

```
StackSim.c
Saved

1 #include<stdio.h>
2 #include<conio.h>
3 #define StackSize 5
4 int top=-1;
5 int s[10];
6 int item;
7 void push()
8 {
9     if(top==StackSize-1)
10    {
11        printf("Stack Overflow\n");
12        return;
13    }
14    top=top+1;
15    s[top]=item;
16 }
17 int pop()
18 {
19     if(top==-1)
20         return -1;
21     return s[top--];
22 }
23 void display()
24 {
25     int i;
26     if(top==-1)
27     {
28         printf("Stack Is Empty\n");
29         return;
30     }
31     printf("Contents of the stack\n");
32     for(i=0;i<=top;i++)
33         printf("%d\n",s[i]);
34 }
35 void main()
36 {
37     int ItemDel;
38     int ch;
39     for(;;)
40     {
41         printf("\n1.Push\n2.Pop\n3.Display\n4.Exit\n");
42         printf("Enter the choice\n");
43         scanf("%d",&ch);
44         switch(ch)
45         {
46             case 1:
47                 printf("Enter the item to be inserted\n");
48                 scanf("%d",&item);
49                 push();
50                 break;
51             case 2:
52                 ItemDel=pop();
53                 if(ItemDel==-1)
54                     printf("Empty Stack\n");
55                 else
56                     printf("Item deleted:%d\n",ItemDel);
57                 break;
58             case 3:
59                 display();
60                 break;
61             default:exit(0);
62         }
63     }
64 }
65
```





```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
22

1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
33

1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
44

1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
55

1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
66

1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
77
Stack Overflow
```



```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
23
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
25
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
2
Item deleted:25
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
2
Item deleted:23
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
2
Empty Stack
```



```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
99
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
80
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
1
Enter the item to be inserted
76
```

```
1.Push
2.Pop
3.Display
4.Exit
Enter the choice
3
Contents of the stack
99
80
76
```

23/9/20

IBM19C5090

RAHIL

2) Program to simulate working of stack using an array. a) Pop b) Display c) Push. The prg. should print appropriate msg for stack overflow and underflow.

Code:

```
#include <stdio.h>
#include <conio.h>
#define StackSize 5
```

```
int top = -1;
```

```
int s[10];
```

```
int item;
```

```
void push()
```

```
{
```

```
if (top == StackSize - 1)
```

```
{
    printf("Stack overflow\n");
    return;
```

```
}
```

```
top = top + 1;
```

```
s[top] = item;
```

```
}
```

```
int pop()
```

```
{
    if (top == -1)
```

```
{
printf("Stack is Empty\n");
    return -1; return s[top--];
}
```

```
}
```

```
void display()
```

```
{
```

```
int i;
```

```
if (top == -1)
```



```

{
    printf("Stack is Empty\n");
    return;
}
printf("Contents of stack\n");
for (i = 0; i <= top; i++)
    printf("%d\n", s[i]);
}

```

```

void main()
{

```

```

    int ItemDel;

```

```

    int ch;

```

```

    for (;;)
    {

```

```


```

```

printf("Enter the item to

```

```

printf("Enter the item to
    printf("\n1. Push\n2. Pop\n3. Display\n4. Exit\n");

```

```

    printf("Enter the choice\n");

```

```

    scanf("%d", &ch);

```

```

    switch (ch)
    {

```

```


```

```

        case 1:

```

```

            printf("Enter the item to be inserted\n");

```

```

            scanf("%d", &item);

```

```

            push();

```

```

            break;

```

```

        case 2:

```

```

            ItemDel = pop();

```

```

            if (ItemDel == pop -1)

```

```

                printf("Empty stack\n");

```

```

            else

```

```

                printf("Item deleted: %d\n", ItemDel);

```

```

            break;

```

case 3:

display();

break;

default : exit(0);

}

}

}