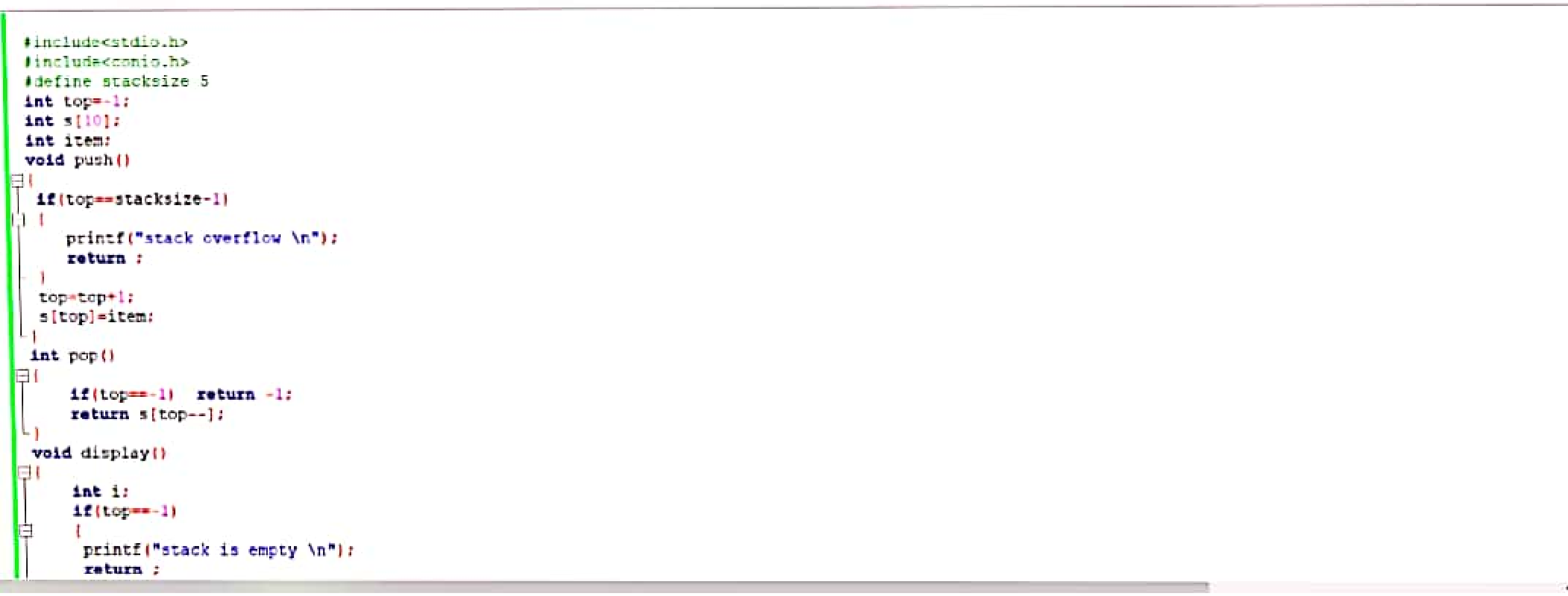
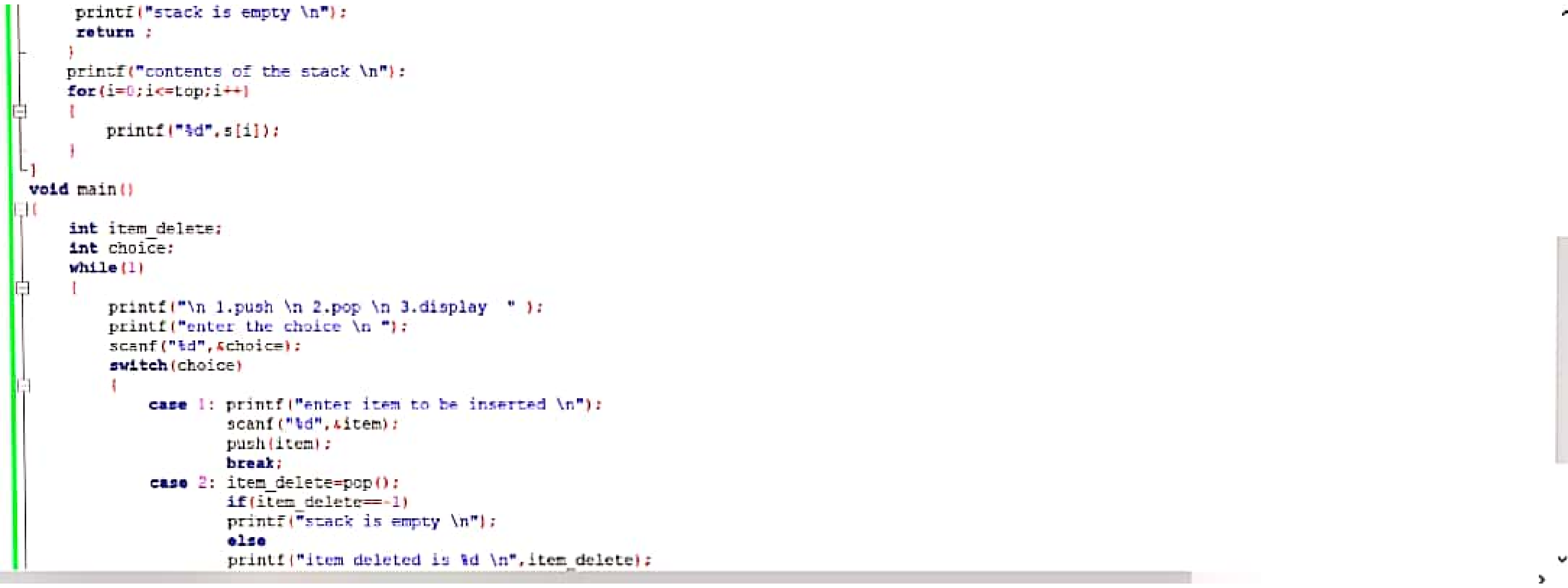
```
# include <stdio.h>
# include <conio.h>
# define stacksize 5
int top = -1;
 ints[10] ;
 int item;
 void push ()
 {
 If (top = = stacksize -1)
 {
  printf ("stack overflow\n");
  return ;
 }
  top = top + 1;
  s [top] = item ;
 }
 int pop ()
  {
   If (top = = -1) return -1;
    return s[top --];
  }
    Void display ()
    {
    int i;
    If (top = = -1)
    {
```

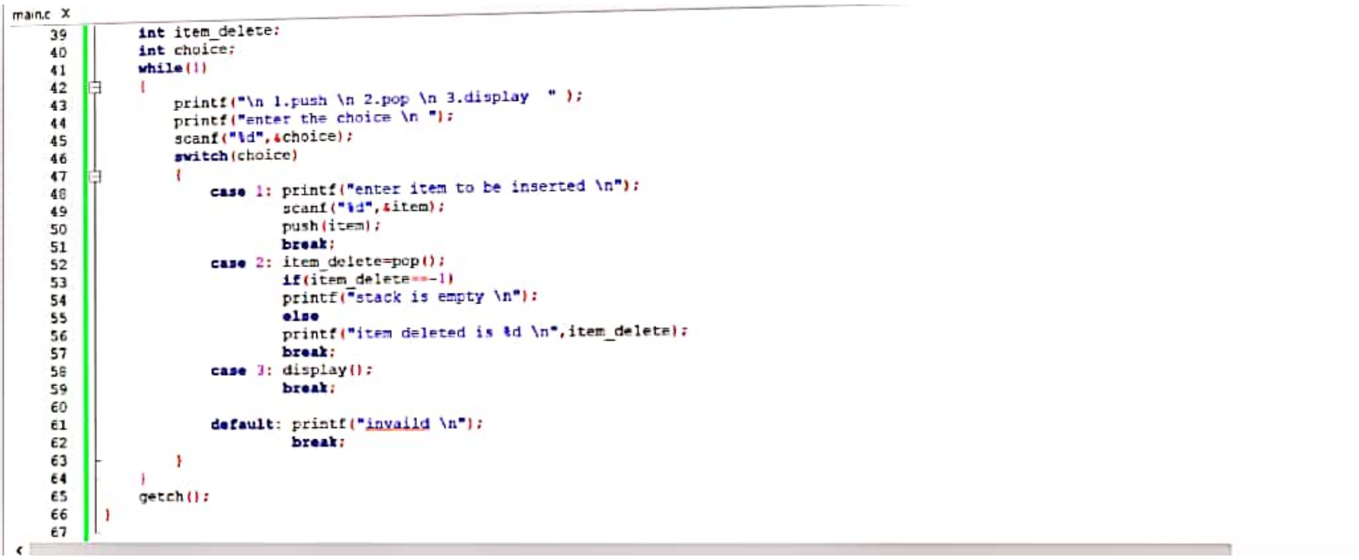
```
printf ("stack is empty \n");
return ;
 }
 printf ("Contents of the stack \n");
 for (i=0; i<=top; i++)
 {
   printf ("%d", s[i]);
  }
 }
  void main ()
  {
    int item_delete;
    int choice;
    while (1)
    {
    printf ("\n 1.push \n 2. pop \n 3. display ");
       printf ("enter the choice \n");
       Scanf ("%d", & choice);
       switch (choice)
       {
        case 1: printf ("enter item to be Inserted\n");
