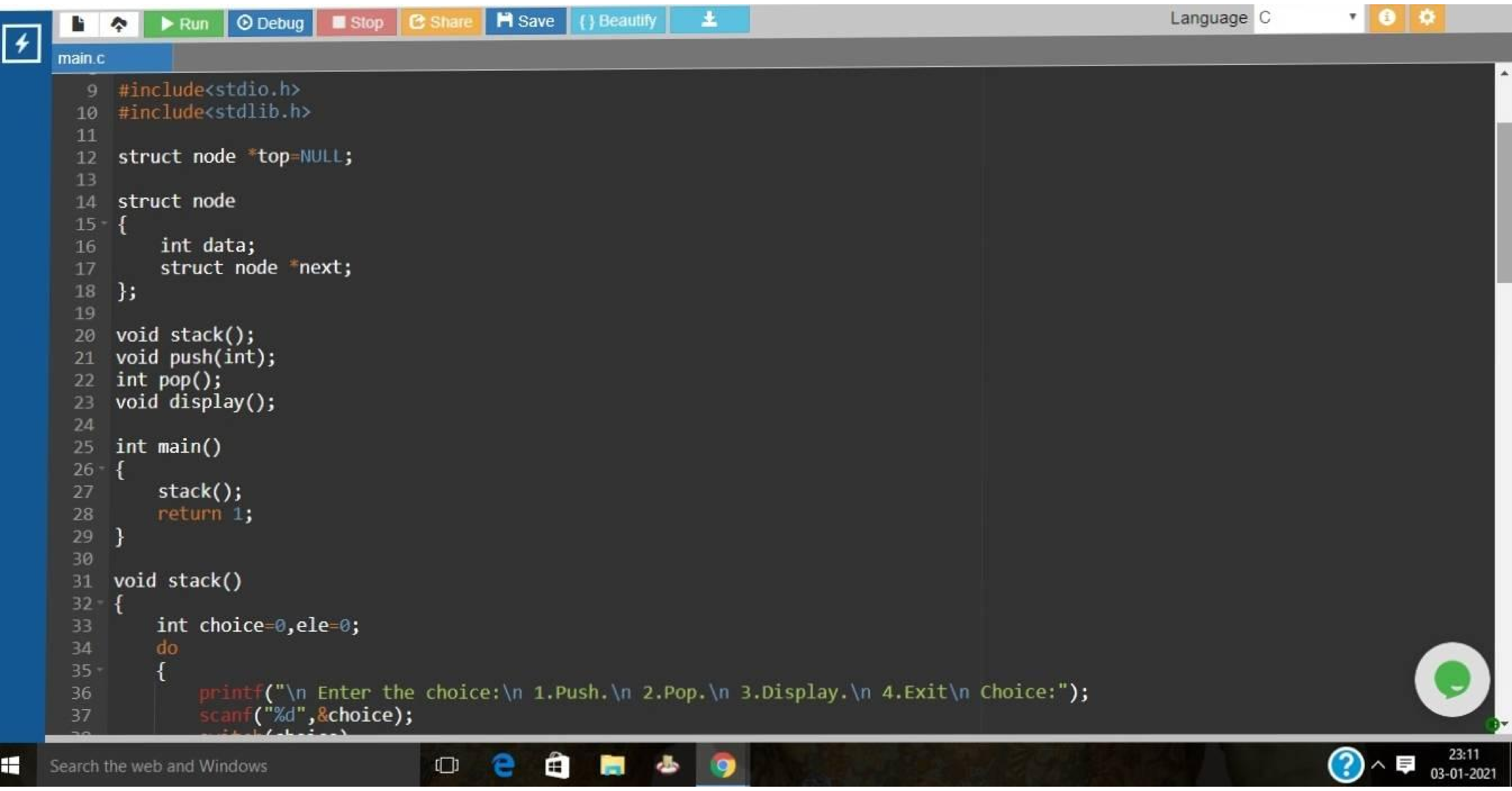
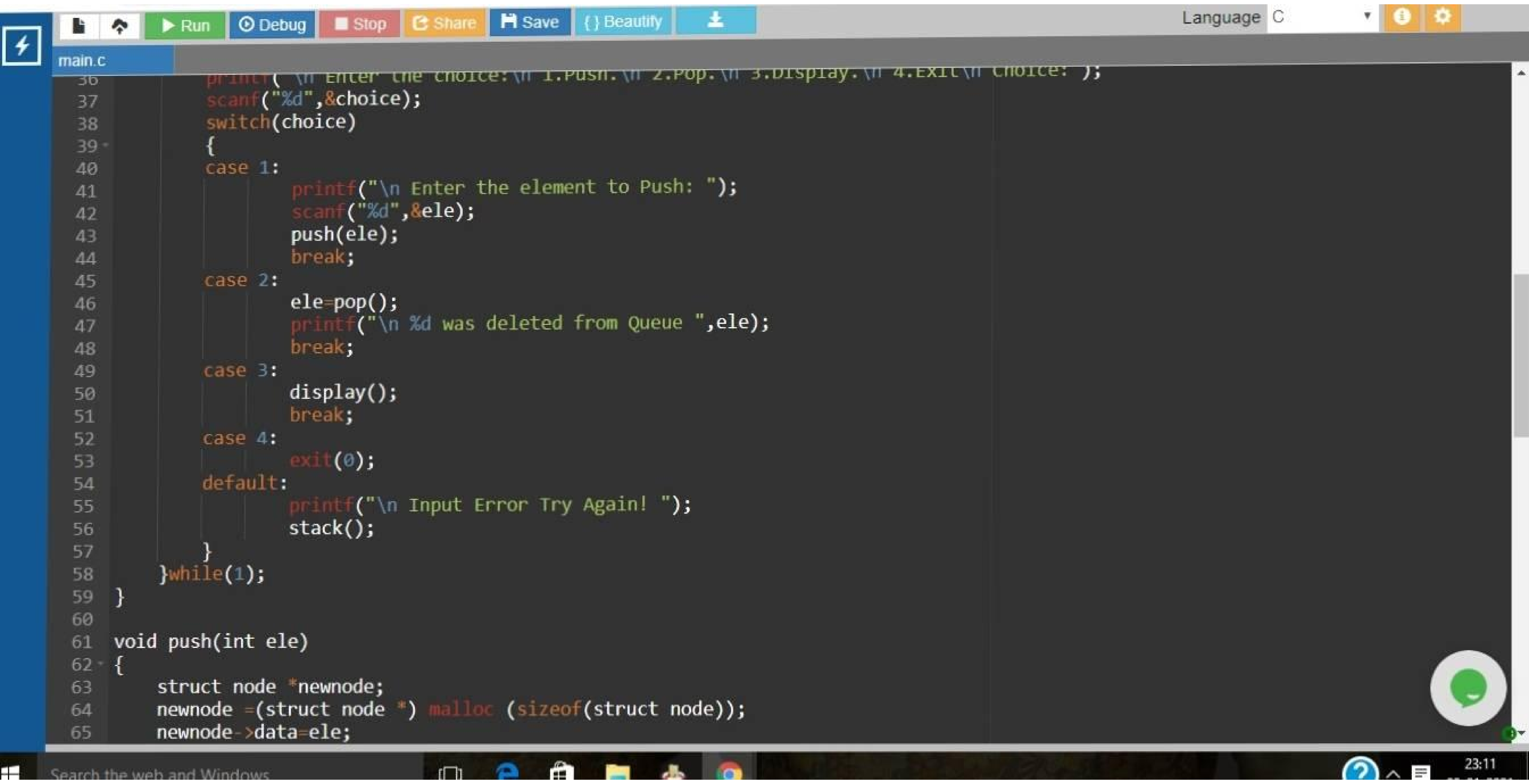
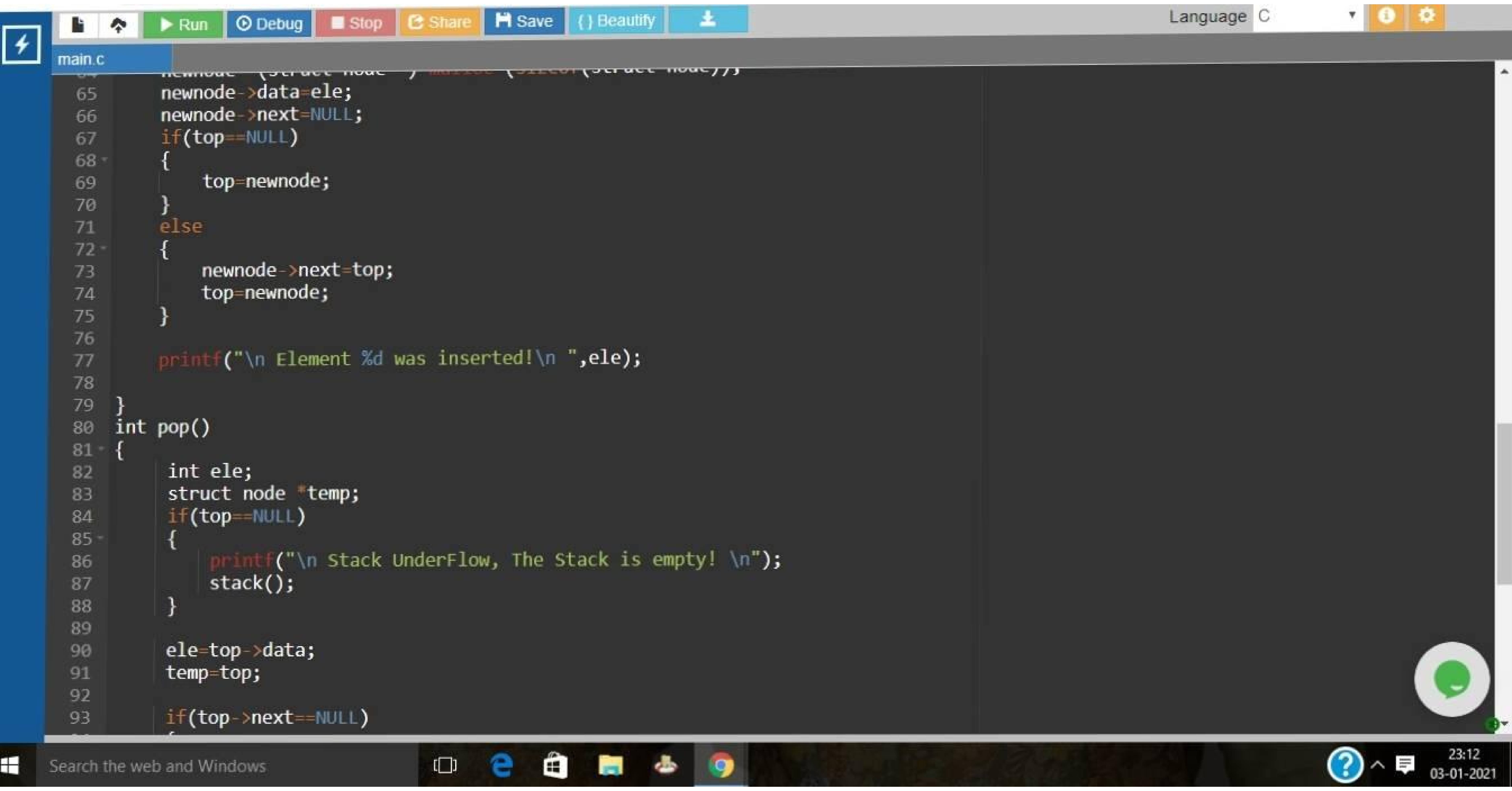


```c
#include<stdio.h>
#include<stdlib.h>

struct node *top=NULL;

struct node
{
    int data;
    struct node *next;
};

void stack();
void push(int);
int pop();
void display();

int main()
{
    stack();
    return 1;
}

void stack()
{
