



new\*



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #define MAX 3
4 int top=-1,stack[MAX];
5 void push();
6 void pop();
7 void display();
8
9 int main()
10 {
11     int ch;
12     while(1)
13     {
14         printf("\n\n1.PUSH\n2.POP\n3.DISPLAY\n4.
EXIT");
15         printf("\n\nENTER YOUR CHOICE(1-4:");
16         scanf("%d",&ch);
17
18         switch(ch)
19         {
20             case 1:push();
21                 break;
22
23             case 2: pop();
24                 break;
25             case 3: display();
26                 break;
27             case 4: exit(0);
28             default: printf("\nWrong choice");
29         }
30     }
31     return 0;
32 }
33 void push()
34 {
35     int val;
36     if(top==MAX-1)
37
38     {
39         printf("\n STACK OVERFLOW");
40     }
```

Tab

{

}

:

;

"



new\*



```
38     {
39     printf("\n STACK OVERFLOW");
40     }
41     else
42     { printf("\n ENTER ELEMENTS TO PUSH:");
43     scanf("%d",&val);
44     top=top+1;
45     stack[top]=val;
46     }
47     }
48     void pop()
49     {
50     if(top== -1)
51
52     { |
53     printf("\n STACK UNDERFLOW");
54     }
55     else
56     { printf("\n DELETED ELEMENT IS %d",
stack[top]);
57     top=top-1;
58     }
59     }
60     void display()
61     { int i;
62     if(top== -1)
63     { printf("\n STACK IS EMPTY");
64     }
65     else
66     { printf("\n STACK IS...");
67     for(i=top; i>=0; --i)
68     printf("%d\n", stack[i]);
69     }
70     }
```





TAB



1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT

ENTER YOUR CHOICE(1-4):1

ENTER ELEMENTS TO PUSH:10

1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT

ENTER YOUR CHOICE(1-4):1

ENTER ELEMENTS TO PUSH:20

1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT

ENTER YOUR CHOICE(1-4):1

ENTER ELEMENTS TO PUSH:30

1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT

ENTER YOUR CHOICE(1-4):1

STACK OVERFLOW

1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT

ENTER YOUR CHOICE(1-4):3

STACK IS...30

20  
10

1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT

ENTER YOUR CHOICE(1-4):2

DELETED ELEMENT IS 30

1.PUSH  
2.POP  
3.DISPLAY  
4.EXIT



TAB



ENTER ELEMENTS TO PUSH:30

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):1

STACK OVERFLOW

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):3

STACK IS...30

20  
10

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):2

DELETED ELEMENT IS 30

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):2

DELETED ELEMENT IS 20

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):2

DELETED ELEMENT IS 10

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):2

STACK UNDERFLOW

- 1.PUSH
- 2.POP
- 3.DISPLAY
- 4.EXIT

ENTER YOUR CHOICE(1-4):

## Lab Program - 1

1. Write a program to simulate the working of stack using an array with the following Push, Pop, & display. The program should print appropriate message for stack underflow & overflow

```
→ #include <stdio.h>
#include <stdlib.h>
```

```
#define MAX 3
```

```
int top = -1, stack[MAX];
```

```
void push();
```

```
void pop();
```

```
void display();
```

```
int main()
```

```
{
```

```
int ch;
```

```
while(1)
```

```
{
```

```
printf("In 1. PUSH In 2. POP In 3. DISPLAY In 4. EXIT\n");
```

```
printf("In Enter Your choice (1-4): ");
```

```
scanf("%d", &ch);
```

~~switch~~

switch(ch)

{

case 1: push();  
break;

case 2: pop();  
break;

case 3: display();  
break;

case 4: ~~exit~~ exit(0);

default: printf("In wrong choice");

{

}

return 0;

}

void push()

{

int val;

if (top == MAX-1)

{ printf("In Stack ~~to~~ ~~full~~ overflow");  
}

else

```
{ printf ("In Enter Elements to Push:");  
scanf ("%d", &val);  
top = top + 1;  
stack[top] = val;  
}
```

}

void pop()

{

```
if (top == -1)  
{ printf ("In Stack Is Empty");  
}
```

else

```
{ printf ("In Deleted Element Is %d", stack[top]);  
top = top - 1;  
}
```

void display()

{

int i;

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

```
printf ("In Stack Underflow");  
}
```

else {

```

printf("\n stack is : ");
for (p = top; p >= 0; p --p)
    printf("%d", stack[p]);
}
}

```

Output

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 1

ENTER ELEMENTS TO PUSH: 10

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 1

ENTER ELEMENTS TO PUSH: 20

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 1

ENTER ELEMENTS TO PUSH: 30

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 1

STACK OVERFLOW

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 3

STACK IS ... 30

20

10

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 2

DELETED ELEMENT IS 30

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 2

DELETED ELEMENT IS 20

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 2

DELETED ELEMENT IS 10

1. PUSH
2. POP
3. DISPLAY
4. EXIT

ENTER YOUR CHOICE (1-4): 2

STACK UNDERFLOW