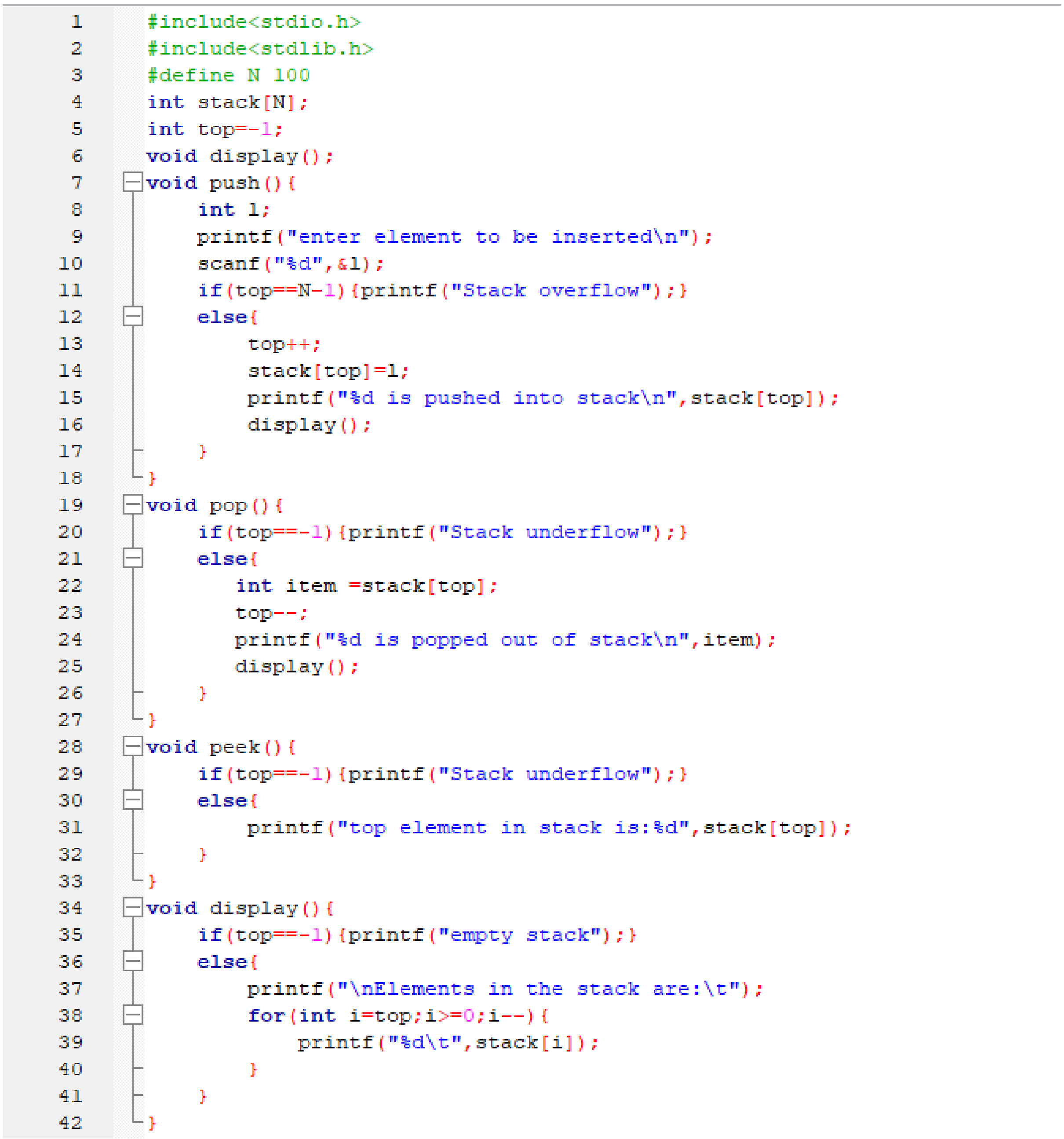
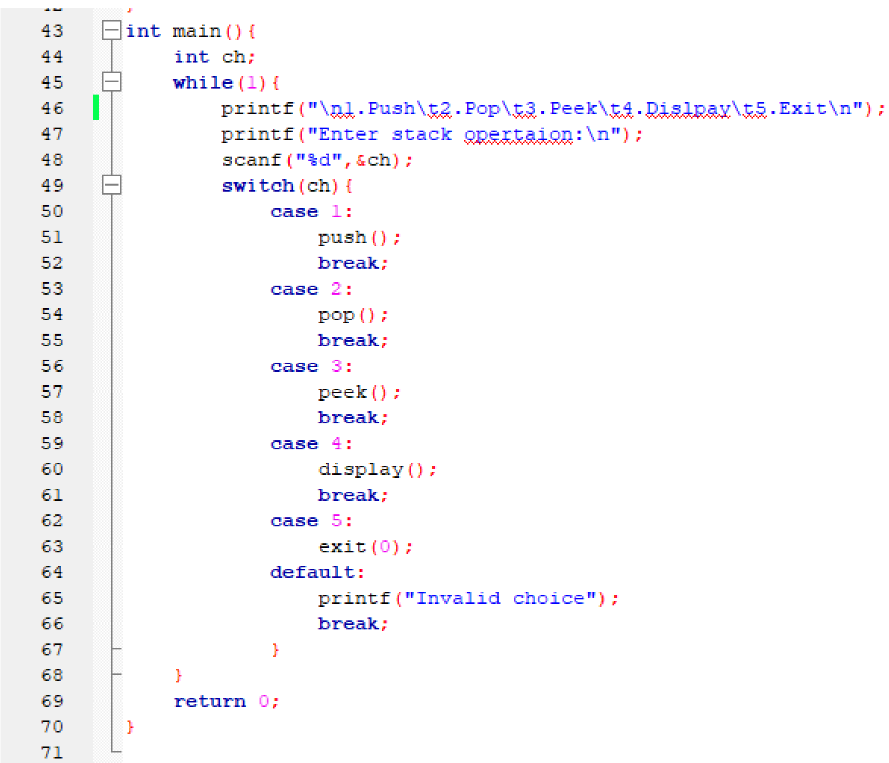
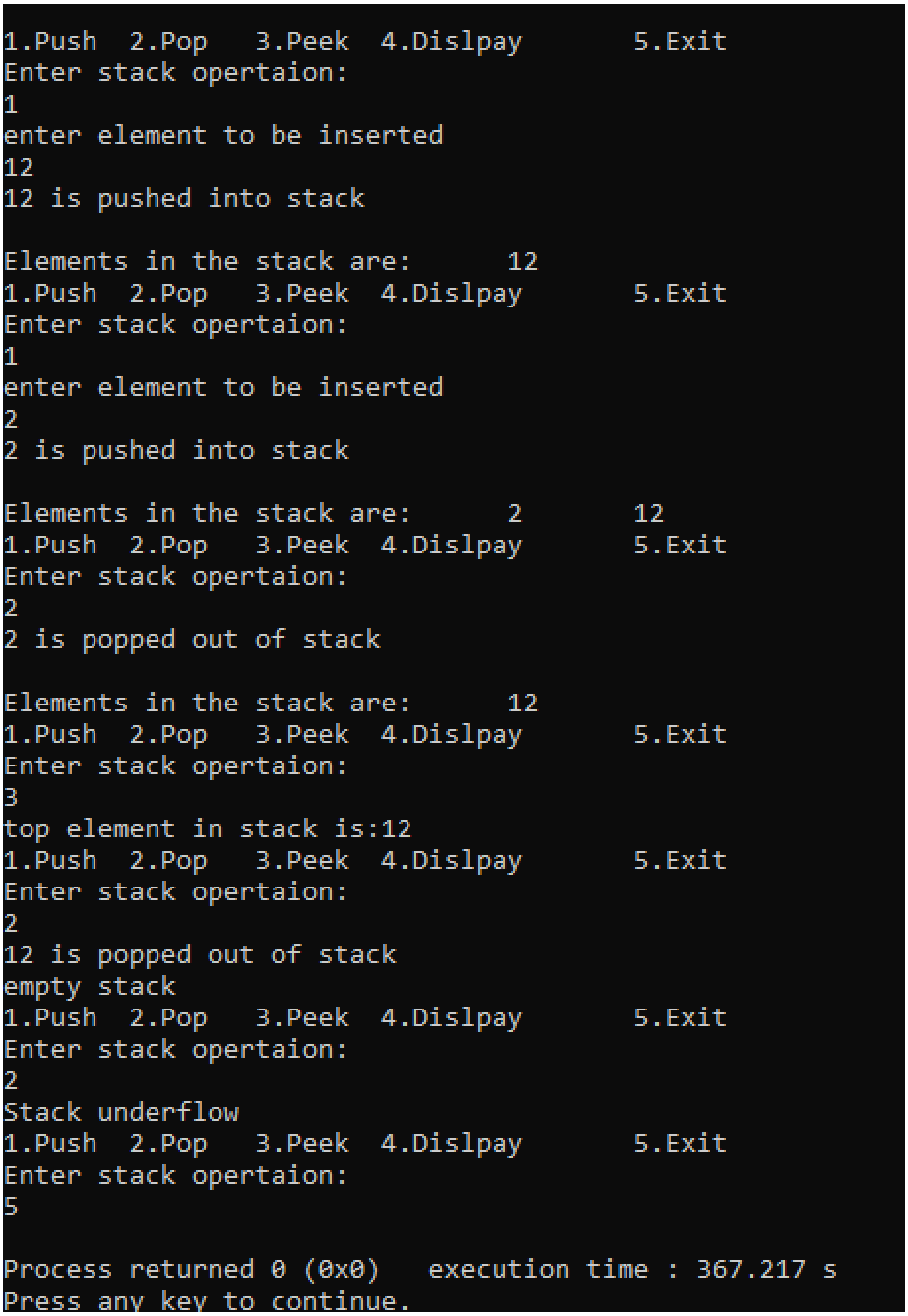


```c
#include<stdio.h>
#include<stdlib.h>
#define N 100
int stack[N];
int top=-1;
void display();
void push(){
    int l;
    printf("enter element to be inserted\n");
    scanf("%d",&l);
    if(top==N-1){printf("Stack overflow");}
    else{
        top++;
        stack[top]=l;
        printf("%d is pushed into stack\n",stack[top]);
        display();
    }
}
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 peek(){
    if(top==-1){printf("Stack underflow");}
    else{
        printf("top element in stack is:%d",stack[top]);
    }
}
void display(){
    if(top==-1){printf("empty stack");}
    else{
        printf("\nElements in the stack are:\t");
        for(int i=top;i>=0;i--){
            printf("%d\t",stack[i]);
        }
    }
}
```



```c
int main(){
    int ch;
    while(1){
        printf("\n1.Push\t2.Pop\t3.Peek\t4.Dislpay\t5.Exit\n");
        printf("Enter stack opertaion:\n");
        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");
                break;
            }
    }
    return 0;
}