    int choice=0,ele=0;
    do
    {
        printf("\n Enter the choice:\n 1.Push.\n 2.Pop.\n 3.Display.\n 4.Exit\n Choice:");
        scanf("%d",&choice);
```

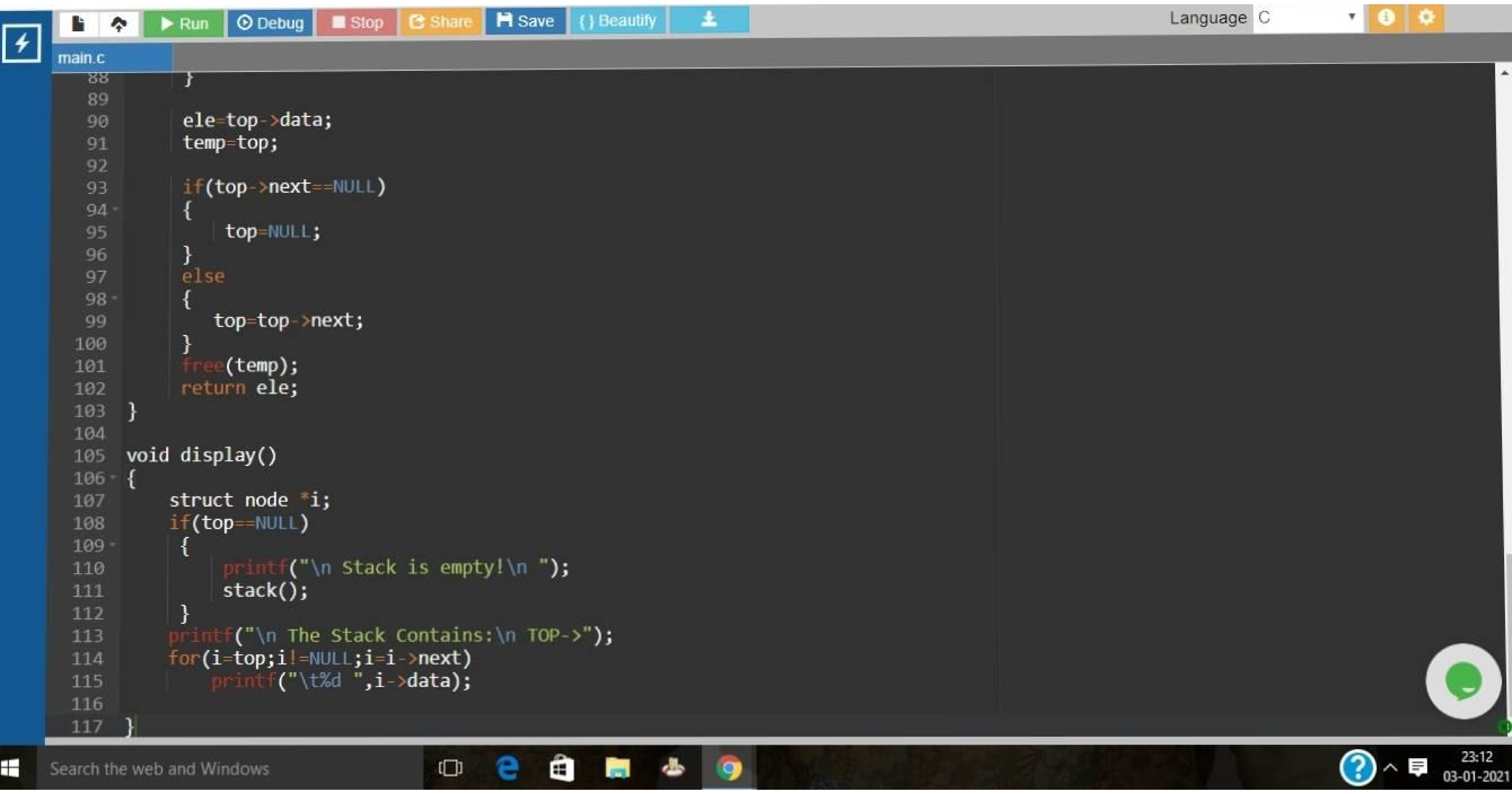


```
        printf("\n Enter the choice:\n 1.Push.\n 2.Pop.\n 3.Display.\n 4.Exit\n Choice: ");
        scanf("%d",&choice);
        switch(choice)
        {
        case 1:
