

Operations for the stack:

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

Enter choice: 1

Integer to push: 1

Operations for the stack:

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

Enter choice: 1

Integer to push: 2

Operations for the stack:

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

Enter choice: 1

Integer to push: 3

Stack Overflow

Operations for the stack:

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

Enter choice: 3

Element: 2

Operations for the stack:

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

Enter choice: 2

the item on the top was 2

Operations for the stack:

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

Enter choice: 2

the item on the top was 1

Operations for the stack:

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

Enter choice: 2

Stack underflow

```
1  //this is shrihari viswanahan program'
2
3  #include <stdio.h>
4
5  #define N 2
6
7  int stack[N];
8  int top = -1;
9
10 void push();
11 void pop();
12 void peek();
13
14 int main()
15 {
16     int i = 1;
17     while(i == 1){
18         printf("\n");
19         printf("\n");
20
21         printf("Operations for the stack:\n");
22         printf("1. Push\n");
23         printf("2. Pop\n");
24         printf("3. Peek\n");
25         printf("4. Exit\n");
26
27         int choice = 0;
28         printf("Enter choice: ");
29         scanf("%d", &choice);
30
31         switch(choice){
32             case 1: push();
33                 break;
34
35             case 2: pop();
36                 break;
37
38             case 3: peek();
39                 break;
40
41             case 4: i = 0;
42                 break;
43         }
44     }
45
46
47 }
48
```

```
49 void push(){
50     int value = 0;
51     printf("Integer to push: ");
52     scanf("%d", &value);
53
54     if (top == N - 1){
55         printf("Stack Overflow");
56     }
57
58     else{
59         top++;
60         stack[top] = value;
61     }
62 }
63
64
65
66 void pop(){
67     if (top == -1){
68         printf("Stack underflow");
69     }
70
71     else{
72         int item = stack[top];
73         printf("the item on the top was %d", item);
74         top--;
75     }
76 }
77
78
79 void peek(){
80
81     if (top == -1){
82         printf("Underflow");
83     }
84
85     else{
86         printf("Element: %d", stack[top]);
87     }
88 }
```

## # Stack Lab - 01:-

Code:-

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

```
#define
```

```
void main()
```

```
{  
    int stack[
```

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

```
#define N 2
```

```
int stack[N];
```

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

```
void push();
```

```
void pop();
```

```
void peek();
```

```
int main()
```

```
{
```

```
    int i = 1;
```

```
    while (i == 1) {
```

```
        printf("\n");
```

```
        printf("\n");
```

```
        printf("Operations for the stack\n");
```

```
        printf("1. Push\n");
```

```
        printf("2. Pop\n");
```

```
        printf("3. Peek\n");
```

```
        printf("4. Exit\n");
```

```
int choice = 0;
printf("Enter choice: ");
scanf("%d", &choice);
```

```
switch (choice) {
    case 1: push();
            break;
    case 2: pop();
            break;
    case 3: peek();
            break;
    case 4: i = 0;
            break;
}
```

```
}
```

```
void push() {
```

```
    int value = 0;
```

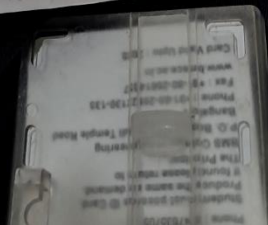
```
    printf("Integer to push: ");
```

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

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

```
        printf("Stack Overflow");
```

```
    }
```



```

else {
    top++;
    stack[top] = value;
}
}

void pop() {
    if (top == -1) {
        printf("Stack Underflow");
    }
    else {
        printf("The item popped: %d", stack[top]);
        top--;
    }
}

void peek() {
    if (top == -1) {
        printf("Stack underflow");
    }
    else {
        printf("Element on top is %d", stack[top]);
    }
}

```

### Operations for the stack

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

Enter choice: 1

Integer to push: 1

### Operations for the stack

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

Enter choice: 1

Integer to push: 2

### Operations for the stack

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

Enter choice: 1

Integer to push: 3

Stack Overflow.

### Operation for the stack

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

Enter Choice: 3

Element on top is 3

### Operations for the stack

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

Enter choice: 2

The item popped: 2

### Operations for the stack

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

Enter choice: 2

The item popped: 1

### Operations for the stack

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

Enter Choice: 2

Stack underflow

Mg  
29/9/25

Operations for the stack:

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

Enter choice: 1

Integer to push: 1

Operations for the stack:

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

Enter choice: 1

Integer to push: 2

Operations for the stack:

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

Enter choice: 1

Integer to push: 3

Stack Overflow

Operations for the stack:

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

Enter choice: 3

Element: 2

Operations for the stack:

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

Enter choice: 2

the item on the top was 2

Operations for the stack:

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

Enter choice: 2

the item on the top was 1

Operations for the stack:

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

Enter choice: 2

Stack underflow