5. Design, Develop and Implement a Program in C for the following Stack Applications:

a) Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %,^.
b) Solving Tower of Hanoi problem with n disks.

Program: a)

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

#define STK_SIZE 10

void fnPush(int [], int*, int);

int fnPop(int [], int*);

int main()

{

        int iaStack[50], i, iOp1, iOp2, iRes;

        char acExpr[50], cSymb;

        int top = -1;

        printf("nEnter a valid postfix expressionn");

        scanf("%s", acExpr);

        for(i=0; i<strlen(acExpr); i++)

        {

                cSymb = acExpr[i];

                if(isdigit(cSymb))
```

```
{
        fnPush(iaStack, &top, cSymb-'0');
}
else
{
        iOp2 = fnPop(iaStack, &top);
        iOp1 = fnPop(iaStack, &top);
        switch(cSymb)
        {
                case '+' :      iRes = iOp1 + iOp2;
                                        break;
                case '-' :      iRes = iOp1 - iOp2;
                                        break;
                case '*' :      iRes = iOp1 * iOp2;
                                        break;
                case '/' :      iRes = iOp1 / iOp2;
                                        break;
                case '%' :      iRes = iOp1 % iOp2;
                                        break;
                case '^' :      iRes = (int)pow(iOp1 , iOp2);
                                        break;
        }
        fnPush(iaStack, &top, iRes);
```

```
                }
        }
        iRes = fnPop(iaStack, &top);
        printf("nValue of %s expression is %d\n", acExpr, iRes);
        return 0;
}
void fnPush(int Stack[], int *t , int elem)
{
        *t = *t + 1;
        Stack[*t] = elem;
}
int fnPop(int Stack[], int *t)
{
        int elem;
        elem = Stack[*t];
        *t = *t -1;
        return elem;
}
```

Program: b) Solving Tower of Hanoi problem with n disks.

```
#include <stdio.h>

void towers(int, char, char, char);

int main()

{

   int num;

   printf("Enter the number of disks : ");

   scanf("%d", &num);

   printf("The sequence of moves involved in the Tower of Hanoi are :\n");

   towers(num, 'A', 'C', 'B');

   return 0;

}

void towers(int num, char frompeg, char topeg, char auxpeg)

{

   if (num == 1)

   {

      printf("\nMove disk 1 from peg %c to peg %c", frompeg, topeg);

      return;

   }

   towers(num - 1, frompeg, auxpeg, topeg);

   printf("\nMove disk %d from peg %c to peg %c", num, frompeg, topeg);

   towers(num - 1, auxpeg, topeg, frompeg);

}
```