           Scanf ("%d", & item);
           break;
Case 2: item_ delete = pop ();
        If ( item_delete == -1)
        printf ("stack is empty \n");
        else
        printf ("item deleted is %d \n", item_delete
          break;
        Case 3: display ();
           break;
        default : printf ("invalid \n");
              break;
        }
      }
      getch ();
   }
```



```
#include<stdio.h>
#include<conio.h>
#define stacksize 5
int top=-1;
int s[10];
int item;
void push()
{
 if(top==stacksize-1)
 {
    printf("stack overflow \n");
    return ;
 }
 top=top+1;
 s[top]=item;
}
int pop()
{
     if(top==-1)  return -1;
     return s[top--];
}
void display()
{
     int i;
     if(top==-1)
     {
      printf("stack is empty \n");
      return ;
```

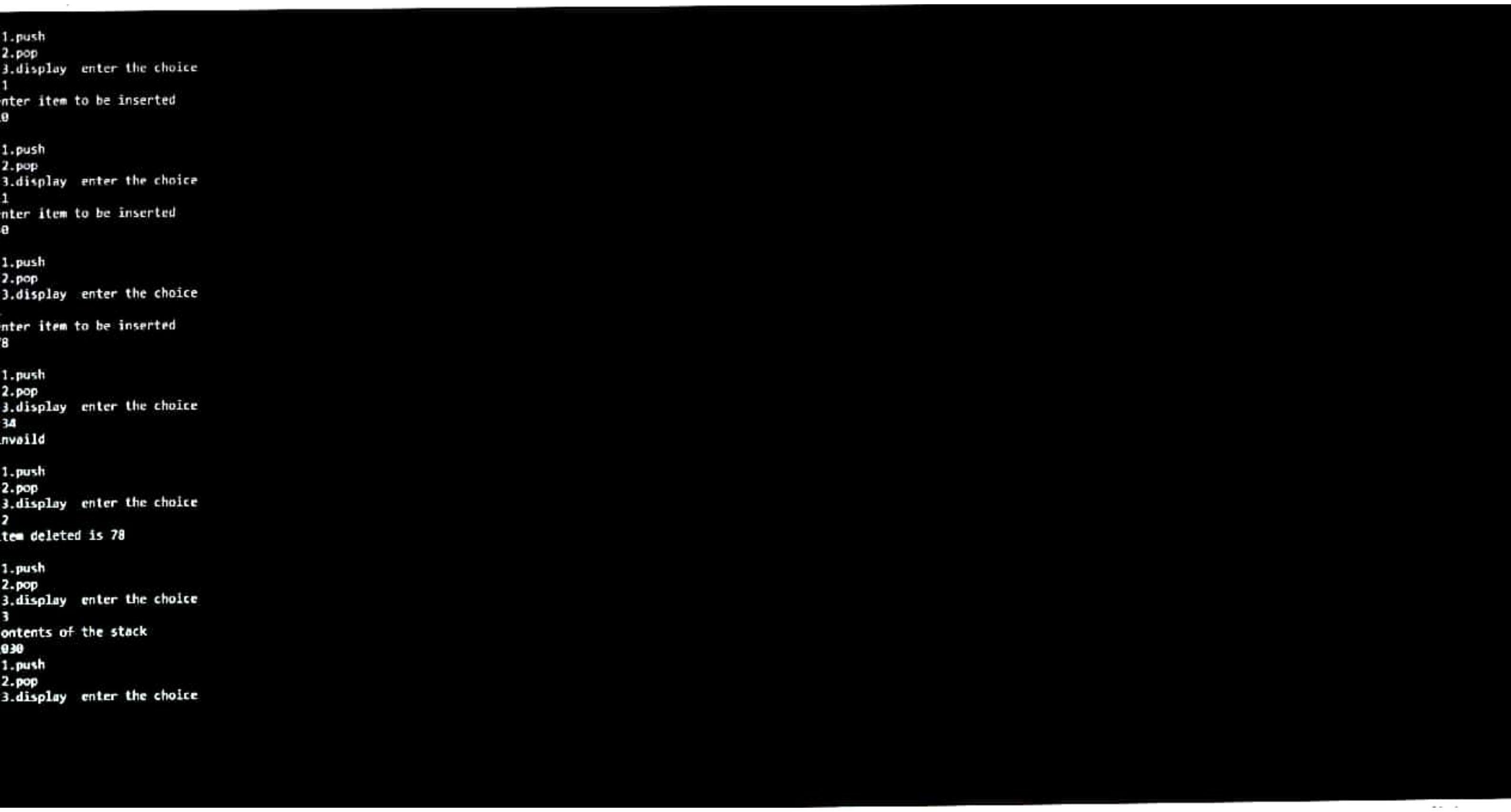


```
     printf("stack is empty \n");
     return ;
    }
   printf("contents of the stack \n");
