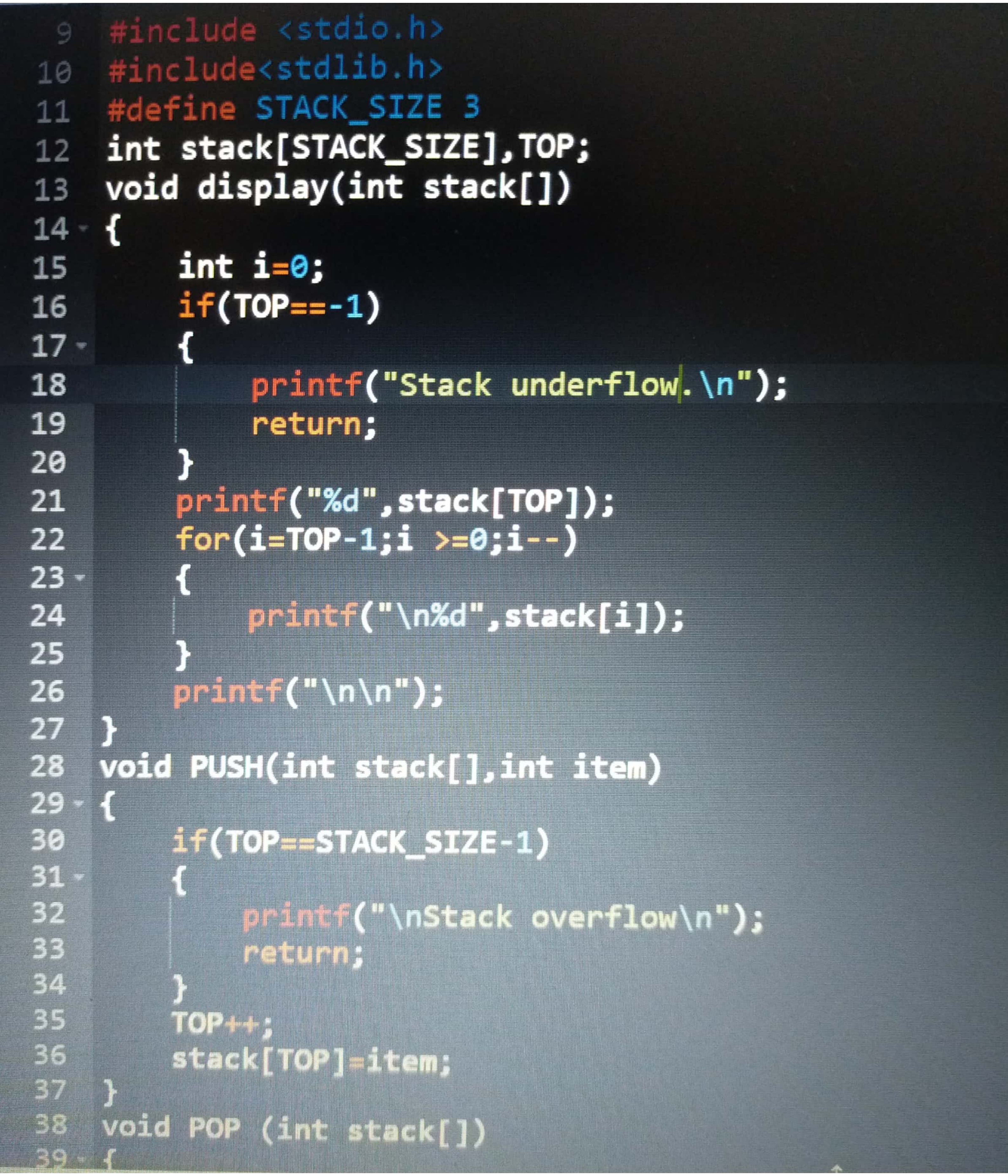
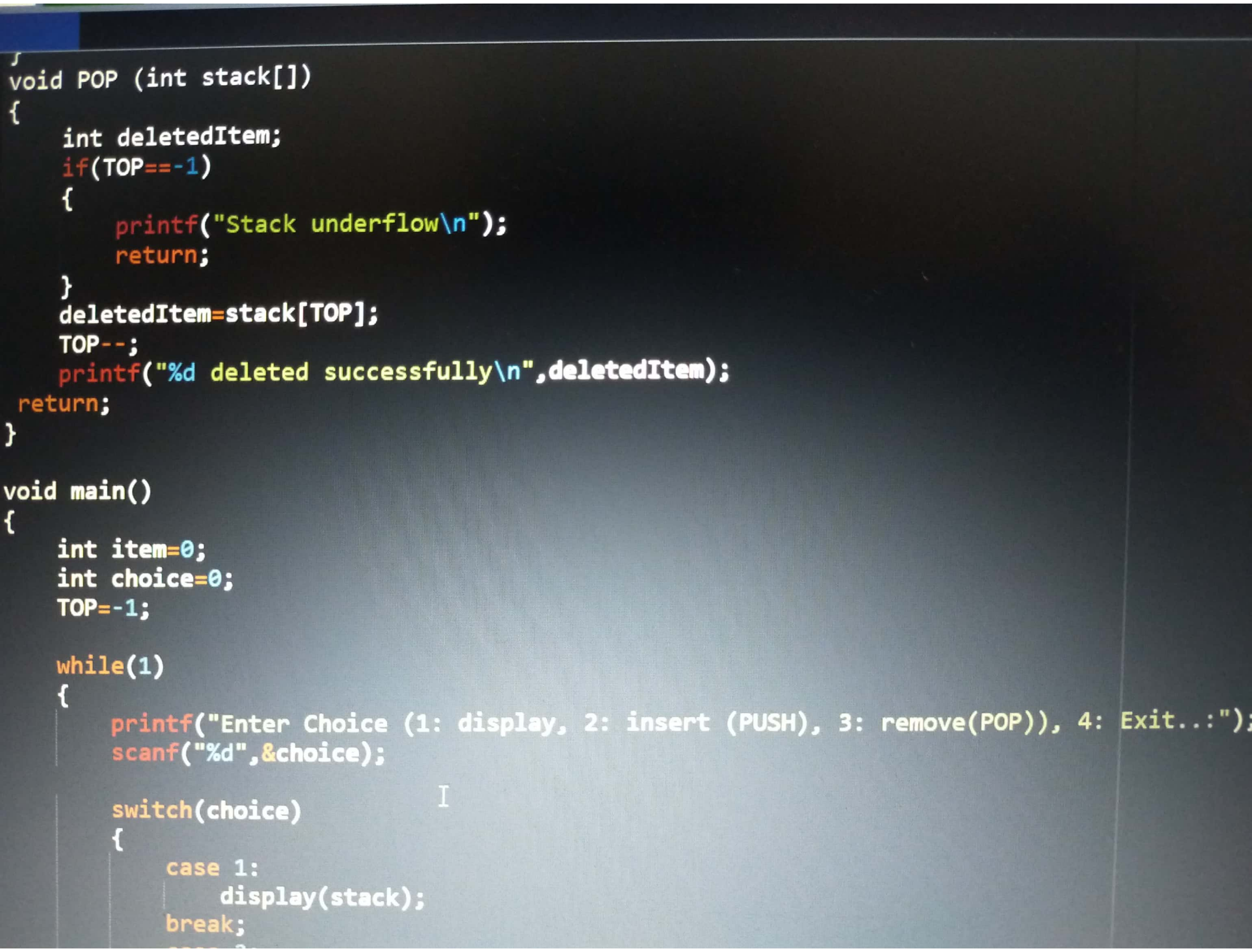


```c
#include <stdio.h>
#include<stdlib.h>
#define STACK_SIZE 3
int stack[STACK_SIZE],TOP;
void display(int stack[])
{
    int i=0;
    if(TOP==-1)
    {
        printf("Stack underflow.\n");
        return;
    }
    printf("%d",stack[TOP]);
    for(i=TOP-1;i >=0;i--)
    {
        printf("\n%d",stack[i]);
    }
    printf("\n\n");
}
void PUSH(int stack[],int item)
{
    if(TOP==STACK_SIZE-1)
    {
        printf("\nStack overflow\n");
        return;
    }
    TOP++;
    stack[TOP]=item;
}
void POP (int stack[])
{
```

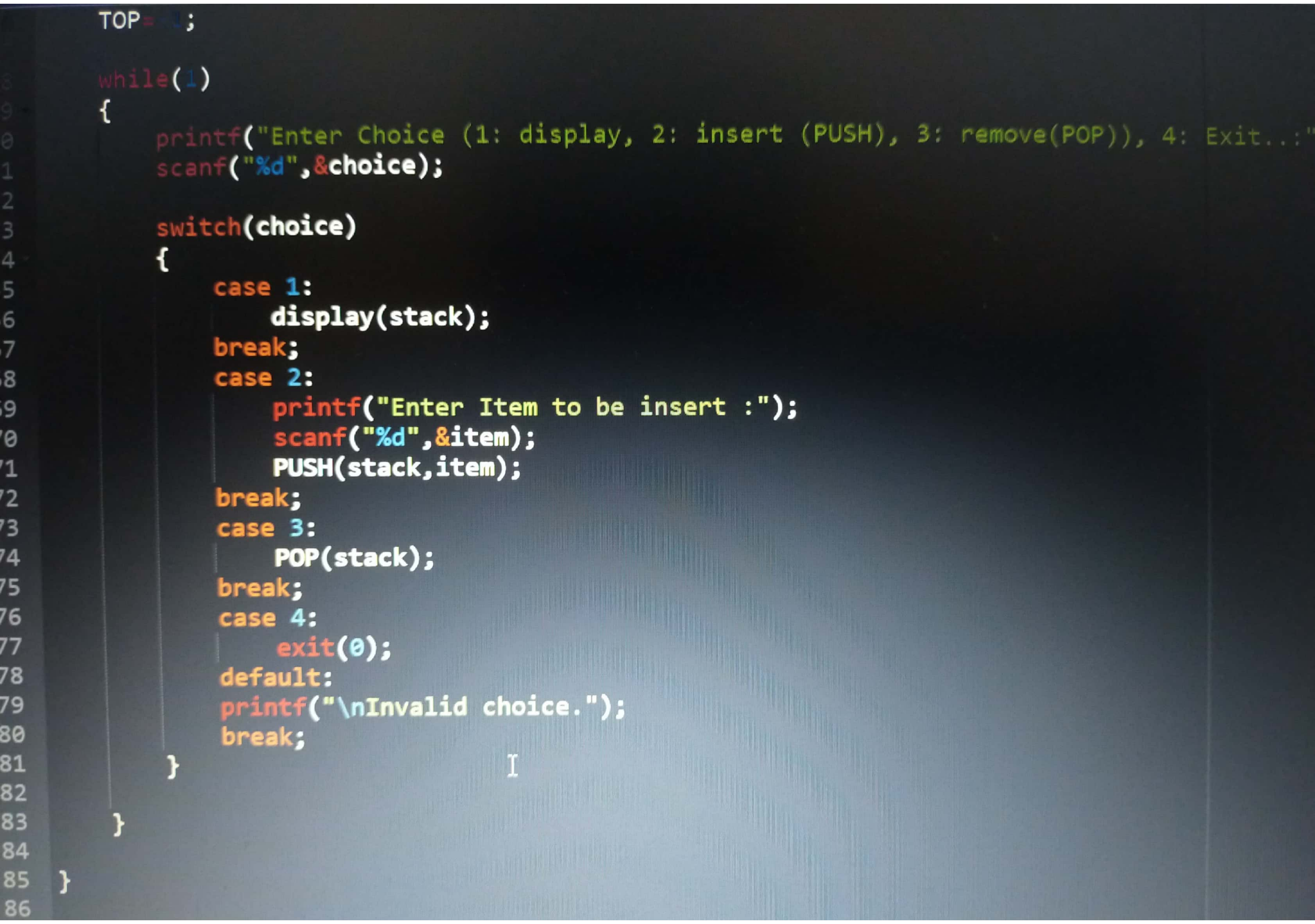


```
}
void POP (int stack[])
{
    int deletedItem;
    if(TOP==-1)
    {
        printf("Stack underflow\n");
        return;
    }
    deletedItem=stack[TOP];
    TOP--;
    printf("%d deleted successfully\n",deletedItem);
 return;
}

void main()
{
    int item=0;
    int choice=0;
    TOP=-1;

    while(1)
