Lab	Practical
1	**1.** Write a program to find factorial of a number. (Using Loop) **2.** Write a program to find factorial of a number. (Using Recursion) **3.** Write a program to check whether a number is prime or not. (Home Work)
2	**4.** Read n numbers in an array then read two different numbers, replace 1st number with 2nd number in an array and print its index and final array. **5.** Read two 2x2 matrices and perform addition of matrices into third matrix and print it. **6.** Read two matrices, first 3x2 and second 2x3, perform multiplication operation and store result in third matrix and print it. (Home Work)
3	**7.** Write a program to swap two numbers using user-defines method. **8.** Create class Employee_Detail with attributes Employee_id, Name, Designation, and Salary. Write a program to read the detail from user and print it. **9.** Create array of object of class Student_Detail with attributes Enrollment_no, Name, Sem, CPI for 5 students, scan their information and print it. (Home Work)
4	**10.** Implement a program for stack that performs following operations using array: PUSH, POP, PEEP, CHANGE & DISPLAY **11.** Write a program to determine if an input character string is of the form a^ib^i where i >= 1 i.e. Number of 'a' should be equal to number of 'b'. (Home Work)
5	**12.** Implement a program to convert in-fix notation to post-fix notation using stack.
6	**13.** Write a program for evaluation of post-fix Expression using Stack. **14.** Write a program for evaluation of pre-fix Expression using Stack. (Home Work)
7	**15.** Implement Simple Queue using array that performs following operations: INSERT, DELETE, DISPLAY **16.** Implement Circular Queue using array that performs following operations: INSERT, DELETE, DISPLAY (Home Work)
8	**17.** Implement a program to create a node for singly linked list. Read the data in a node, print the node. **18.** Write a menu driven program to implement following operations on the singly linked list. a. Insert a node at the front of the linked list. b. Display all nodes. c. Delete a first node of the linked list. d. Insert a node at the end of the linked list. (Home Work) e. Delete a last node of the linked list. (Home Work) f. Delete a node from specified position. (Home Work) **19.** Write a program to implement stack using singly linked list. (Home Work) **20.** Write a program to implement queue using singly linked list. (Home Work)
9	**21.** Write a menu driven program to implement following operations on the circular linked list. a. Insert a node at the front of the linked list. b. Delete a node from specified position. c. Insert a node at the end of the linked list. (Home Work) d. Display all nodes. (Home Work)

Lab	Practical
10	**22.** Write a menu driven program to implement following operations on the doubly linked list. a. Insert a node at the front of the linked list. b. Delete a node from specified position. c. Insert a node at the end of the linked list. (Home Work) d. Display all nodes. (Home Work)
11	**23.** Write a program to implement Linear/Sequential Search. **24.** Write a program to implement Binary Search.
12	**25.** Read n numbers in an array from user and sort them in ascending order and print sorted array using bubble sort algorithm. **26.** Read n numbers in an array from user and sort them in ascending order and print sorted array using insertion sort algorithm. **27.** Read n numbers in an array from user and sort them in ascending order and print sorted array using selection sort algorithm. (Home Work)
13	**28.** Read n numbers in an array from user and sort them in ascending order and print sorted array using bucket sort algorithm. **29.** Read n numbers in an array from user and sort them in ascending order and print sorted array using radix sort algorithm.
14	**30.** Read n numbers in an array from user and sort them in ascending order and print sorted array using shell sort algorithm. **31.** Read n numbers in an array from user and sort them in ascending order and print sorted array using counting sort algorithm.
15	**32.** Read n numbers in an array from user and sort them in ascending order and print sorted array using tree sort algorithm. **33.** Read n numbers in an array from user and sort them in ascending order and print sorted array using heap sort algorithm.
16	**34.** Read n numbers in an array from user and sort them in ascending order and print sorted array using merge sort algorithm. **35.** Read n numbers in an array from user and sort them in ascending order without using comparison.
17	**36.** Read n numbers in an array from user and sort them in ascending order and print sorted array using quick sort algorithm.