```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);
    }

}
```

```
54
55  int isEmpty(struct stack *s)
56  {
57      if (s->top==-1)
58      {
59          return(1);
60      }
61      else
62      {
63          return(0);
64      }
65  }
66
67  int Pop(struct stack *s)
68  {
69      if (s->top==-1)
70      {
71          return(0);
72
73      }
74      else
75      {
76          int d;
77          d=s->data[s->top];
78          s->top--;
79          return(1);
80      }
81
82  }
83
84  void Display(struct stack *s)
85  {
86      int i;
87      if(s->top==-1)
88      {
89          printf("\n\t Stack is empty!");
90      }
91      else
92      {
93          printf("\n\t Stack content is: ");
94          for(i=s->top;i>=0;i--)
95          {
96              printf("\t%d",s->data[i]);
97          }
98      }
99  }
100
101
102 void Peek(struct stack *s)
103 {
104     if(s->top==-1)
105     {
106         printf("\n\t Stack is empty!");
107     }
108     else
109     {
```

```
110         printf("\n\t Stack content top-most element: ");
111 
112             printf("%d\t",s->data[s->top]);
113     }
114 }
115 
116 
117 
118 int main()
119 {
120     int n,num;
121     struct stack s;
122 
123     initialize(&s);
124     while(1)
125     {
126         printf("\n\n\t\t\tMenu");
127         printf("\n1.Push\n2.Pop\n3.Peek\n4.Display\n0.Exit");
128         printf("\nEnter a choice: ");
129         scanf("%d",&n);
130         switch(n)
131         {
132             case 1:
133                 printf("\n\tWelcome in push operation!");
134                 printf("\n\tEnter a element to push: ");
135                 scanf("%d",&num);
136                 if (push(&s,num))//1= true
137                 {
138                     printf("\n\tPushing of element perform successfully");
139                 }
140                 else
141                 {
142                     printf("\n\tStack is full cannot perform PUSH OPERATION!");
143                 }
144 
145                 break;
146 
147             case 2:
148                 printf("\n\tWelcome in pop operation!");
149 
150                 if (Pop(&s))
151                 {
152                     printf("\n\tPoping of element perform successfully");
153                 }
154                 else
155                 {
156                     printf("\n\tStack is empty OPERATION!");
157                 }
158 
159                 break;
160 
161             case 3:
162                 Peek(&s);
163                 break;
164 
165             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");
        }
    }
}
```