    {
        printf("Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:");
        scanf("%d",&choice);

        switch(choice)
        {
            case 1:
                display(stack);
            break;
            case 2:
```



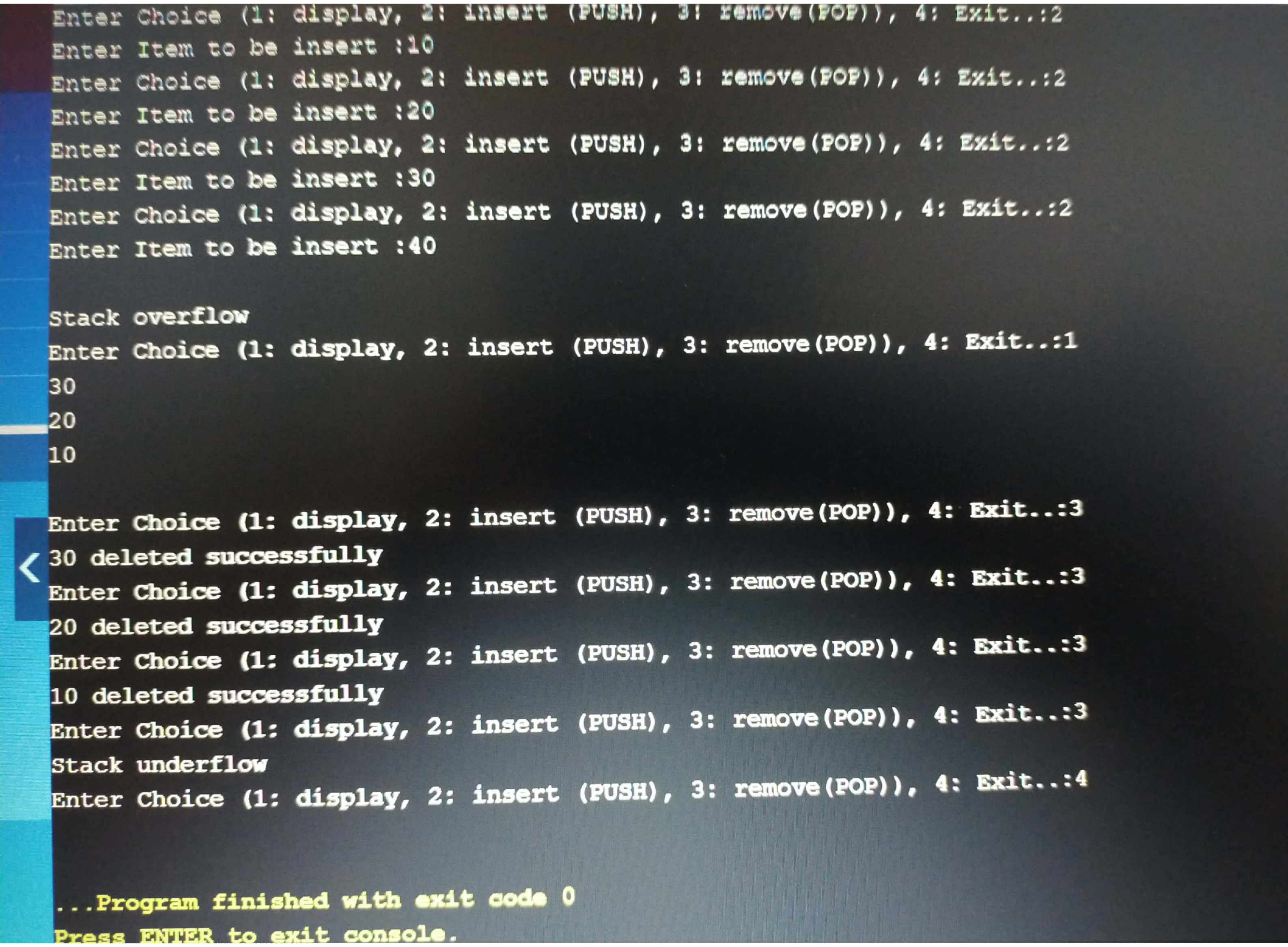
```c
    TOP=-1;

    while(1)
    {
        printf("Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:"
        scanf("%d",&choice);

        switch(choice)
        {
            case 1:
                display(stack);
            break;
            case 2:
                printf("Enter Item to be insert :");
                scanf("%d",&item);
                PUSH(stack,item);
            break;
            case 3:
                POP(stack);
            break;
            case 4:
                exit(0);
            default:
            printf("\nInvalid choice.");
            break;
         }

     }

}
```



```
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:2
