```
#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");
printf("_____");
while(1)
{
printf("\n\n *** MENU ***\n\n");
printf("-----.");
printf("\n1. push \n2. pop\n3. display \n4. search
\n5. exit");
printf("\n enter your choice!");
scanf("%d", &choice");
switch(choice)
{
case 1: printf("enter the value to be insert:");
scanf("%d", &value);
```

```
Push(value);
break;
case2: POP(); break;
case3: display(); break;
case4 : search(); break;
case5 : exit(0); break;
default: printf("\n Invalid selection \n");
}
}
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("\n insertion is sucess \n");
}
void POP()
{
if (top == NULL)
printf("\n Stack is empty \n");
```

```
else
{
struct node *temp=top;
printf("\n deleted element : %d \n", temp->data);
top=temp->next;
free(temp);
}
}
void display()
{
if(top==NULL)
printf("\n stack is empty\n");
else
{
struct Node *temp = top;
while(temp->next!=NULL)
{
printf("%d", temp->data);
temp=temp->next;
}
printf("%d", temp->data);
}
}
void search()
{
struct Node *ptr;
int item, i=0, flag;
ptr=top;
```

```
if (ptr == NULL)
{
printf ("\nempty List\n");
}
else
{
Printf("\n enter item which to be searched:");
scanf ("%d", &item);
while (ptr != NULL)
else {
if(ptr -> data == item)
{
printf ("Item found at location %d \n", i+1);
flag = 1;
}
i++;
ptr = ptr -> next;
}
if (flag == 0)
{
printf ("item not found \n");
}
}
}
```