Implementation of Singly Linked List In C++

Problem:

Write a menu driven program to illustrate basic operations of Singly Linked List with following operations:

1. Insert at first
2. Insert at last
3. Insert at nth position
4. Delete from first
5. Delete from last
6. Delete from nth position
7. Traverse all the nodes
8. Search any value

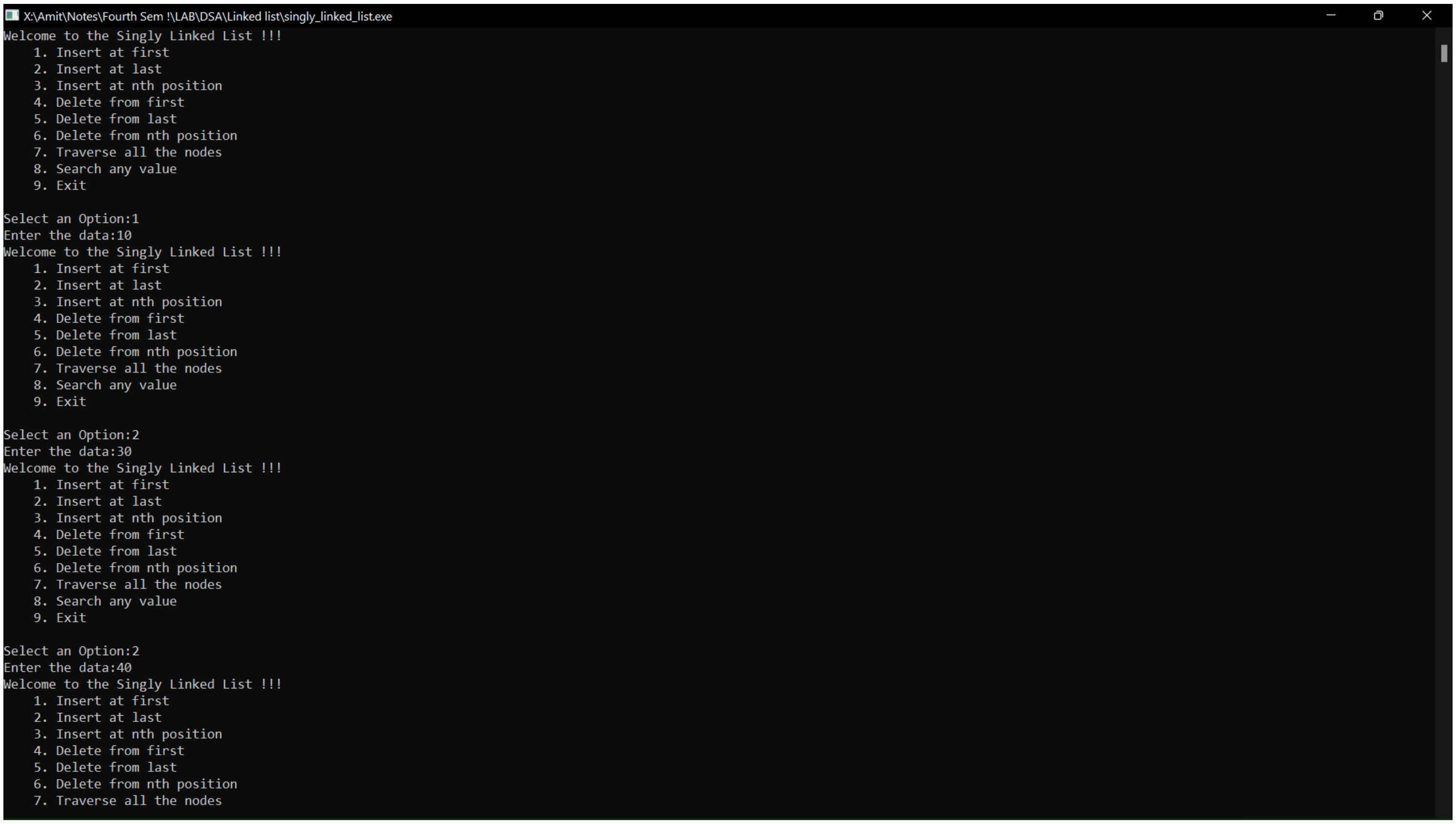
Code:

```
#include<iostream>
using namespace std;
class node
{
public:
  int info;
  node *next;
  void getdata()
  {
    cout<<"Enter the data:";
    cin>>info;
  }
};

node *start=NULL;

//Fuctions
void insert_first()
{
  //empty case
  if (start==NULL)
  {
    node *ptr = new(node);
    // ptr = (node*)malloc(sizeof(node))
    ptr->getdata();
    ptr->next = NULL;
    start = ptr;
  }
  else
  {
    node *ptr = new(node);
    ptr->getdata();
    ptr->next = start;
    start= ptr;

  }

}

void insert_last()
{
  //empty case
  if (start==NULL)
  {
    node *ptr = new(node);
    // ptr = (node*)malloc(sizeof(node))
    ptr->getdata();
    ptr->next = NULL;
    start = ptr;
  }
  else
  {
    node *temp=start;
    while(temp->next!=NULL)