                printf("\n Enter the element to Push: ");
                scanf("%d",&ele);
                push(ele);
                break;
        case 2:
                ele=pop();
                printf("\n %d was deleted from Queue ",ele);
                break;
        case 3:
                display();
                break;
        case 4:
                exit(0);
        default:
                printf("\n Input Error Try Again! ");
                stack();
        }
    }while(1);
}

void push(int ele)
{
    struct node *newnode;
    newnode =(struct node *) malloc (sizeof(struct node));
    newnode->data=ele;
```

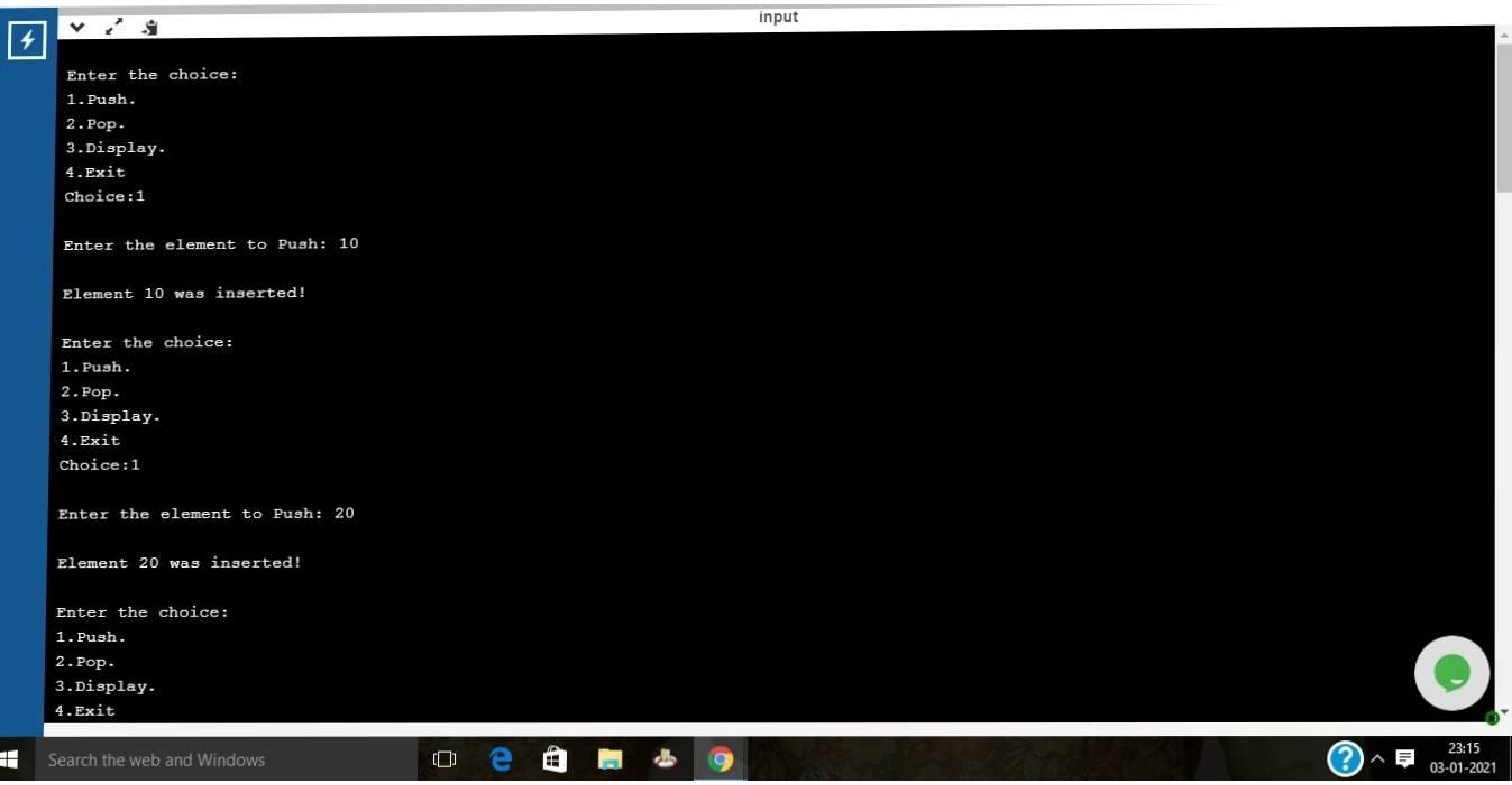
Scanned by TapScanner



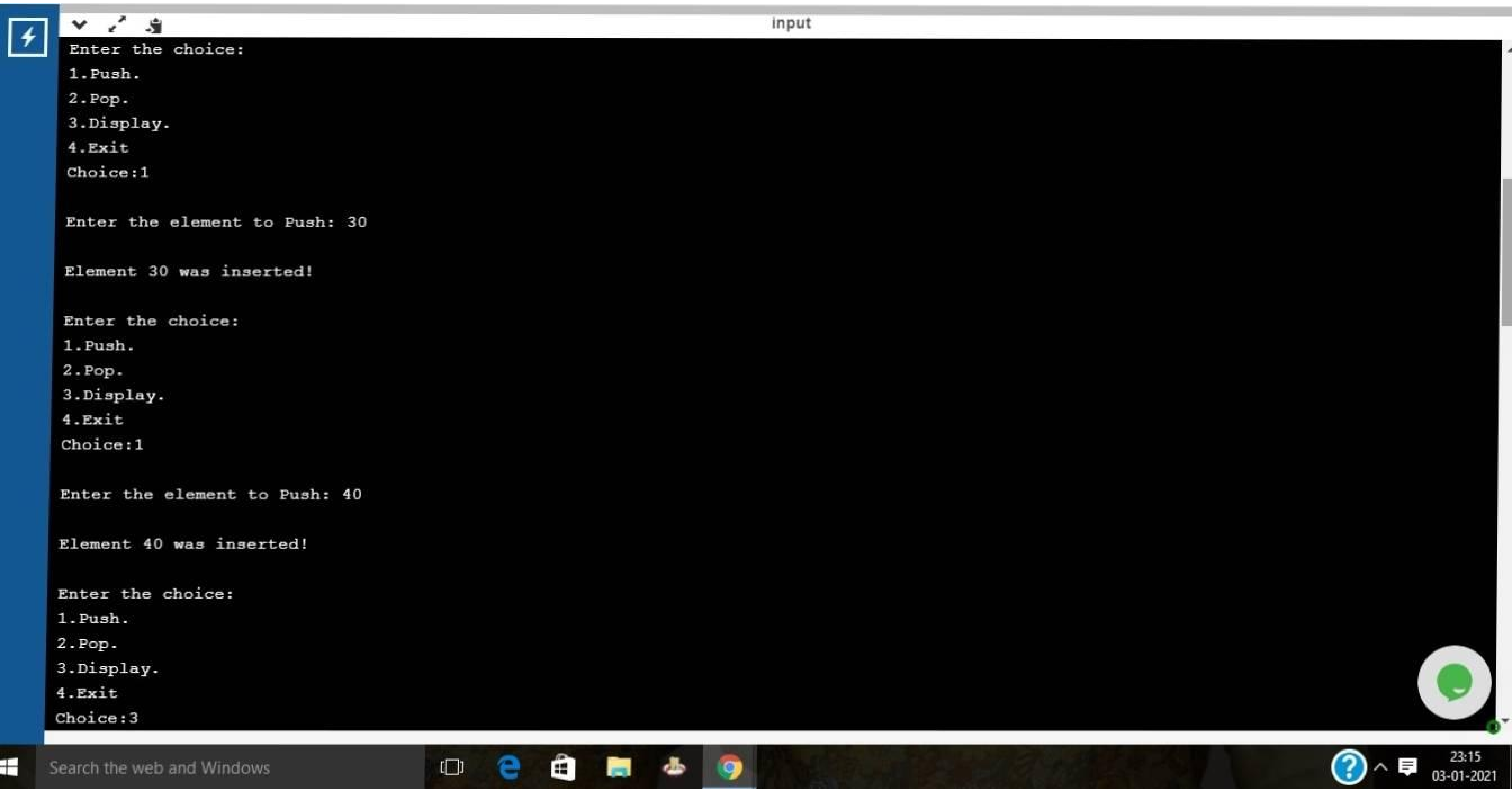
```c
