

SOURCE CODE: LINKED STACK:

```
#include<stdio.h>

#include<conio.h>

#include<malloc.h>

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

struct stack *top=NULL;

struct stack *push(struct stack *,int);

struct stack *display(struct stack *);

struct stack *pop(struct stack *);

int peek(struct stack *);

void main()
{
    int val,ch;

    clrscr();

    do
    {
        printf("\n*****Main Menu*****\n");
        printf("1.Push\n2.Pop\n3.Peek\n4.Display\n");
        printf("Enter your choice\n");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1:
                printf("Enter the element to the stack\n");
                scanf("%d",&val);
                top=push(top,val);
                break;
```

```

        case 2:
            top=pop(top);
            break;
        case 3:
            val=peek(top);
            if(val!=1)
                printf("The value at top of stack is %d\n",val);
            else
                printf("Stack is empty\n");
            break;
        case 4:
            top=display(top);
            break;
    }
}while(ch>=1&&ch<=4);
getch();
}

struct stack *push(struct stack *top, int val)
{
    struct stack *ptr;
    ptr=(struct stack *)malloc(sizeof(struct stack *));
    ptr->data=val;
    ptr->next=NULL;
    if(top==NULL)
    {
        top=ptr;
    }
    else
    {
        ptr->next=top;
        top=ptr;
    }
}

```

```

        return top;
    }
    struct stack *pop(struct stack *top)
    {
        struct stack *ptr;
        ptr=top;
        if(top==NULL)
            printf("Stack is Overflow\n");
        else
        {
            top=top->next;
            printf("The value being deleted is %d\n",ptr->data);
            free(ptr);
        }
        return top;
    }
    int peek(struct stack *top)
    {
        if(top==NULL)
            return -1;
        else
            return top->data;
    }
    struct stack *display(struct stack *top)
    {
        struct stack *ptr;
        ptr=top;
        if(top==NULL)
            printf("Stack is empty\n");
        else
        {

```

```

        while(ptr!=NULL)
        {
            printf("%d\t",ptr->data);
            ptr=ptr->next;
        }
    }
    return top;
}

```

OUTPUT:

```

*****Main Menu*****
1.Push
2.Pop
3.Peek
4.Display
Enter your choice
1
Enter the element to the stack
10

*****Main Menu*****
1.Push
2.Pop
3.Peek
4.Display
Enter your choice
1
Enter the element to the stack
20

*****Main Menu*****
1.Push
2.Pop
3.Peek
4.Display
Enter your choice
1
Enter the element to the stack
30

*****Main Menu*****
1.Push
2.Pop
3.Peek
4.Display
Enter your choice
2
The value being deleted is 30

```

*****Main Menu*****

1.Push

2.Pop

3.Peek

4.Display

Enter your choice

3

The value at top of stack is 20

*****Main Menu*****

1.Push

2.Pop

3.Peek

4.Display

Enter your choice

4

20 10

*****Main Menu*****

1.Push

2.Pop

3.Peek

4.Display

Enter your choice

5