ATMA RAM SANATAN DHARMA

DSA ASSIGNMENT

-Shishirant Singh

-22/28081

ASSIGNMENT

Q5) Singly linked list:

ANS:

CODE:

```
#include <iostream>
#include <stdio.h>
#include <string>
using namespace std;
class Node{
   public:
   int data;
   Node* next;

   Node(int data){
      this -> data = data;
      next = NULL;
   }
   // INSERTING ELEMENT IN SINGLY LINKED LIST:
   void insertAtBeg(Node* &head, int val){
      if(head==NULL){
         Node* newNode = new Node(val);
         head = newNode;
      }
      else{
         Node* newNode = new Node(val);
         newNode->next = head;
         head = newNode;
      }
   }
   void insertAtEnd(Node* &head, int val){
      if (head == NULL){
         Node* newNode = new Node(val);
         head = newNode;
      }
      else{
         Node* newNode = new Node(val);
         Node* temp = head;
```

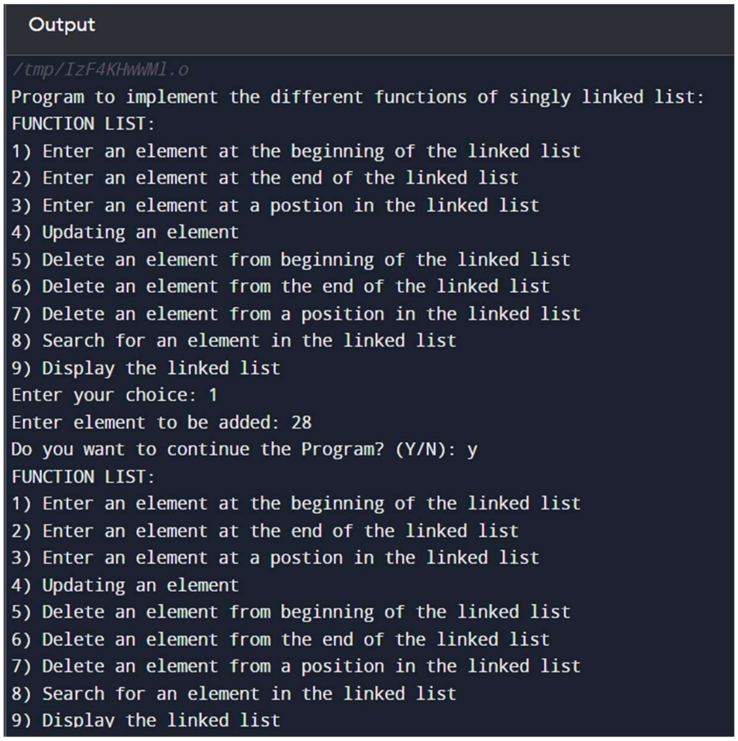
```
        while (temp->next != NULL){
            temp = temp->next;}
        temp->next = newNode;
    }
}
void insertAtPos(Node* &head, int val, int pos){
    if(pos==1){
        insertAtBeg(head,val);
    }
    else if(pos<=0){
        cout<<"Position is out of range!"<<endl;
    }
    else{
        Node* newNode = new Node(val);
        Node* temp = head;
        for(int i=0; i<(pos-2); ++i){
            temp = temp->next;
        }
        newNode->next = temp->next;
        temp->next = newNode;
    }
}
void updateAtPos(Node* &head, int val, int pos){
    Node* temp = head;
    for(int i=0; i<(pos-1);++i){
        temp = temp->next;
    }
    temp->data = val;
}

// DELETING FROM SINGLY LINKED LIST:
void deleteAtBeg(Node* &head){
    if (head==NULL){
        cout<<"No element to delete! Empty list"<<endl;
    }
    else{
        Node* temp = head;
        head = head->next;
        cout<<"Deleted element is: "<<temp->data<<endl;
        delete(temp);
    }
}

void deleteAtEnd(Node* &head){
    if(head==NULL){
        cout<<"Empty list!"<<endl;
    }
    else if(head->next==NULL){
        delete(head);
    }
    else{
        Node* temp = head;
        while(temp->next !=NULL){
            temp = temp->next;
        }
        Node* temp2 = temp->next;
        temp->next = NULL;
        cout<<"Deleted element is: "<<temp2->data<<endl;
        delete(temp2);
    }
}

void deleteAtPos(Node* head, int pos){
    if(pos==1){
        deleteAtBeg(head);
    }
    else if(pos<=0){
        cout<<"Position out of range!"<<endl;
    }
    else{
        Node* temp = head;
        for(int i=0;i<(pos-2);++i){
            temp = temp->next;
        }
        Node* temp2 =  temp->next;
        temp->next = temp2->next;
        cout<<"Deleted element: "<<temp2->data<<endl;
        delete(temp2);
    }
}
```

```
    void search(Node* head, int val){
        Node* temp = head;
        Node* temp2;
        int i=1;
        while(temp != NULL){
            if(temp->data == val){
                break;
            }
            else{
                temp2 = temp;
                temp = temp->next;
                i++;
            }
        }
        if(temp !=NULL){
            cout<<"The pointer of the element "<<val<<" in the linked is "<<temp2-
>next<<" and is at the position "<<i<<endl;
        }
        else{
            cout<<"The element "<<val<<" is not present in the linked list."<<endl;
        }
    }

    void displayLL(Node* head){
        Node* temp = head;
        while(temp !=NULL){
            cout<<temp->data<<" -> ";
            temp = temp->next;
        }
        cout<<"Null"<<endl;

    }
};

int main(){
    cout<<"Program to implement the different functions of singly linked list:
"<<endl;
    char ans = 'y';
    Node* head = NULL;
```

