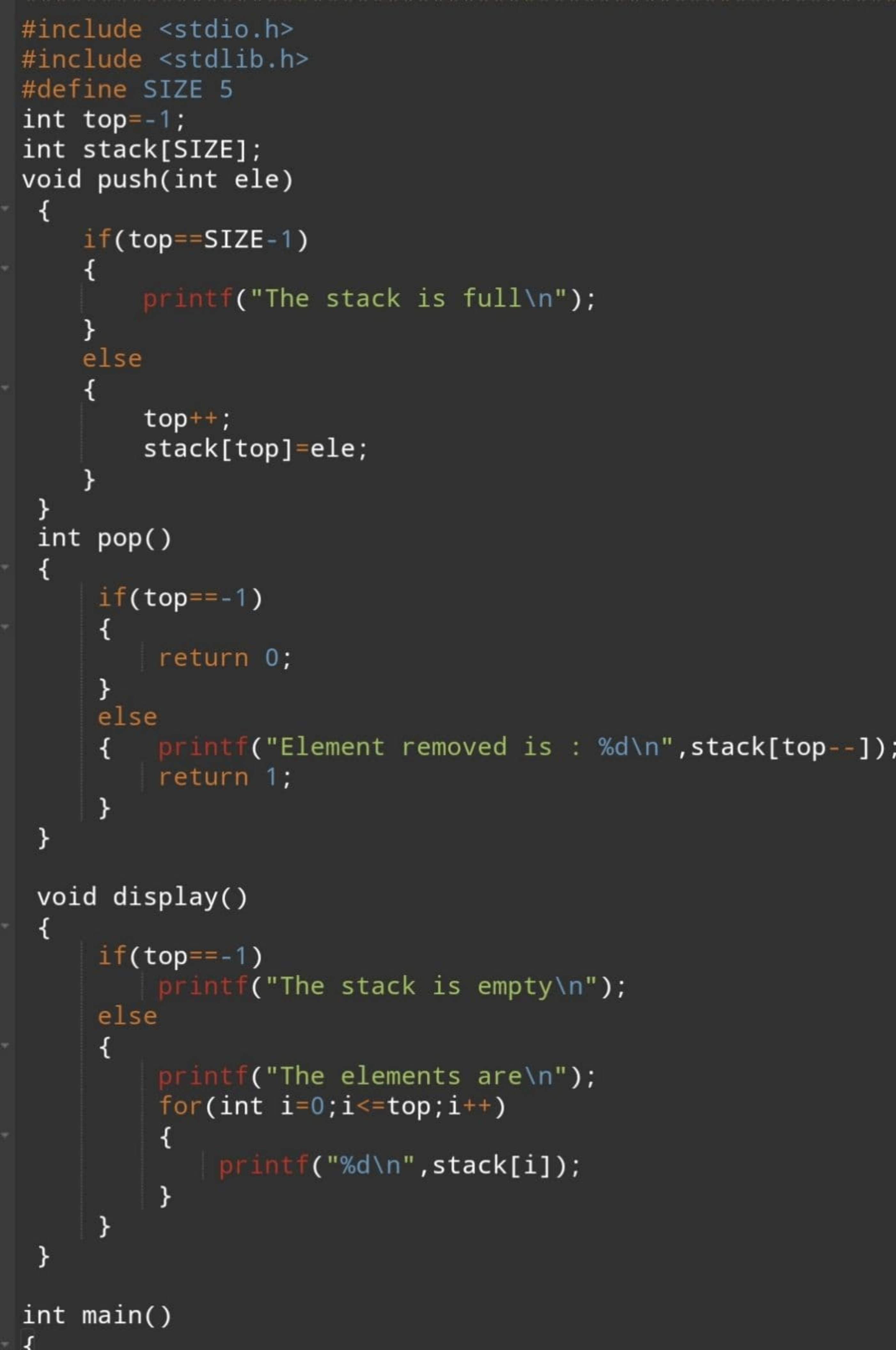
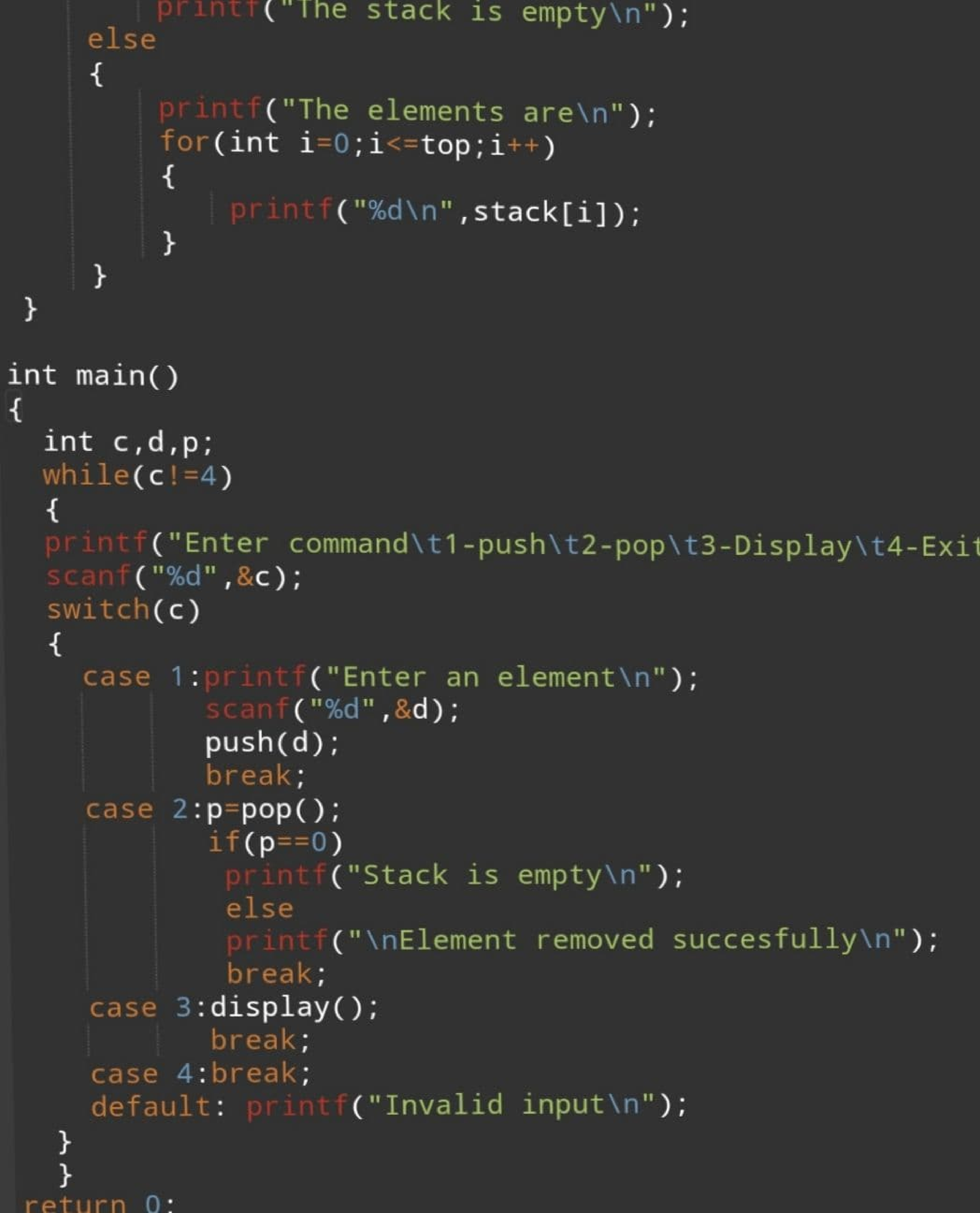
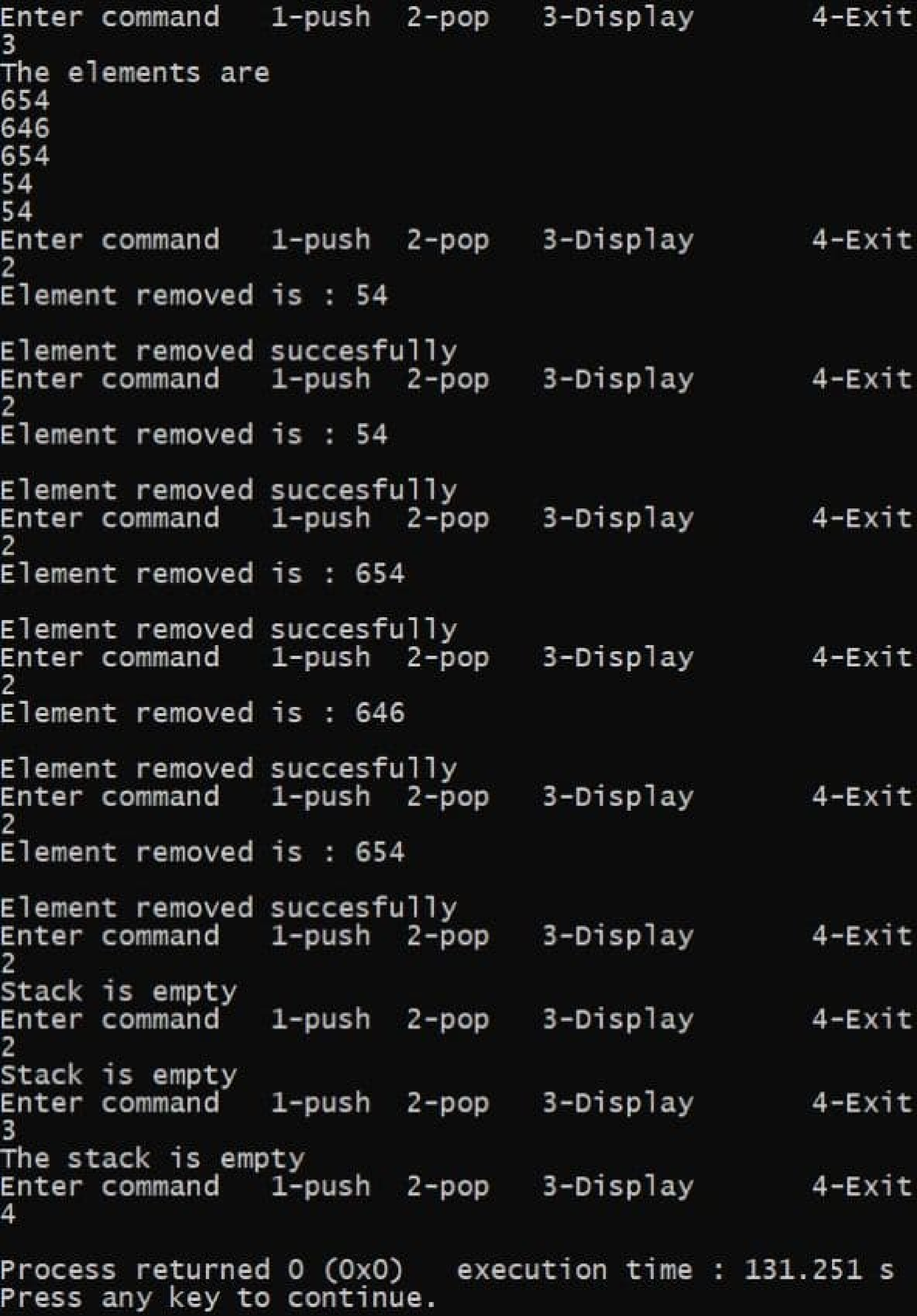


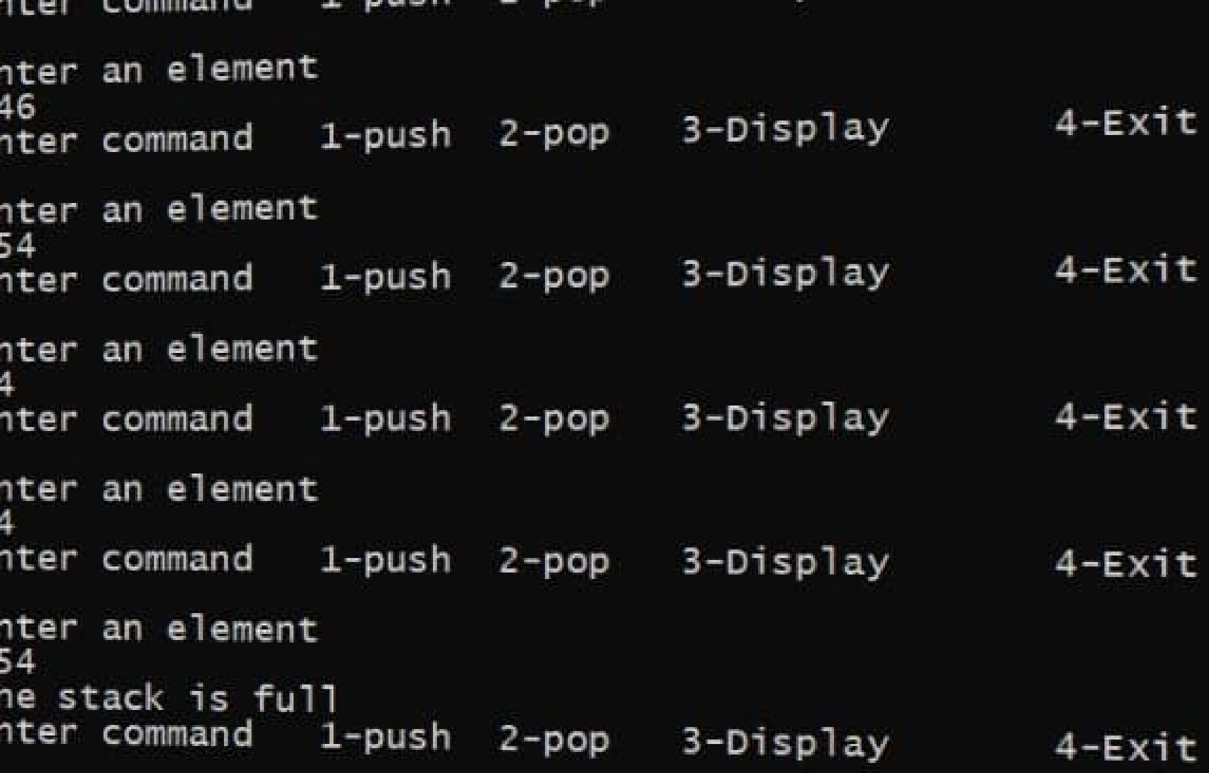
```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 5
int top=-1;
int stack[SIZE];
void push(int ele)
 {
    if(top==SIZE-1)
    {
        printf("The stack is full\n");
    }
    else
    {
        top++;
        stack[top]=ele;
    }
 }
 int pop()
 {
     if(top==-1)
     {
         return 0;
     }
     else
     {   printf("Element removed is : %d\n",stack[top--]);
         return 1;
     }
 }

 void display()
 {
     if(top==-1)
         printf("The stack is empty\n");
     else
     {
         printf("The elements are\n");
         for(int i=0;i<=top;i++)
         {
             printf("%d\n",stack[i]);
         }
     }
 }

int main()
{
```



```
            printf("The stack is empty\n");
        else
        {
            printf("The elements are\n");
            for(int i=0;i<=top;i++)
            {
                printf("%d\n",stack[i]);
