DATA STRUCTURES
LAB- II

1. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)
   a. Push an Element on to Stack
   b. Pop an Element from Stack
   c. peep(i)// return ith element from stack
   d. peek()//return topmost element of the stack
   e. Demonstrate Overflow and Underflow situations on Stack
   f. display() //displays all elements of the stack
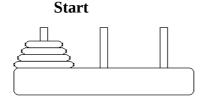   g. Exit
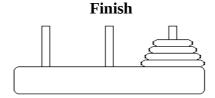Support the program with appropriate functions for each of the above operations

2. Repeat program :1 using Linked list, on employee data, where as every NODE in Linked list stores **Emp_Id**(String), **Emp_Name**(String), **Emp_Address**(String), **Employee_Age**(integer), **Employee_Salary**(Float).
Note : Include extra operaton for searching and displaying complete data of a Employee, corresponding to the given Emp_Id.

3. Implement a Program in C using Stack for the following operations.
   a. Check a Given number is Palindrome or not.
   b. Check a given string is Palindrome or not.

4. Given a game board with three pegs and a set of disks of different diameter all stacked from smallest to largest on the leftmost peg, move all of the disks to the rightmost peg following these two rules.
a. Only one disk may be moved at a time.
b. A larger diameter disk may never be placed on a smaller disk.
Output the set of moves along with total number of moves for a given 'n' number of disks.

Start   **Finish**

5. Design, Develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, %(Remainder), ^ (Power).

6. Design, Develop and Implement a Program in C for evaluating Postfix Expression.

7. Design, Develop and Implement a menu driven Program in C for the following operations on QUEUE of Characters (Array Implementation of Queue with maximum size MAX)
   a. Insert an Element on to QUEUE
   b. Delete an Element from QUEUE
   c. Demonstrate Overflow and Underflow situations on QUEUE
   d. Display all elements of QUEUE
   e. Exit
Support the program with appropriate functions for each of the above operations

8. Implement program: 7 using Linked lists, on employee data, where as every NODE in Linked list stores **Emp_Id**(String), **Emp_Name**(String), **Emp_Address**(String), **Employee_Age**(integer), **Employee_Salary**(Float).
Note : Include extra operaton for searching and displaying complete data of a Employee, corresponding to the given Emp_Id.

9. Implement Program: 7 using Circular Queue's.

10. Implement Program: 8 using Circular Queue's.