for Left View Traversal
166		16	Binary Tree	Tree	Program for Right View Traversal
167	**Lab 24**	17	Binary Tree	Tree	Write a program to create a copy of the given Binary Tree
168		18	Binary Tree	Tree	write a program to delete to entire binary tree.
169		19	Binary Tree	Tree	wrirte a program to check the two given binary tree is identical or not(structure as well as node value)
170		20	Binary Tree	Tree	write a program to find out mirror image of given binary tree.
171		21	Binary Tree	Tree	Program to build the Expression Tree from the given Infix expression
172		**22**	**Binary Tree**	**Project**	***Program for Huffman Coding***
173		**23**	**Binary Tree**	**Project**	**write a program to construct a tree from given traversals.**
174	***Lab 25***	24	Binary Search Tree	Tree	write a program to check if the given tree is BST or not.
175		25	Binary Search Tree	Tree	write a program to implement Insertion and Search operation in BST (Iterative)
176		26	Binary Search Tree	Tree	Program to find the diameter of the Binary Tree (distance between the farthest node)
177		27	Binary Search Tree	Tree	write a program to implement min,max,successor, predesessor in the BST
178		28	Binary Search Tree	Tree	write a program to implement deletion in BST.
179		29	Binary Search Tree	Tree	Write a Program for BST insertion (using Recursion)
180		30	AVL	Tree	write a program to perform insertion operation for AVL tree.
181	**Lab 26**	1	Introduction to C++ STL	STL	Vector, Pair, Queue
182		2	Reading Graph	Graph	Program to read a graph and print the adjacency List
183		3	Reading Graph	Graph	Program to read the adjacency matrix and convert that to Adjacency List
184		4	Traversal	Graph	Program for BFS on a Graph
185		5	Traversal	Graph	Program for DFS on a Graph
186		6	Connected Components	Graph	Program to find the number of connected components in the undirected Graph
187		7	Shortest Path	Graph	Program for Warshall's Algorithm for APSP

188	**Lab 27**	8	Transitive Closure	Graph	Program for Warshall's Algorithm for Transitive Closure
189		9	Transitive Closure	Graph	Program for finding Transitive Closure using Multiplication Method
190	***Lab 28***	1	*Sparse Matrix*	Matrices with shape	*Program for 1-D array implementation of Upper Traingular Sparse Matrix*
191		2	*Sparse Matrix*	Matrices with shape	*Program for 1-D array implementation of Lower Traingular Sparse Matrix*
192		3	*Sparse Matrix*	Matrices with shape	*Program for 1-D array implementation of Tridiagonal Sparse Matrix*
193		4	*Sparse Matrix*	Matrices without shape	*Program for Vector Representation of General Sparse Matrix*
194		5	*Sparse Matrix*	Matrices without shape	*Program For Linked List Implementation of General Sparse Matrix*
195		6	*Sparse Matrix*	Matrices without shape	*Program for Addition of two sparse Matrices*
196		7	*Sparse Matrix*	Matrices without shape	*Program for finding transpose of a sparse Matrix*
197		8	*Sparse Matrix*	Matrices without shape	Program for Multiplication of Sparse Matrix