            }
        }
}

int main()
{
    int c,d,p;
    while(c!=4)
    {
    printf("Enter command\t1-push\t2-pop\t3-Display\t4-Exit
    scanf("%d",&c);
    switch(c)
    {
        case 1:printf("Enter an element\n");
               scanf("%d",&d);
               push(d);
               break;
        case 2:p=pop();
               if(p==0)
                printf("Stack is empty\n");
                else
                printf("\nElement removed succesfully\n");
                break;
        case 3:display();
               break;
        case 4:break;
        default: printf("Invalid input\n");
    }
    }
 return 0;
```



```
Enter command   1-push  2-pop   3-Display       4-Exit
3
The elements are
654
646
654
54
54
Enter command   1-push  2-pop   3-Display       4-Exit
2
Element removed is : 54

Element removed succesfully
Enter command   1-push  2-pop   3-Display       4-Exit
2
Element removed is : 54

Element removed succesfully
Enter command   1-push  2-pop   3-Display       4-Exit
2
Element removed is : 654

Element removed succesfully
Enter command   1-push  2-pop   3-Display       4-Exit
2
Element removed is : 646

Element removed succesfully
Enter command   1-push  2-pop   3-Display       4-Exit
2
Element removed is : 654

Element removed succesfully
Enter command   1-push  2-pop   3-Display       4-Exit
2
Stack is empty
Enter command   1-push  2-pop   3-Display       4-Exit
2
Stack is empty
Enter command   1-push  2-pop   3-Display       4-Exit
3
The stack is empty
Enter command   1-push  2-pop   3-Display       4-Exit
4

Process returned 0 (0x0)   execution time : 131.251 s
Press any key to continue.
```



```
nter command   1-push  2-pop   3-Display        4-Exit

nter an element
46
nter command   1-push  2-pop   3-Display        4-Exit

nter an element
54
nter command   1-push  2-pop   3-Display        4-Exit

nter an element
4
nter command   1-push  2-pop   3-Display        4-Exit

nter an element
4
nter command   1-push  2-pop   3-Display        4-Exit

nter an element
