1. Implement Stack ADT using arrays.
2. Convert an infix expression to postfix expression using Stack ADT
3. Evaluate postfix expression using Stack ADT
4. Implement Queue ADT using arrays
5. Implement Circular Queue ADT using arrays
6. Implement following operations of Singly Linked List ADT
    a. Create Linked List with n nodes
    b. Insert at beginning
    c. Insert at end
    d. Insert before specified node
    e. Display (Forward Traversal)
7. Implement following operations of Singly Linked List ADT
    a. Create Linked List with n nodes
    b. Delete at beginning
    c. Delete at end
    d. Delete before specified node
    e. Display (Forward Traversal)
8. Implement following operations of Singly Linked List ADT
    a. Create Linked List with n nodes
    b. Insert before specified node
    c. Count No of Nodes
    d. Sort nodes in ascending order
    e. Display (Forward Traversal)
9. Implement following operations of Singly Linked List ADT
    a. Create Linked List with n nodes
    b. Insert at end
    c. Count No of Nodes
    d. Search an element
    e. Display (Forward Traversal)
10. Implement following operations of Circular Linked List ADT
    a. Create list with n nodes
    b. Insert at beginning
    c. Insert at end
    d. Display (Forward Traversal)
11. Implement following operations of Circular Linked List ADT
    a. Create list with n nodes
    b. Delete at beginning
    c. Delete at end
    d. Display (Forward Traversal)
12. Implement following operations of Circular Linked List ADT
    a. Create list for n nodes
    b. Insert at beginning
    c. Delete at end
    d. Count no of nodes

    e. Display (Forward Traversal)

13. Implement Stack ADT using Linked List
14. Implement Queue ADT using Linked List
15. Implement following operations of Binary Search Tree using Linked List
    a. Insertion
    b. Searching
    c. Traversal (Preorder)
    d. Count total no of nodes
16. Implement following operations of Binary Search Tree using Linked List
    a. Insertion
    b. Deletion
    c. Traversal (Inorder)
17. Implement following operations of Binary Search Tree using Linked List
    a. Insertion
    b. Traversal (Postorder)
    c. Count total no of internal nodes
    d. Count total no of leaf nodes
    e. Height of the tree
18. Implement following operations of Binary Search Tree using Linked List
    a. Insertion
    b. Traversal (Preorder, Inorder, Postorder)
    c. Height of the tree
19. Implement Graph Traversal Technique: DFS
20. Implement Graph Traversal Technique: BFS
21. Implement Hashing using array. Demonstrate Linear Probing to handle collision.
22. Implement Hashing using array. Demonstrate Quadratic Probing to handle collision.