**-LAB EXPT No 1.**

1. Write a C program to sort given list of n integers into ascending order using selection sort. Use function to sort.
2. Given an array of distinct elements. The task is to find triplets in an array whose sum is zero. Take the array as input. (Triplets Game)

Sample Input

0 -1 2 -3 1

Sample output

0 -1 1

1. -3 1
2. Check whether the given matrix is sparse matrix or not using functions.
3. Implement an iterative Lsearch(….) function to search for an element in an 1D array of type integer using linear search technique.
4. Implement a C program to read, display and to find the product of two matrices using functions with suitable parameters. Check for the compatibility of the input matrices before multiplication.

**LAB EXPT No 2.**

1. Write a C program to implement binary search
2. To copy one string to another using Recursion.
3. To check whether a given String is Palindrome or not, using Recursion
4. Simulate the working of Tower of Hanoi for n disks. Print the number of moves.
5. To implement selection sort using recursion

**LAB EXPT No 3.**

1. Write a program to read n names of different sports and store them using array pointers. Use dynamic memory allocation and deallocation functions. The program should display all the names and deallocate the dynamic memory at the end of the program.
2. Write a function Reverse to reverse the elements of an integer array using pointer. Access the elements of the array using dereference operator. Read the values from the keyboard in main function. Display the result in the main function.
3. Write a function Smallest to find the smallest element in an array using pointer. Create a dynamically allocated array and read the values from keyboard in main. Display the result in the main function.
4. i. Write a C program to Demonstrate passing pointers to a function.
5. Demonstrate Returning pointer from a function.
6. Using pointer to pointer.
7. Implement a C program to read, display and to find the product of two matrices using functions with suitable parameters. Note that the matrices should be created using dynamic memory allocation functions and the elements are accessed using array dereferencing.

**LAB EXPT NO 4.**

* 1. Write a C program to implement a ragged array dynamically.
  2. Implement Complex numbers using structures. Write functions to add, multiply, subtract two complex numbers.
  3. Write a C program to implement the following functions. Use pointers and dynamic memory management functions. To read one Student object where Student is a structure with name, roll number and CGPA as the data members
     1. To display one Student object
     2. To sort an array of Student structures according to the roll number.
     3. DOB {day, month (use pointer ), year},
     4. STU\_INFO {reg\_no, name(use pointer), address},
     5. COLLEGE {college\_name (use pointer), university\_name}

d. Samuel wants to store the data of his employees, which includes the following fields: (i) Name of the employee (ii) Date of birth which is a collection of {day, month, year} (iii) Address which is a collection of {house number, zip code and state}. Write a 'C' program to read and display the data of N employees using pointers to array of structures.

e. Create a structure STUDENT consisting of variables of structures:

where structure types from i to iii are declared outside the STUDENT independently. Show how to read and display member variables of DOB type if pointer variable is created for DOB inside STUDENT and STUDENT variable is also a pointer variable. The program should read and display the values of all members of STUDENT structure.

**LAB EXPT NO 5.**

* 1. a. Write a c program to check if the given parenthesized expression has properly matching open and closing parenthesis.

b. Implement a menu driven program to define a stack of characters. Include push, pop

and display functions. Also include functions for checking error conditions such

as underflow and overflow (ref. figure 1) by defining isEmpty and isFull functions.

Use these function in push, pop and display functions appropriately. Use type defined

structure to define a STACK containing a character array and an integer top. Do not

* 1. use global variables.

c. Convert a given decimal number to binary using stack.

d. Determine whether a given string is palindrome or not using stack.

e. Given an array *arr* with n elements and a number k, k<n. The task is to delete k

elements which are smaller than next element (i.e., we delete arr[i] if arr[i] <

arr[i+1]) or become smaller than next because next element is deleted. Example:

Input: arr[] = {20, 10, 25, 30, 40}, k = 2

Output: 25 30 40

Explanation: First we delete 10 because it follows arr[i] < arr[i+1]. Then we delete 20

* 1. because 25 is moved next to it and it also starts following the condition.

1. **LAB EXPT NO 6.**
2. a. Program for evaluation of postfix expression in C
3. b. Evaluate a given prefix expression using stack.
4. c. Convert an infix expression to prefix.
5. d. Implement two stacks in an array.
6. e. To convert a prefix expression to postfix using stack.

**LAB EXPT NO 7**.

* 1. Implement a queue of integers. Include functions insertq, deleteq and displayq
  2. Implement a circular queue of Strings using structures. Include functions insertcq, deletecq and displaycq.
  3. Implement two circular queues of integers in a single array where first queue will run from 0 to N/2 and second queue will run from N/2+1 to N-1 where N is the size of the array.