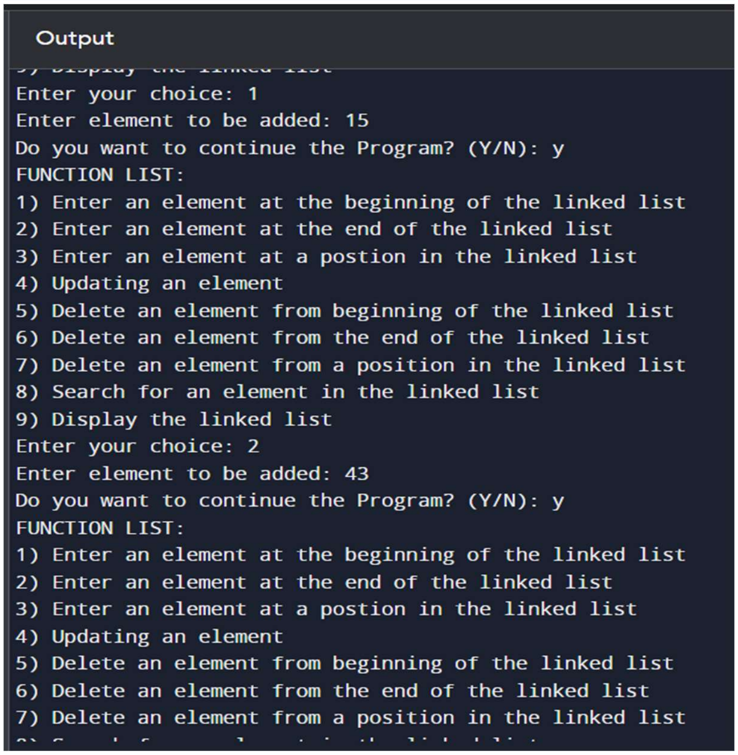
```
while(ans == 'y' || ans == 'Y'){
    cout<<"FUNCTION LIST: "<<endl;
    char choice;
    cout<<"1) Enter an element at the beginning of the linked list "<<endl;
    cout<<"2) Enter an element at the end of the linked list "<<endl;
    cout<<"3) Enter an element at a postion in the linked list "<<endl;
    cout<<"4) Updating an element "<<endl;
    cout<<"5) Delete an element from beginning of the linked list "<<endl;
    cout<<"6) Delete an element from the end of the linked list "<<endl;
    cout<<"7) Delete an element from a position in the linked list "<<endl;
    cout<<"8) Search for an element in the linked list "<<endl;
    cout<<"9) Display the linked list "<<endl;
    cout<<"Enter your choice: ";
    cin>>choice;
    switch(choice)
    {
        case '1':
            cout<<"Enter element to be added: ";
            int val1;
            cin>>val1;
            head->insertAtBeg(head,val1);
            break;
        case '2':
            cout<<"Enter element to be added: ";
            int val2;
            cin>>val2;
            head->insertAtEnd(head,val2);
            break;
        case '3':
            cout<<"Enter element to be added: ";
            int val3;
            cin>>val3;
            cout<<"Enter the position for new element: ";
            int pos1;
            head->insertAtPos(head,val3,pos1);
            break;
        case '4':
            cout<<"Enter the position to be updated: ";
            int pos2;
            cin>>pos2;
        cout<<"Enter element to be updated with: ";
        int val4;
        cin>>val4;
        head->updateAtPos(head,val4,pos2);
        break;
    case '5':
        head->deleteAtBeg(head);
        break;
    case '6':
        head->deleteAtEnd(head);
        break;
    case '7':
        cout<<"Enter the position from where you want to delete: ";
        int pos3;
        cin>>pos3;
        head->deleteAtPos(head,pos3);
        break;
    case '8':
        cout<<"Enter element to be searched: ";
        int val;
        cin>>val;
        head->search(head,val);
        break;
    case '9':
        head->displayLL(head);
        break;

    default:
        break;
}

cout<<"Do you want to continue the Program? (Y/N): ";
cin>>ans;
cout<<endl;}
if(ans == 'n' || ans == 'N'){
    cout<<"----------PROGRAM EXITED----------"<<endl;
}
else{
    cout<<"Invalid choice!"<<endl;
    main();
```

```
}
return 0;
}
```

