DATA STRUCTURES ASSIGNMENT 1

**Theory.**

1. Define data structure, what is advantage of defining a proper data structure, Give difference between linear and non-linear data structure with example.
2. Define structures, what are the advantages of structure data type where they are used give example of storing of nodes in data structure.
3. What is the pointer data type and what does it signify?
4. Define a linked list and explain using examples. Explain various terminologies of linked list like node, start \*, malloc, dynamic memory allocation and garbage collection. Explain algorithm and methods for insertion and deletion in singly, circular and doubly linked lists.
5. Define stack. Why is stack known as LIFO (last in first out)? With respect to Stack,  
   explain following:  
   •Push  
   •Pop  
   •Popup Stack  
   •Overflow  
   •Underflow

Write functions for Push and Pop in Stack using Array.

1. With examples explain the method of Infix to Postfix 10 examples and Postfix evaluation.
2. Explain method of postfix evaluation using stack with example.
3. Explain the method of infix to postfix using stack with an example.
4. Define linear queue: - List application. Explain front, rear, overflow, and underflow.
5. Explain limitations of linear queue, then how it can be resolved through circular queue.
6. Write an algorithm for insertion and deletion in a circular queue.
7. Define priority queue and give its applications in the operating system.
8. Explain basic terminology of trees and define their types.
9. For the given expressions, draw the expression tree.
10. What do you mean by tree traversal? Write inorder, preorder, postorder traversal for the given tree. (3 trees minimum)
11. For the given inorder and preorder traversal, find the tree. (3 minimum )
12. For the given expression, construct an expression tree and prove that the preorder and postorder traversal of the tree is equal to prefix & postfix expression of the given expression.

**Practical**

1. Implement singly linked list and perform all insertion & deletion in singly linked list, count no of nodes in singly linked list, Count no of odd and even in singly linked list.
2. Perform all insertion and deletion in singly circular linked list.
3. Perform all insertion and deletion in a doubly linked list.
4. Perform stack with push and pop function.
5. Write a program for postfix evaluation.
6. Write a program for Infix to postfix.
7. Program to find factorial of a number through recursion.
8. Program to print fibonacci series through recursion.
9. Program for reversing a string and checking whether it is palindrome or not.
10. Program to access an element from the queue, multiple it by 5 and insert it back in the queue.
11. Program for implementing linear queue using array and linked list.
12. Program for implementing circular queues using arrays.