    newnode=(struct node*)malloc(sizeof(struct node));
    newnode->data=ele;
    newnode->next=NULL;
    if(top==NULL)
    {
        top=newnode;
    }
    else
    {
        newnode->next=top;
        top=newnode;
    }

    printf("\n Element %d was inserted!\n ",ele);

}
int pop()
{
     int ele;
     struct node *temp;
     if(top==NULL)
     {
         printf("\n Stack UnderFlow, The Stack is empty! \n");
         stack();
     }

     ele=top->data;
     temp=top;

     if(top->next==NULL)
```



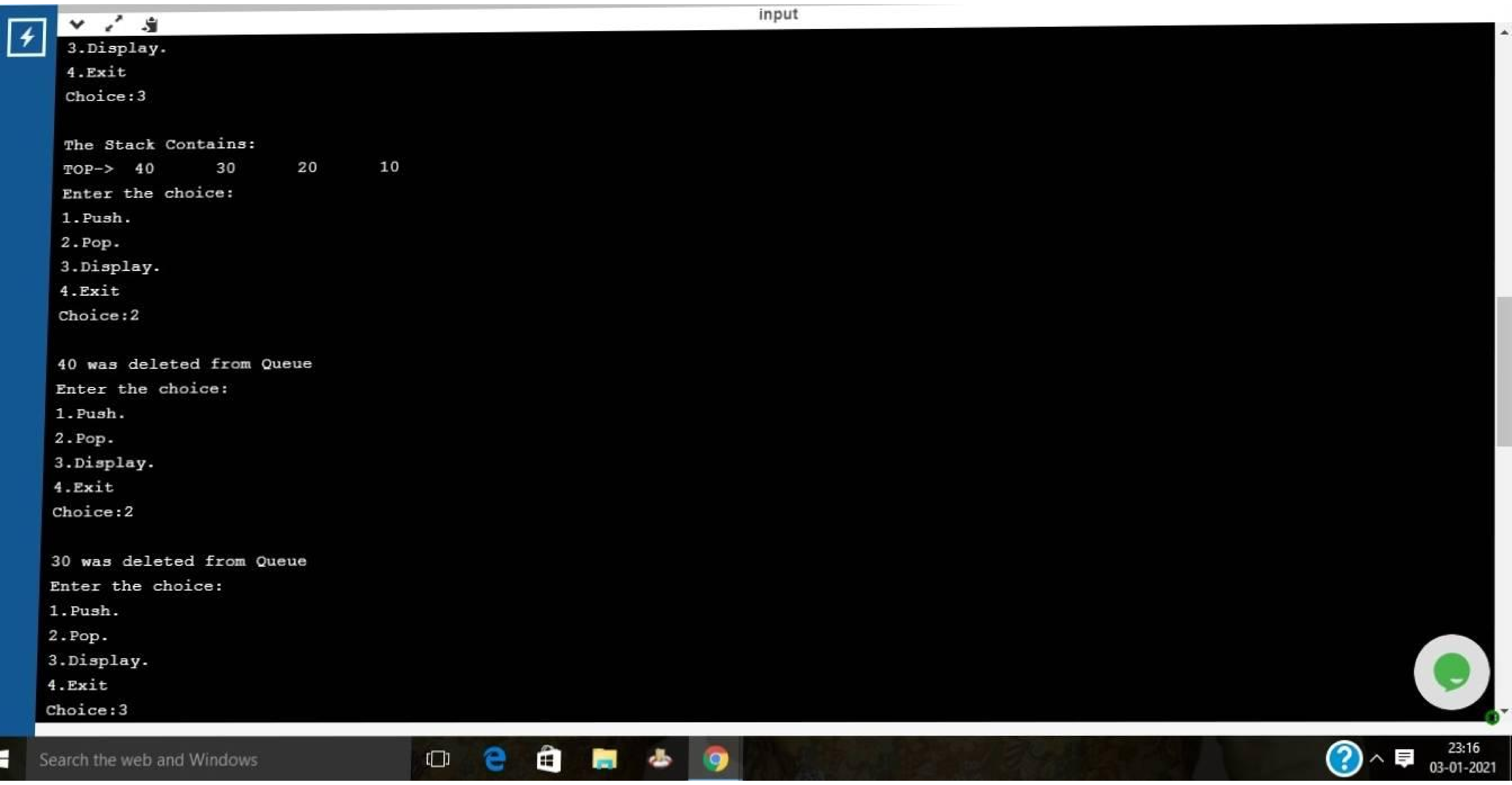
```
    }

    ele=top->data;
    temp=top;

    if(top->next==NULL)
    {
        top=NULL;
    }
    else
    {
       top=top->next;
    }
    free(temp);
    return ele;
}

void display()
{
    struct node *i;
