

```

1  #include<stdio.h>
2  #include<stdlib.h>
3  #define N 100
4  int stack[N];
5  int top=-1;
6  void display();
7  void push(){
8      int l;
9      printf("enter element to be inserted\n");
10     scanf("%d",&l);
11     if(top==N-1){printf("Stack overflow");}
12     else{
13         top++;
14         stack[top]=l;
15         printf("%d is pushed into stack\n",stack[top]);
16         display();
17     }
18 }
19 void pop(){
20     if(top== -1){printf("Stack underflow");}
21     else{
22         int item =stack[top];
23         top--;
24         printf("%d is popped out of stack\n",item);
25         display();
26     }
27 }
28 void peek(){
29     if(top== -1){printf("Stack underflow");}
30     else{
31         printf("top element in stack is:%d",stack[top]);
32     }
33 }
34 void display(){
35     if(top== -1){printf("empty stack");}
36     else{
37         printf("\nElements in the stack are:\t");
38         for(int i=top;i>=0;i--){
39             printf("%d\t",stack[i]);
40         }
41     }
42 }

```

```

43 int main() {
44     int ch;
45     while(1) {
46         printf("\n1.Push\t2.Pop\t3.Peek\t4.Display\t5.Exit\n");
47         printf("Enter stack operation:\n");
48         scanf("%d",&ch);
49         switch(ch) {
50             case 1:
51                 push();
52                 break;
53             case 2:
54                 pop();
55                 break;
56             case 3:
57                 peek();
58                 break;
59             case 4:
60                 display();
61                 break;
62             case 5:
63                 exit(0);
64             default:
65                 printf("Invalid choice");
66                 break;
67         }
68     }
69     return 0;
70 }
71

```

```

1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
1
enter element to be inserted
12
12 is pushed into stack

Elements in the stack are:      12
1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
1
enter element to be inserted
2
2 is pushed into stack

Elements in the stack are:      2      12
1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
2
2 is popped out of stack

Elements in the stack are:      12
1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
3
top element in stack is:12
1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
2
12 is popped out of stack
empty stack
1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
2
Stack underflow
1.Push  2.Pop  3.Peek  4.Dislpay      5.Exit
Enter stack opertaion:
5

Process returned 0 (0x0)   execution time : 367.217 s
Press any key to continue.

```

@ write a program to Stimulate the working of stack using an array with the following.

a) Push b) pop c) Display

Code:-

```
#include <stdio.h>
#include <stdlib.h>
#define N 50
int stack[N];
int top = -1;

void push() {
    int l;
    printf("Enter Elements to be inserted\n");
    scanf("%d", &l);
    if (top == N-1) { printf("Stack overflow"); }
    else {
        top++;
        stack[top] = l;
        printf("%d is popped out of stack\n", item);
        display();
    }
}

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

void pop() {
    if (top == -1) { printf("Stack underflow"); }
    else {
        int item = stack[top];
        top--;
        printf("%d is popped out of stack\n", item);
        display();
    }
}

void display() {
    if (top == -1) { printf("empty stack"); }
    else {
        printf("\n Elements in the stack are:");
        for (int i = top; i >= 0; i--) {
            printf("%d\t", stack[i]);
        }
    }
}
```

```

// main program
int main() {
    int ch;
    while(1) {
        printf("1.push 2.pop 3. peek 4. display 5. Exit\n");
        printf("Enter stack operation elm: ");
        scanf("%d", &ch);
        switch(ch) {
            case 1:
                push();
                break;
            case 2:
                pop();
                break;
            case 3:
                peek();
                break;
            case 4:
                display();
                break;
            case 5:
                exit(0);
        }
        default:
            printf("Invalid choice\n");
            break;
    }
    return 0;
}

```

o/p:- 1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 1
 Enter element to be inserted:
 12
 12 is pushed into stack
 elements in the stack are: 12

1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 1
 Enter element to be inserted:
 2
 2 is pushed into stack

Elements in stack are: 2 12
 1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 2

2 is popped out of stack

Elements in stack are: 12

1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 2

Pop Element in stack is: 12

1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 2

12 is popped out of stack
 Empty stack

1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 2

stack underflow

1.push 2.pop 3. peek 4. display 5. Exit
 Enter stack operation:
 5

MS
 27/9/25