

11.STACK OPERATIONS :-

Code:-

```
#include<stdio.h>
#include<process.h>
#include<stdlib.h>
#define MAX 5 //Maximum number of elements that can be stored
int top=-1,stack[MAX];
void push();
void pop();
void display();
void main()
{
int ch;
while(1) //infinite loop, will end when choice will be 4
{
printf("\n** Stack Menu **");
printf("\n\n1.Push\n2.Pop\n3.Display\n4.Exit");
printf("\n\nEnter your choice(1-4):");
scanf("%d",&ch);
switch(ch)
{
case 1: push();
break;
case 2: pop();
break;
case 3: display();
break;
case 4: exit(0);
default: printf("\nWrong Choice!!");
}
}
```

```
}  
  
void push()  
{  
    int val;  
    if(top==MAX-1)  
    {  
        printf("\nStack is full!!");  
    }  
    else  
    {  
        printf("\nEnter element to push:");  
        scanf("%d",&val);  
        top=top+1;  
        stack[top]=val;  
    }  
}  
  
void pop()  
{  
    if(top==--1)  
    {  
        printf("\nStack is empty!!");  
    }  
    else  
    {  
        printf("\nDeleted element is %d",stack[top]);  
        top=top-1;  
    }  
}  
  
void display()  
{  
    int i;
```

```

if(top== -1)
{
printf("\nStack is empty!!");
}
else
{
printf("\nStack is...\n");
for(i=top; i>=0; --i)
printf("%d\n", stack[i]);
}
}

```

OUTPUT:-

```

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

Enter your choice(1-4):1

Enter element to push:4

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

Enter your choice(1-4):1

Enter element to push:5

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

Enter your choice(1-4):3

Stack is...
5
4

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

```

```

Enter your choice(1-4):2

Deleted element is 5

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

Enter your choice(1-4):5

Wrong Choice!!

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

Enter your choice(1-4):3

Stack is...
4

** Stack Menu **

1.Push
2.Pop
3.Display
4.Exit

Enter your choice(1-4):

```