    {
      temp=temp->next;

    }
    node *ptr = new(node);
    ptr->getdata();
    temp->next= ptr;
    ptr->next= NULL;

  }

}

void insert_n()
{
  int n, count=1;
  cout<<"Enter Pos to start after:";
  cin>>n;
  node *ptr = new(node);

  //empty case
  if (start==NULL)
  {
    // ptr = (node*)malloc(sizeof(node))
    ptr->getdata();
    ptr->next = NULL;
    start = ptr;
  }

  //finding the position
  else
  {
    node *temp=start;
    while(count != n)
    {
      temp = temp->next;
      count++;
    }
    ptr->getdata();
    ptr->next= temp->next;
    temp->next= ptr;

  }


}

void del_first()
{
  //Empty list
  if (start == NULL)
    cout<<"The List is empty";
  else
  {
  node *ptr= start;
  start= ptr->next;
  cout<<"\nElement Freed:"<<ptr->info<<endl;
  free(ptr);

  }

}

void del_last()
{
  //Empty list
  if (start == NULL)
    cout<<"The List is empty";
  else
  {
    node *ptr= start;
    node *temp;
    while(ptr->next!=NULL)
    {
      temp=ptr;
      ptr= ptr->next;
    }
    cout<<"\nElement Freed:"<<ptr->info<<endl;
    free(ptr);
    temp->next=NULL;
  }

}

void del_n()
```