```



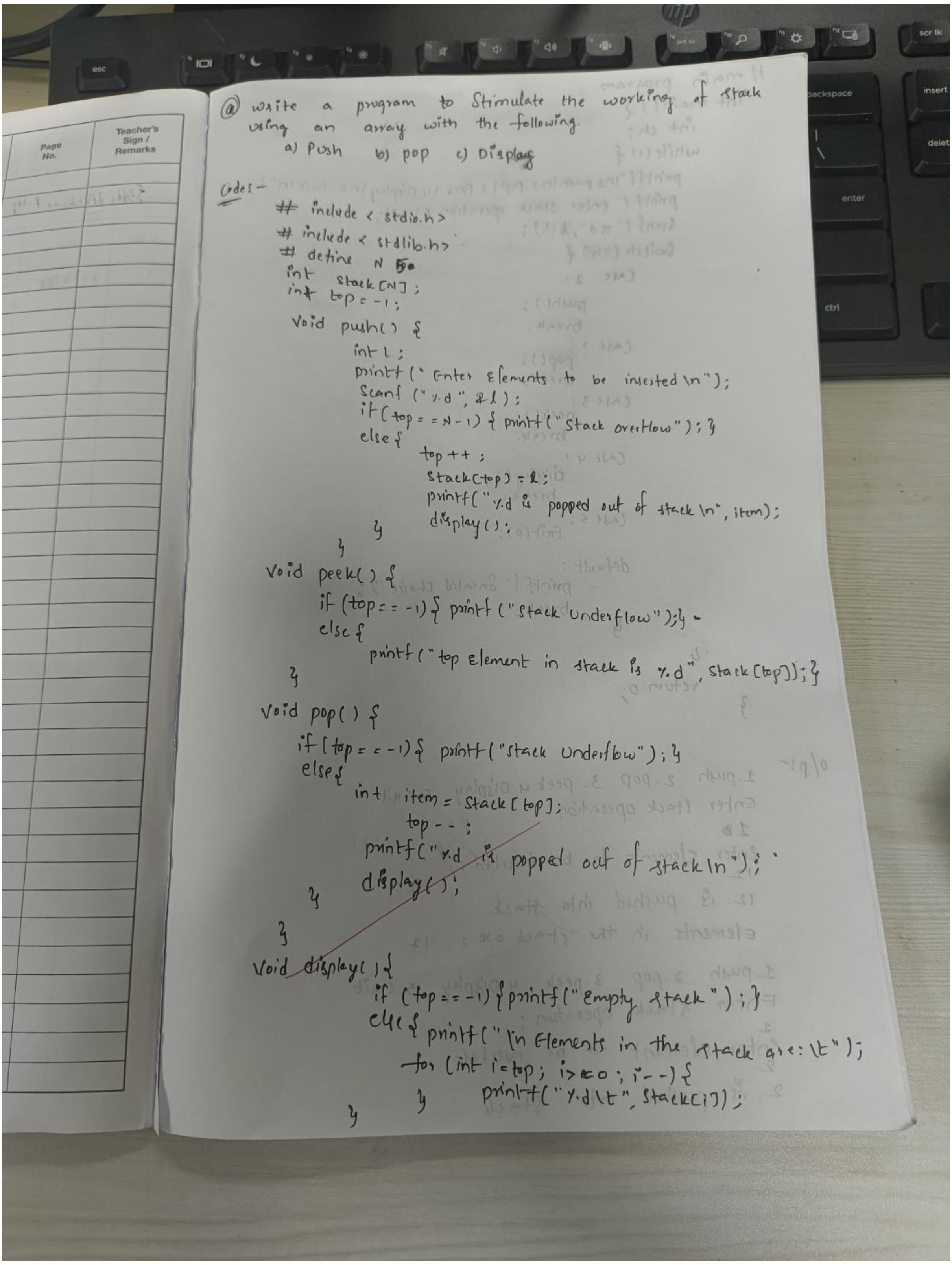
```
1.Push  2.Pop   3.Peek  4.Dislpay       5.Exit
Enter stack opertaion:
1
enter element to be inserted
12
12 is pushed into stack

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.
```



