```
#include<stdio.h>
#include<stdlib.h>
struct Node
{
 int data;
 struct Node *next;
}
*top = NULL;
void push(int);
void pop();
void display();
void search();
void main()
{
 int choice, value;
 printf("\n:: Stack using Linked List ::\n");
 while(1)
{
 printf("1. Push\n2. Pop\n3. Display\n4. Search\n5. Exit\n");
 printf("Enter your choice: ");
 scanf("%d",&choice);
 switch(choice)
{
 case 1: printf("Enter the value to be insert: ");
 scanf("%d", &value);
 push(value);
 break;
 case 2: pop(); break;
 case 3: display(); break;
```

```
 case 4: search(); break;
case 5: exit(0); break;
 default: printf("Wrong”);
 }
 }
}
void push(int value)
{
 struct Node *newNode;
 newNode = (struct Node*)malloc(sizeof(struct Node));
 newNode->data = value;
 if(top == NULL)
 newNode->next = NULL;
 else
 newNode->next = top;
 top = newNode;
 printf("\nInsertion is Success\n");
}
void pop()
{
 if(top == NULL)
 printf("\nStack is Empty\n");
 else{
 struct Node *temp = top;
 printf("\nDeleted element: %d", temp->data);
 top = temp->next;
 free(temp);
 }
}
```

```
void display()
{
 if(top == NULL)
printf("\nStack is Empty!!!\n");
 else{
 struct Node *temp = top;
 while(temp->next != NULL){
 printf("%d--->",temp->data);
 temp = temp -> next;
 }
 printf("%d--->NULL",temp->data);
 }
}
void search()
{
 struct Node *ptr;
 int item,i=0,flag;
 ptr = top;
 if(ptr == NULL)
 {
 printf("\nEmpty List\n");
 }
 else
 {
 printf("\nEnter item which you want to search:");
 scanf("%d",&item);
 while (ptr!=NULL)
 {
 if(ptr->data == item)
```

```
 {
 printf("item found at location %d ",i+1);
 flag=1;
 }
 Else
{
 flag=0;
 }
 i++;
 ptr = ptr -> next;
 }
 if(flag==0)
 {
 printf("Item not found\n");
 }
 }
}
```