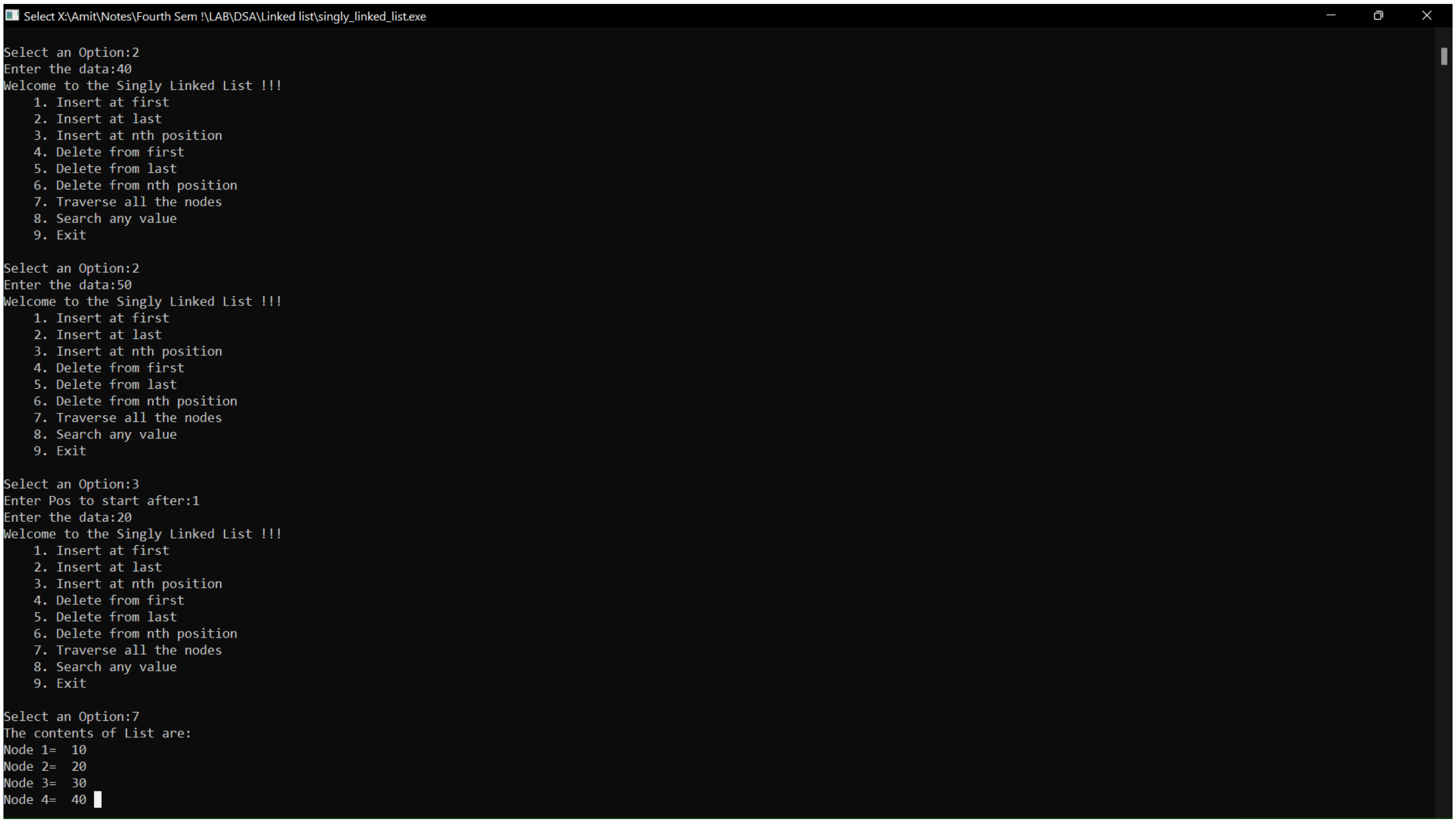
```
{    //Empty list
  int n, count=1;
  node *ptr=start,*temp;
  cout<<"Enter the Pos:";
  cin>>n;
  if (start == NULL)
    cout<<"The List is empty";
  else
  {
    while(count !=n)
    {
      temp=ptr;
      ptr= ptr->next;
      count++;
    }
    temp->next=ptr->next;
    cout<<"\nElement Deleted:"<<ptr->info<<endl;
    free(ptr);
  }
}

void traverse()
{
  int i=1;
  node *ptr=start;
  cout<<"The contents of List are:\n";
  while (ptr!=NULL)
  {
    cout<<"Node "<< i<<"=  "<<ptr->info<<endl;
    ptr=ptr->next;
    i++;

  }
  cout<<endl<<endl;

}

void search()
{
  int data,i=1;
  bool found=false;
  cout<<"Enter the element : ";
  cin>>data;
  node *ptr=start;
  while (ptr!=NULL)
  {
    if(data == ptr->info)
    {
      cout<<"The element found at Pos : "<<i<<endl;
      found= true;
      break;
    }
    ptr=ptr->next;
    i++;
  }
  if(found == false)
  {
    cout<<"The element NOT FOUND in the List\n";
  }

}


int main()
{   int input;
  while(true)
  {
  //capturing Input
    cout<<"Welcome to the Singly Linked List !!!  \n";
    cout<<"    1. Insert at first \n    2. Insert at last \n    3. Insert at nth
position \n    4. Delete from first \n    5. Delete from last\n    6. Delete from
nth position\n    7. Traverse all the nodes\n    8. Search any value\n    9. Exit \
n";
    cout<<"\nSelect an Option:";
    cin>>input;

  //Switch case to perform Respond
    switch(input)
    {
      case 1:
        insert_first();
        break;
      case 2:
        insert_last();
        break;
      case 3:
        insert_n();
        break;
      case 4:
        del_first();
        break;
      case 5:
        del_last();
        break;
      case 6:
        del_n();
        break;
      case 7:
        traverse();
        break;
      case 8:
        search();
        break;
      case 9:
        exit(0);
        break;
    }
  }

  return 0;
```

```
}
```

