```c
/*
    Tittle:-Implement a program to create stacks by using switch case by implementing function
    using PUSH,POP,PEEK AND DISPLAY operation
    Name:- TAUSEEF MUSHTAQUE ALI SHAIKH
    Roll No.;- 18CO63
    Class:- S.Y. [CO]
    Date:- 22-07-2019
    Discription:- In This Program a menu based implementation for a stack is held for some operations such as PUSH, PEEK, POP, DISPLAY using fumctionsmade for an individual action to be performed
*/


#include <stdio.h>
#define MAX 5

struct stack
{
    int data[MAX];
    int top;
};

void initialize(struct stack *s)
{
    s->top=-1;
}


int isFull(struct stack *s)
{
    if (s->top >= (MAX-1))
    {
        printf("Error Can't insert!");
        return(1);
    }
    else
    {
        return(0);
    }
}

int push(struct stack *s,int num)
{
    if (s->top >= (MAX-1))
    {
        return(0);
    }
    else
    {
        s->top++;
        s->data[s->top]=num;
        return(1);
    }

}
```

```

int isEmpty(struct stack *s)
{
    if (s->top==-1)
    {
        return(1);
    }
    else
    {
        return(0);
    }
}

int Pop(struct stack *s)
{
    if (s->top==-1)
    {
        return(0);

    }
    else
    {
        int d;
        d=s->data[s->top];
        s->top--;
        return(1);
    }

}

void Display(struct stack *s)
{
    int i;
    if(s->top==-1)
    {
        printf("\n\t Stack is empty!");
    }
    else
    {
        printf("\n\t Stack content is: ");
        for(i=s->top;i>=0;i--)
        {
            printf("\t%d",s->data[i]);
        }
    }
}


void Peek(struct stack *s)
{
    if(s->top==-1)
    {
        printf("\n\t Stack is empty!");
    }
    else
    {
        printf("\n\t Stack content top-most element: ");

            printf("%d\t",s->data[s->top]);
    }
}



int main()
{
    int n,num;
    struct stack s;

    initialize(&s);
    while(1)
    {
        printf("\n\n\t\t\tMenu");
        printf("\n1.Push\n2.Pop\n3.Peek\n4.Display\n0.Exit");
        printf("\nEnter a choice: ");
        scanf("%d",&n);
        switch(n)
        {
            case 1:
                printf("\n\tWelcome in push operation!");
                printf("\n\tEnter a element to push: ");
                scanf("%d",&num);
                if (push(&s,num))//1= true
                {
                    printf("\n\tPushing of element perform successfully");
                }
                else
                {
                    printf("\n\tStack is full cannot perform PUSH OPERATION!");
                }

                break;

            case 2:
                printf("\n\tWelcome in pop operation!");

                if (Pop(&s))
                {
                    printf("\n\tPoping of element perform successfully");
                }
                else
                {
                    printf("\n\tStack is empty OPERATION!");
                }

                break;

            case 3:
                Peek(&s);
                break;

            case 4:
                Display(&s);
                break;

            case 0:
                printf("\n\tExiting the program");
                return(0);
                break;

            default:
                printf("\n\tInput error\n\tEnter a valid no.\n");
        }
    }
}

```