   for(i=0;i<=top;i++)
   {
       printf("%d",s[i]);
   }
}
void main()
{
   int item_delete;
   int choice;
   while(1)
   {
       printf("\n 1.push \n 2.pop \n 3.display  " );
       printf("enter the choice \n ");
       scanf("%d",&choice);
       switch(choice)
       {
           case 1: printf("enter item to be inserted \n");
                   scanf("%d",&item);
                   push(item);
                   break;
           case 2: item_delete=pop();
                   if(item_delete==-1)
                   printf("stack is empty \n");
                   else
                   printf("item deleted is %d \n",item_delete);
```



```c
    int item_delete;
    int choice;
    while(1)
    {
        printf("\n 1.push \n 2.pop \n 3.display  " );
        printf("enter the choice \n ");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1: printf("enter item to be inserted \n");
                    scanf("%d",&item);
                    push(item);
                    break;
            case 2: item_delete=pop();
                    if(item_delete==-1)
                    printf("stack is empty \n");
                    else
                    printf("item deleted is %d \n",item_delete);
                    break;
            case 3: display();
                    break;

            default: printf("invaild \n");
                     break;
        }
    }
    getch();
}
```



```
1.push
2.pop
3.display  enter the choice
1
nter item to be inserted
0

1.push
2.pop
3.display  enter the choice
1
nter item to be inserted
0

1.push
2.pop
3.display  enter the choice

nter item to be inserted
8

1.push
2.pop
3.display  enter the choice
34
nvaild

1.push
2.pop
3.display  enter the choice
2
tem deleted is 78

1.push
2.pop
3.display  enter the choice
3
ontents of the stack
030
1.push
2.pop
3.display  enter the choice
```