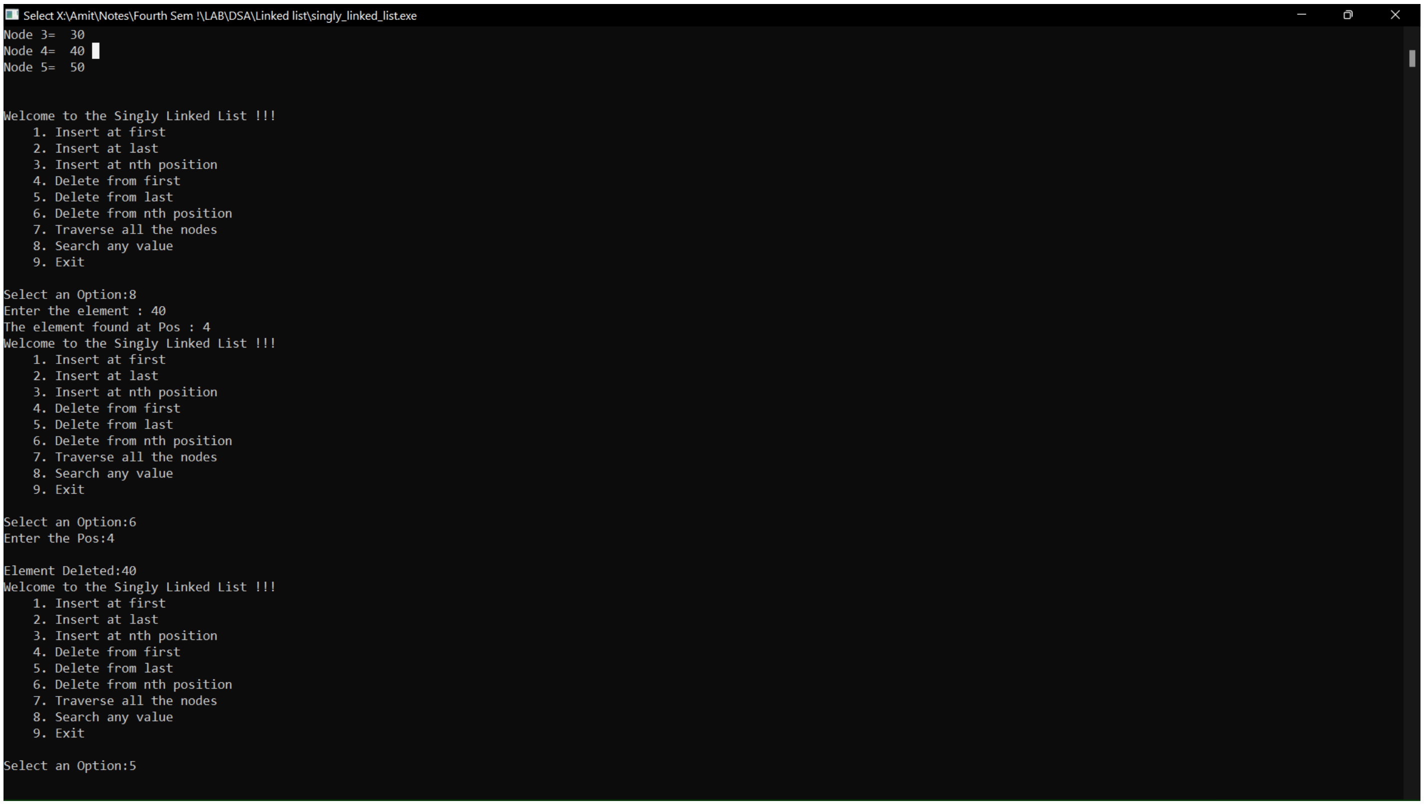
Output:


X:\Amit\Notes\Fourth Sem !\LAB\DSA\Linked list\singly_linked_list.exe

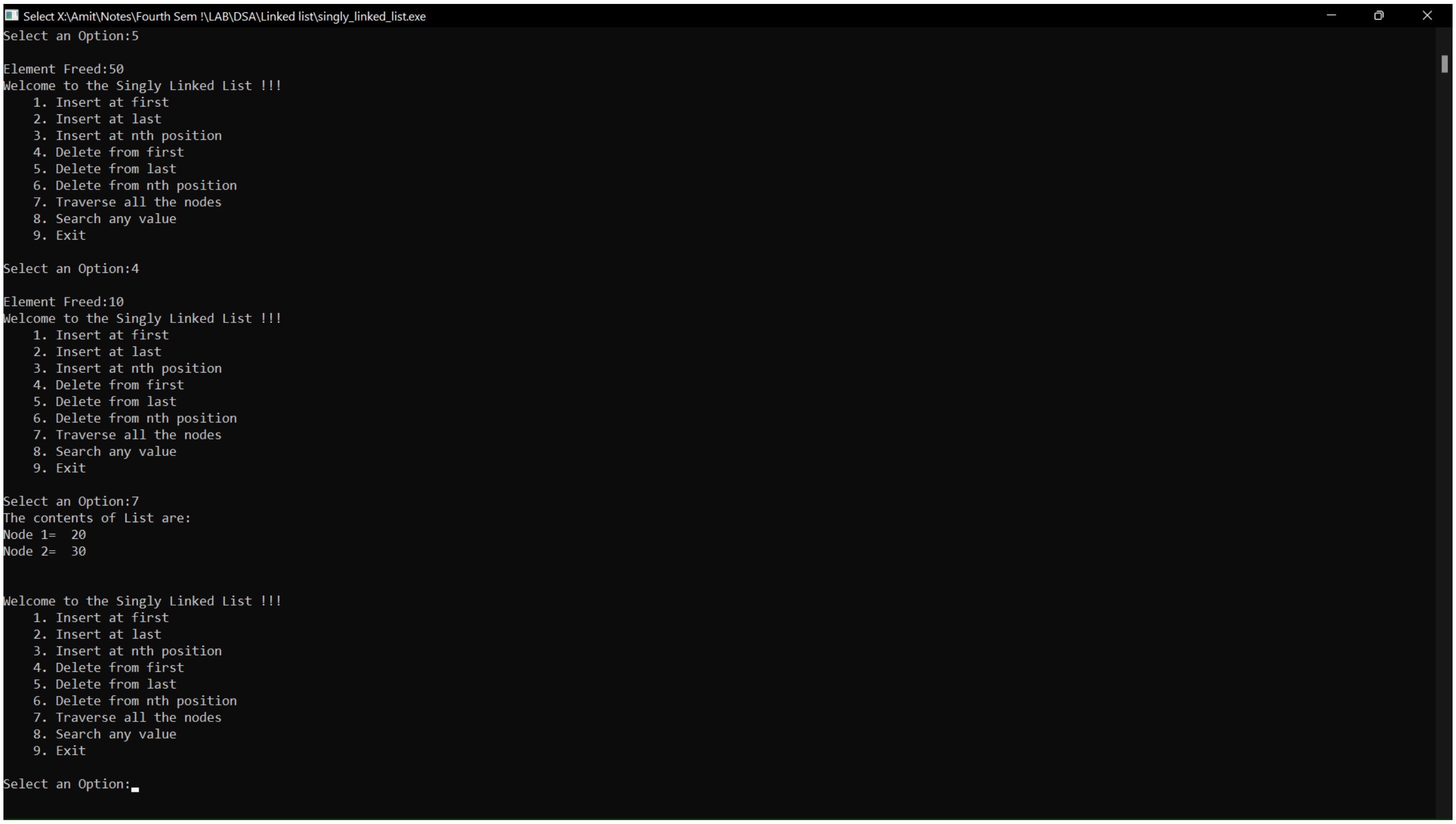
```
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:1
Enter the data:10
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:2
Enter the data:30
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:2
Enter the data:40
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
```


Select X:\Amit\Notes\Fourth Sem !\LAB\DSA\Linked list\singly_linked_list.exe

```
Select an Option:2
Enter the data:40
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:2
Enter the data:50
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:3
Enter Pos to start after:1
Enter the data:20
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:7
The contents of List are:
Node 1=  10
Node 2=  20
Node 3=  30
Node 4=  40
```


Select X:\Amit\Notes\Fourth Sem !\LAB\DSA\Linked list\singly_linked_list.exe

```
Node 3=  30
Node 4=  40
Node 5=  50


Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:8
Enter the element : 40
The element found at Pos : 4
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:6
Enter the Pos:4

Element Deleted:40
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:5
```



Select X:\Amit\Notes\Fourth Sem !\LAB\DSA\Linked list\singly_linked_list.exe

```
Select an Option:5

Element Freed:50
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:4

Element Freed:10
Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:7
The contents of List are:
Node 1=  20
Node 2=  30


Welcome to the Singly Linked List !!!
    1. Insert at first
    2. Insert at last
    3. Insert at nth position
    4. Delete from first
    5. Delete from last
    6. Delete from nth position
    7. Traverse all the nodes
    8. Search any value
    9. Exit

Select an Option:_
```