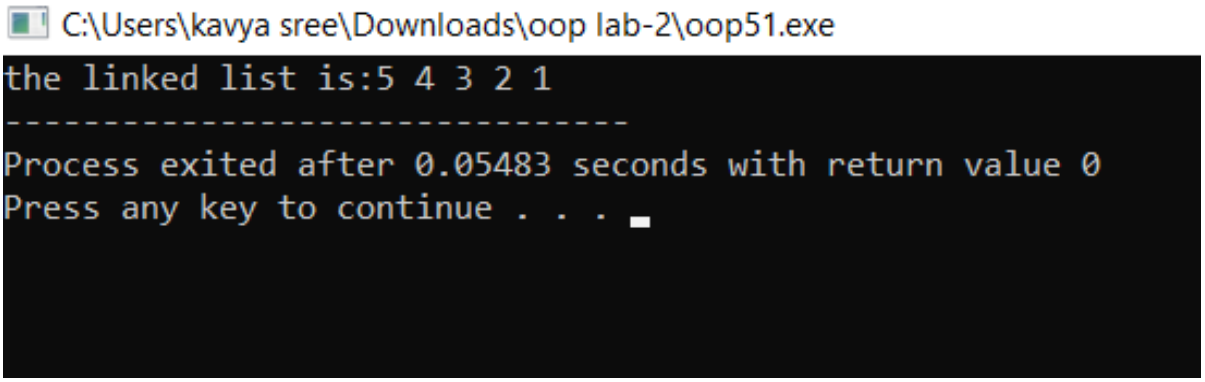
```
1. #include <iostream>
#include <stdlib.h>
using namespace std;
/* run this program using the console pauser or add your own getch, system("pause") or input loop */
struct node{
int data;
struct node *next;
};
struct node* head = NULL;
void createList(int new_data)
{
struct node* new_node = (struct node*)malloc(sizeof(struct node));
new_node->data = new_data;
new_node->next = head;
head = new_node;
}
void display()
{
struct node* ptr;
ptr = head;
while(ptr!=NULL)
```

```
{
cout<< ptr->data<<" ";
ptr = ptr->next;
}
}
int main(int argc, char** argv) {
createList(1);
createList(2);
createList(3);
createList(4);
createList(5);
cout<<"the linked list is:";
display();
return 0;
}
```



2. #include <iostream>

#include <cstdlib>

using namespace std;

struct node

{

int data;

struct node *next;

};

struct node *head = NULL, *temp;

void createList(struct node *h)

{

int value=1;

while(1)

{

cout<<"\nEnter the value : ";

cin>>value;

if(value==0)

break;

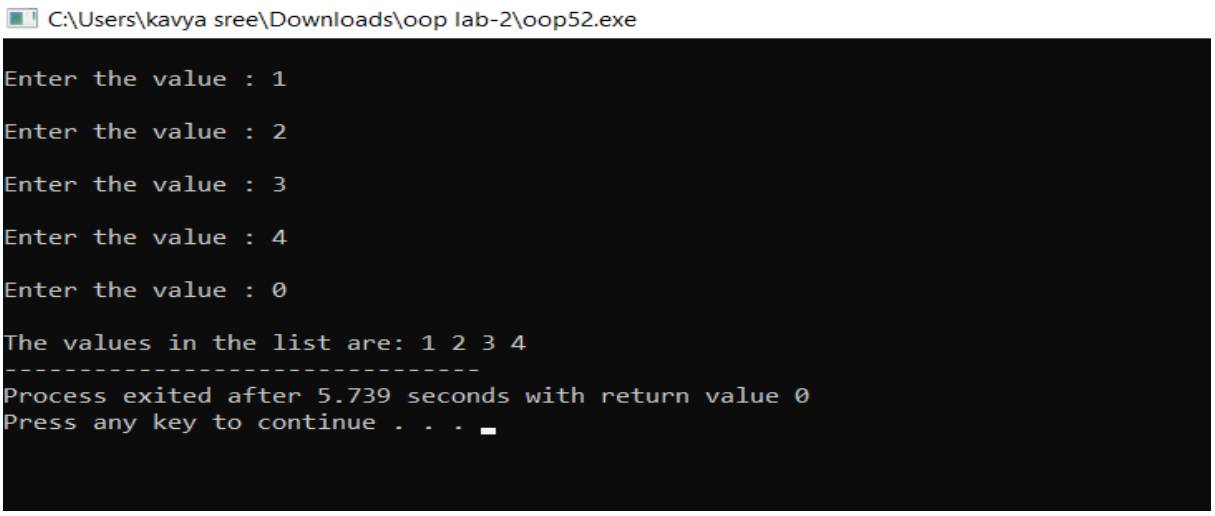
if(h==NULL)

{

h = (struct node*)malloc(sizeof(struct node));

```
temp = h;
}
else
{
temp->next = (struct node*)malloc(sizeof(struct node));
temp = temp->next;
}
temp->data = value;
}
temp->next = NULL;
head = h;
}
void display()
{
cout<<"\nThe values in the list are: ";
for(temp=head;temp!=NULL;temp=temp->next)
{
cout<<temp->data<<" ";
}
}
int main()
{
```

