

```
Start here X first lab program.c X
1  #include<stdio.h>
2  #include<stdlib.h>
3  #define SIZE 10
4
5  void push(int);
6  void pop();
7  void peek();
8  void display();
9
10 int stack[SIZE], top = -1;
11
12 int main(){
13     int value, choice;
14     while(1){
15         printf("*** MENU***");
16         printf("\n 1. push\n 2. pop\n 3. peek\n 4. Display\n 5. Exit ");
17         printf("\nEnter your choice: ");
18         scanf("%d",&choice);
19         switch(choice){
20             case 1:
21                 printf("Enter the value to push: ");
22                 scanf("%d",&value);
23                 push(value);
24                 break;
25             case 2:
26                 pop();
27                 break;
28             case 3:
29                 peek();
30                 break;
31             case 4:
32                 display();
33                 break;
34             case 5:
35                 exit(0);
36             default:
37                 printf("\nWrong selection!!! Try again!!!\n");
38         }
39     }
40     return 0;
}
```

```
Start here X first lab program.c X
41 }
42
43 void push(int value){
44     if (top == SIZE - 1){
45         printf("\nThe stack is full!!! Insertion is not possible\nStack overflow\n");
46     } else {
47         top++;
48         stack[top] = value;
49         printf("Insertion success\n");
50     }
51 }
52
53 void pop(){
54     if (top == -1){
55         printf("\nStack is empty, deletion not possible\nStack underflow\n");
56     } else {
57         printf("\nDeleted: %d\n", stack[top]);
58         top--;
59     }
60 }
61
62 void peek(){
63     if(top == -1){
64         printf("Stack underflow\n");
65     } else {
66         printf("%d\n", stack[top]);
67     }
68 }
69
70 void display(){
71     if (top == -1){
72         printf("\nStack is empty\n");
73     } else {
74         int i;
75         printf("\nStack elements are:\n");
76         for(i = top; i >= 0; i--){
77             printf("%d ", stack[i]);
78         }
79     }
80 }
```

```
"C:\c dsa\first lab program.e" X + v
*** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice: 1
Enter the value to push: 2
Insertion success
*** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice: 1
Enter the value to push: 3
Insertion success
*** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice: 4

Stack elements are:
3 2 *** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice:

Stack elements are:
3 2 *** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice: 2

Deleted: 3
*** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice: 3
2
*** MENU***
1. push
2. pop
3. peek
4. Display
5. Exit
Enter your choice: 5

Process returned 0 (0x0)    execution time : 146.315 s
Press any key to continue.
|
```

1) Write a program to simulate the working of stack using an array using with the following. 29/09/25

- a) Push
- b) Pop
- c) Peek
- d) Display

The program should print appropriate message for stack overflow, stack underflow.

```
#include <stdio.h>
#include <conio.h>
#define SIZE 10

void push(int);
void pop();
void peek();
void display();

int stack[SIZE], top = -1;

void main()
{
    int value, choice;
    clrscr();
    while (1)
    {
        printf(" *** MENU *** \n");
        printf(" 1. Push  2. POP  3. Peek  4. Display  5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);
        switch (choice)
        {
            case 1: printf("Enter the value to be inserted: ");
                    scanf("%d", &value);
                    push(value);
                    break;
            case 2: pop();
                    break;
            case 3: peek();
                    break;
            case 4: display();
                    break;
            case 5: exit(0);
        }
    }
}
```

```

    default: printf("\n Wrong selection!!! Try Again!!!");
}
}

Void push(int value) {
    if (top == size-1)
        printf("\n Stack is Full!!! Insertion is not possible\n stack overflow");
    else {
        top++;
        stack[top] = value;
        printf("\n Insertion Success!");
    }
}

Void pop() {
    if (top == -1)
        printf("\n Stack is empty, deletion not possible\n stack underflow");
    else {
        printf("\n Deleted: %d", stack[top]);
        top--;
    }
}

Void peek() {
    if (top == -1)
        printf("\n underflow");
    else
        printf("\n %d", stack[top]);
}

Void display() {
    if (top == -1)
        printf("\n stack is empty");
    else {
        int i;

```

```

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

```

### Output

```

*** MEN
1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice
Enter the value to be inserted
*** MEN
1. Push
2. Pop
3. Peek
4. Display
5. Exit
Enter your choice
Enter the value to be inserted
*** MEN
1. push
2. pop
3. peek
4. Display
5. Exit
Enter the value to be inserted
Stack is empty
32 ***
1. Push
2. Pop
3. peek
4. Display
5. Exit
Enter your choice
Deleted

```



```

again!!!"); printf("\nStack elements are: \n");
for (i=top; i>0; i--)
    printf("%d\n", stack[i]);
    ?
    }

```

### Output

\*\*\* MENU \*\*\*

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

Enter your choice: 1

Enter the value to push: 2  
Insertion Success.

\*\*\* MENU \*\*\*

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

Enter your choice: 1

Enter the value to push: 3  
Insertion Success

\*\*\* MENU \*\*\*

1. push
2. pop
3. peek
4. Display.
5. Exit

Enter your choice: 4

Stack elements are:

32 \*\*\* MENU \*\*\*

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

Enter your choice: 2

Deleted: 3

\*\*\* MENU \*\*\*

1. push
2. pop
3. peek
4. display
5. exit

Enter your choice: 2

\*\*\* MENU \*\*\*

1. push
2. pop
3. peek
4. display
5. exit

Enter your choice: 5

06/10

WAP to  
arithm  
pressio  
the b  
and

# inclu  
# incl  
# def

Char  
int

Void

Char

64 in  
##