Enter Item to be insert :10
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:2
Enter Item to be insert :20
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:2
Enter Item to be insert :30
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:2
Enter Item to be insert :40

Stack overflow
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:1
30
20
10

Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:3
30 deleted successfully
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:3
20 deleted successfully
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:3
10 deleted successfully
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:3
Stack underflow
Enter Choice (1: display, 2: insert (PUSH), 3: remove(POP)), 4: Exit..:4


...Program finished with exit code 0
Press ENTER to exit console.
```

18. Write a program to simulate the working of stack using an array with the following a: push, pop, display. The program should print appropriate msgs for stack overflow, stack underflow.

```
# include <stdio.h>
# define ~~MAX~~ STACK_SIZE ~~10~~ 2
int stack [~~MAX~~ STACK_SIZE], TOP;
void display (int* stack [])
{
 int i=0;
 if (Top==-1) {
   Printf (" stack underflow\n");
    return;
 }
 Printf (" %d", stack[top]);
 for (i=Top -1; i>=0; i--){
    Printf(" %d", stack[i]);
 }
  Printf ("\n");
}.
```

```
void push ( int stack[ ], int item)
 {
  if (Top = = STACK_SIZE -1)
 {
 Printf ("stack Overflow\n" );
   return;
 }
  Top ++;
 stack [Top] = item;
}
 void pop( int stack[ ]) {
 int deleted_item;
 if (Top = = -1) {
 Printf (" stack underflow \n");
   return;
}
 deleted_item = stack [Top];
   Top --;
  Printf ("%d deleted successfully "%d, deleted_item
  return;
}
 void main(){
 int item = 0 ;
 int choice = 0;
int Top = -1
```

```
~~while~~ for(;;){
 Printf("Enter choice (1: display, 2: insert (Push)
  3: Remove (Pop), 4: Exit.:");
scanf("%d", &choice);
switch (choice) {
  case 1:  display (stack);
           break;
  case 2:  printf("Enter item to be inserted:")
           scanf("%d", &item);
           Push(stack, item);
           break;
  case 3:  pop(stack);
           break;
  case 4:  exit(0);
  default: printf("\n Invalid input");
           break;
    }
 }.
```

output:

```
Enter choice (1: display, 2: insert, 3: pop, 4: exit): 2

Enter item to be inserted: 10

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 2

Enter item to be inserted: 20.

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 2

 Stack overflow

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 1

 20
 10

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 3

 20 deleted successfully.

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 3

 10 deleted succesfully.

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 3

 Stack underflow

Enter choice (1: display, 2: insert, 3: pop, 4: exit): 4

 ... Program finished with exit code 0.
```