    if(top==NULL)
    {
        printf("\n Stack is empty!\n ");
        stack();
    }
    printf("\n The Stack Contains:\n TOP->");
    for(i=top;i!=NULL;i=i->next)
        printf("\t%d ",i->data);

}
```



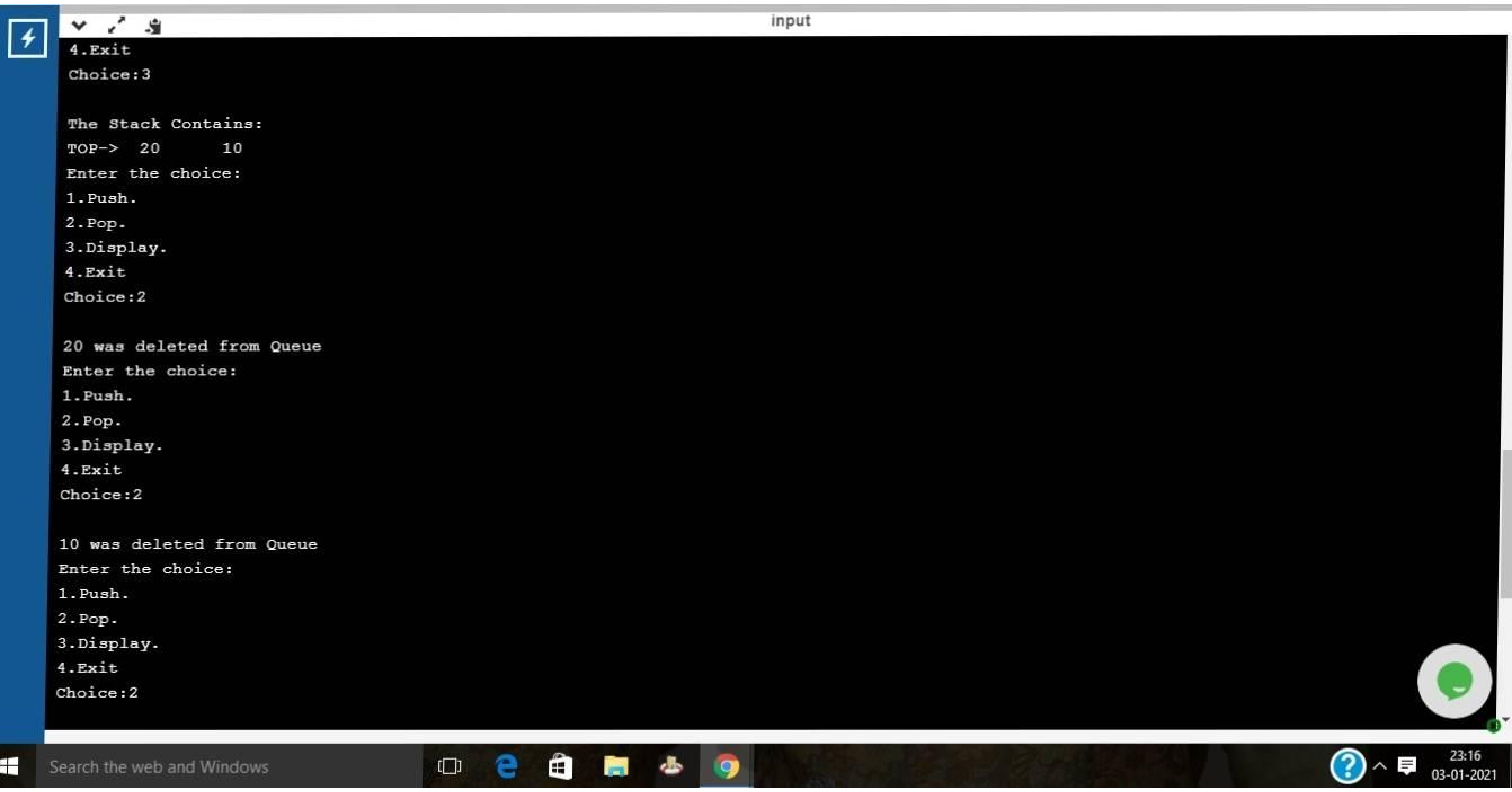
```
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:1

Enter the element to Push: 10

Element 10 was inserted!

Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:1

Enter the element to Push: 20

Element 20 was inserted!

Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
```



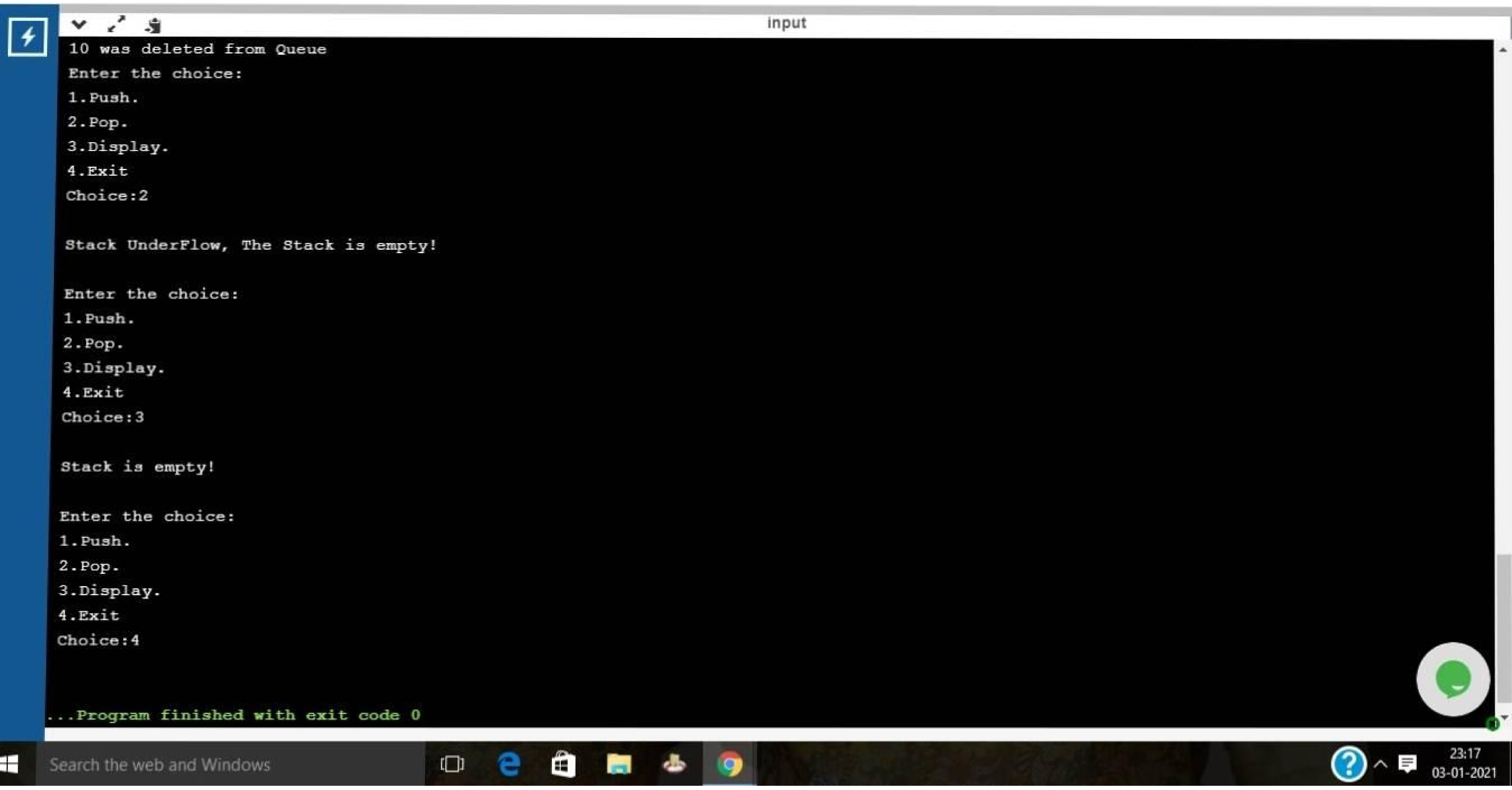
```
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:1

Enter the element to Push: 30

Element 30 was inserted!

Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:1

Enter the element to Push: 40

Element 40 was inserted!

Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:3
```



```
3.Display.
4.Exit
Choice:3

The Stack Contains:
TOP->  40       30       20       10
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:2

40 was deleted from Queue
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:2

30 was deleted from Queue
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:3
```



```
4.Exit
Choice:3

The Stack Contains:
TOP->  20      10
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:2

20 was deleted from Queue
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:2

10 was deleted from Queue
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:2
```



```
10 was deleted from Queue
Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:2

Stack UnderFlow, The Stack is empty!

Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:3

Stack is empty!

Enter the choice:
1.Push.
2.Pop.
3.Display.
4.Exit
Choice:4


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