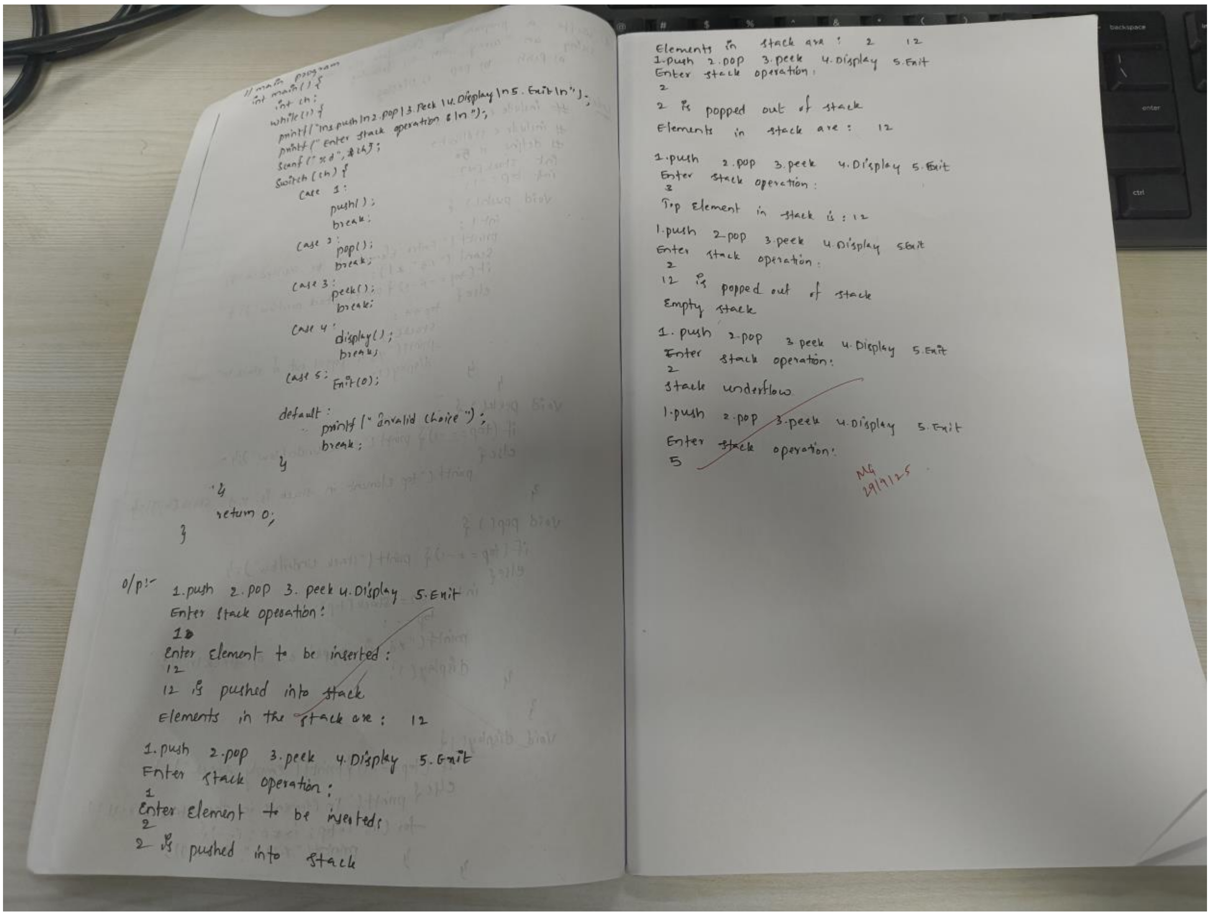
Page No.	Teacher's Sign / Remarks

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

a) Push b) pop c) Display

Codes -

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



```
// main program
int main() {
    int ch;
    while(1) {
    printf("\n1.push\n2.pop\3.Peek\4.Display\n5.Exit\n");
    printf("Enter stack operation &\n");
    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");
            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:
3
Top 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
```

MG
29/9/25