

≡ File Edit Search Run Compile Debug Project Options Window Help

[■] \TURBOC3\SARANYA\STACKLIN.C

1=[■]

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct stack
{
    int data;
    struct stack *next;
}
*top=NULL,*temp;
int main()
{
    int c;
    clrscr();
    while(1)
    {
        printf("\nMain Menu\n");
        printf("\n1.Push\n");
        printf("\n2.Pop\n");
        printf("\n3.Display\n");
        printf("\n4.exit\n");
        printf("\n enter the choice");
```

1:1

F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

≡ File Edit Search Run Compile Debug Project Options Window Help

[■] \TURBOC3\SARANYA\STACKLIN.C 1=[■]

```
printf("\n enter the choice");
scanf("%d",&c);
switch(c)
{
case 1: push();
        break;
case 2: pop();
        break;
case 3: display();
        break;
case 4: exit(1);
        break;
default :printf("wrong");
}
}
}
push()
{
struct stack *temp;
int item;
temp=(struct stack*)malloc(sizeof(struct stack));
```

41:1

F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

≡ File Edit Search Run Compile Debug Project Options Window Help

[■] \TURBOC3\SARANYA\STACKLIN.C

1=[■]

```
temp=(struct stack*)malloc(sizeof(struct stack));
printf("\n insert element on to the stack");
scanf("%d",&item);
temp->data=item;
temp->next=top;
top=item;
return;
}
pop()
{
struct stack *ptr;
if(top==NULL)
printf("stack is empty");
else
{
temp=top;
printf("\n popped item is %d \n:",temp->data);
top=top->next;
free(temp);
}
return;
```

61:1

F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

≡ File Edit Search Run Compile Debug Project Options Window Help

[■] \TURBOC3\SARANYA\STACKLIN.C

1=[■]

```
display()
{
int i;
struct stack *ptr;
ptr=top;
if(top==NULL)
{
printf("stack is empty");
}
else
{
printf("stack elements are:\n");
while(ptr!=NULL)
{
printf("%d\n ",ptr->data);
ptr=ptr->next;
}
}
return;
}
```

83:1

F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

Main Menu

1.Push

2.Pop

3.Display

4.exit

enter the choice 2
stack is empty
Main Menu

1.Push

2.Pop

3.Display

4.exit

enter the choice_

Program no: 7

Program to implement ^{stack using} singly linked list

Program

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

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

*top = NULL, *temp;

int main()
{
    int c;
    clrscr();
    while(1)
    {
        printf("\n main menu\n");
        printf("\n 1. push");
        printf("\n 2. pop");
    }
}
```

```

printf("ln3. Display\n");
printf("ln4. exit\n");
printf("Enter the choice!");
scanf("%d", &c);
switch(c)
{
    case 1: push();
            break;
    case 2: pop();
            break;
    case 3: display();
            break;
    case 4: exit();
            break;
    default: printf("Wrong");
}
}
}
}
push()
{
    struct stack *temp;
    int temp item;

```

```

temp = (struct stack*) malloc (sizeof (struct stack));
printf("In insert element on to the stack");
scanf("%d", &iitem);
temp->data = iitem;
temp->next = top;
top = iitem;
return;
}
pop()
{
struct stack *ptr;
if (top == NULL)
printf("stack is empty");
else
{
temp = top;
printf("In popped item is %d\n", temp->data);
top = top->next;
free (temp);
}
return;
}
display()
{

```



```

int i;
struct stack *ptr;
ptr = top;
if (top == NULL)
{
    printf("stack is empty");
}
else
{
    printf("stack elements are : \n");
    while (ptr != NULL)
    {
        printf("%d\n", ptr->data);
        ptr = ptr->next;
    }
}
return;
}

```

output

Main menu

1. push
2. pop
3. Display
4. exit

enter the choice : 2
stack is empty.

Main menu

1. push
2. pop
3. Display
4. exit

enter the choice : 1

insert the element on to the stack: 10.