54
he stack is full
nter command   1-push  2-pop   3-Display        4-Exit
```

```
#include <stdio.h>
#include <stdlib.h>
#define SIZE 5
int top=-1;
int stack [SIZE];
void push(int ele)
{
 if (top == SIZE - 1)
 {
 printf ("The stack is full \n")
 }
 else
 {
   top++
   stack [top] = ele;
   }
}
int pop()
 {
 if (top==-1)
 {
 return 0;
 }
else
{ printf(" Element removed is : %d \n", stack[t
return 1;
}
}
void display()
{
if (top==-1)
printf(" The stack is empty \n");
else
 {
 printf(" The eliments are \n");
 for (int i=0; i<=top; i++)
  {
   printf("%d \n", stack [i]);
   }
 }
}
int main()
{
int c, d, p;
while (c!=4)
{
printf(" Enter command \t1-push \t2-pop
\t3-Display \t4-Exit \n");
scanf ("%d", &c);
```

```
switch(c)
 {
 case 1: printf("Enter an element \n");
  scanf("%d", &d);
  push(d);
   break;
case 2: p = pop();
 if (p==0)
 printf(" stack is empty \n");
  else
 printf("\n Element removed  \n");
     break;
 case 3: display();
        break;
  case 4: break;
            tf (" Invalid input \n");
```