

Program (Stack using Linkedlist)

```
#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;

while(1){

printf("\nChoices: \n");

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("\nError..! Please try again\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("\nInsertion is Successfull\n");
}

void pop()
{
    if(top == NULL)
        printf("\nStack is Empty\n");
    else{
        struct Node *temp = top;
        printf("\nPopped 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 the item to be searched:");
scanf("%d",&item);

```

```
while (ptr!=NULL)
{
if(ptr->data == item)
{
flag=1;
printf("location of the item is %d ",i+1);
}
else
{
flag=0;
}
i++;
ptr = ptr -> next;
}
if(flag==0)
{
printf("Item not found\n");
}
}
```

Output

```
main.c
Choices:
1.Push
2.Pop
3.Display
4.Search
5.Exit
Enter your choice: 1
Enter the value to be insert: 7

Insertion is Successfull

Choices:
1.Push
2.Pop
3.Display
4.Search
5.Exit
Enter your choice: 1
Enter the value to be insert: 14

Insertion is Successfull

Choices:
1.Push
2.Pop
3.Display
4.Search
5.Exit
Enter your choice: 2

Popped element: 14
Choices:
1.Push
5.Exit
Enter your choice: 2
Popped element: 14
Choices:
1.Push
2.Pop
3.Display
4.Search
5.Exit
Enter your choice: 3
7-->NULL
Choices:
1.Push
2.Pop
3.Display
4.Search
5.Exit
Enter your choice: 4

Enter the item to be searched:2
Item not found

Choices:
1.Push
2.Pop
3.Display
4.Search
5.Exit
Enter your choice: 5

...Program finished with exit code 0
Press ENTER to exit console.
```