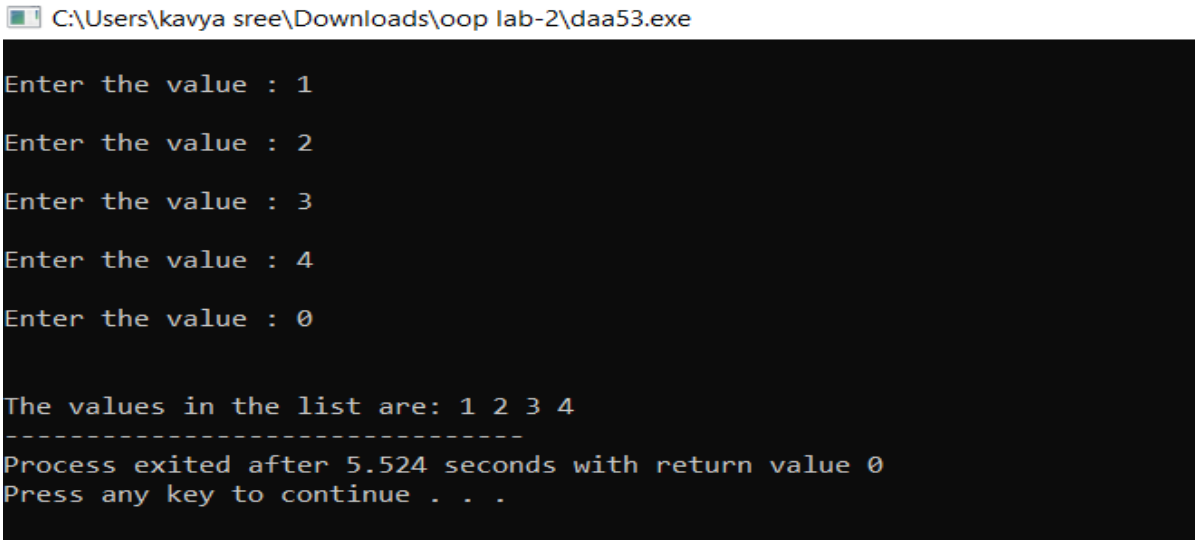
```
createList(head);
display();
}
```



3.
```
#include <iostream>
#include <cstdlib>
using namespace std;
struct node
{
int data;
struct node *next;
};
struct node *head, *temp;
struct node* createList(struct node *head)
{
int value=1;
```

```
while(1)
{
cout<<"\nEnter the value : ";
cin>>value;
if(value==0)
break;
if(head!=NULL)
{
temp->next = (struct node*)malloc(sizeof(struct node));
temp = temp->next;
}
else
{
head = (struct node*)malloc(sizeof(struct node));
temp = head;
}
temp->data = value;
}
temp->next = NULL;
return head;
}
void display(struct node *head)
```

```
{
cout<<"\n\nThe values in the list are: ";
for(temp=head;temp!=NULL;temp=temp->next)
{
cout<<temp->data<<" ";
}
}
int main()
{
head=NULL;
head = createList(head);
display(head);
}
```



4.
```
#include <iostream>
#include <stdlib.h>
```

```
using namespace std;
struct node{
int data;
struct node* next;
};
struct node *head=NULL, *temp;
void createList()
{
int value=1;
while(1)
{
cout<<"\nEnter the value to be inserted : ";
cin>>value;
if(value==0)
break;
if(head!=NULL)
{
temp->next = (struct node*)malloc(sizeof(struct node));
temp = temp->next;
}
else
{
```

```
head = (struct node*)malloc(sizeof(struct node));
temp = head;
}
temp->data = value;
}
temp->next = NULL;
}
void display()
{
cout<<"The values in the list are: ";
for(temp=head;temp!=NULL;temp=temp->next)
{
cout<<temp->data<<" ";
}
}
int deleteElement(int x)
{
for(temp=head;temp!=NULL;temp=temp->next)
{
if(temp->data==x)
{
head = temp->next;
```

```
free(temp);
break;
}
else if(temp->next->data==x)
{
temp->next = temp->next->next;
break;
}
}
if(temp==NULL)
cout<<"Element not found in list"<<endl;
return 0;
}
int main()
{
int del;
createList();
cout<<"\n::Before Deletion::"<<endl;
display();
cout<<"\n\nThe element that you want to delete: ";
cin>>del;
deleteElement(del);
```

```
cout<<"::After Deletion::"<<endl;
display();
return 0;
}
```

C:\Users\kavya sree\Downloads\oop lab-2\daa54.exe

```
Enter the value to be inserted : 1

Enter the value to be inserted : 2

Enter the value to be inserted : 3

Enter the value to be inserted : 4

Enter the value to be inserted : 0

::Before Deletion::
The values in the list are: 1 2 3 4

The element that you want to delete: 2
::After Deletion::
The values in the list are: 1 3 4
--------------------------------
Process exited after 14.35 seconds with return value 0
Press any key to continue . . .
```