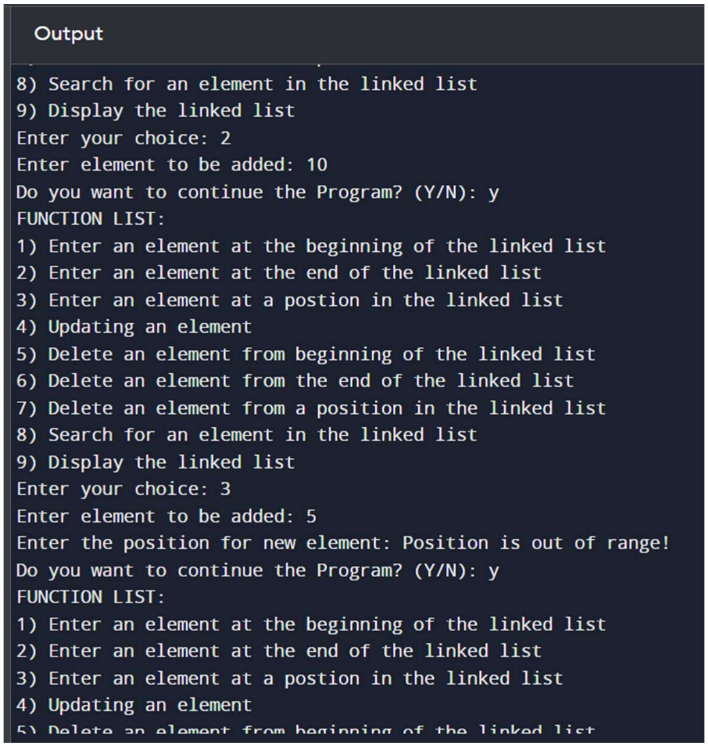
OUTPUT:



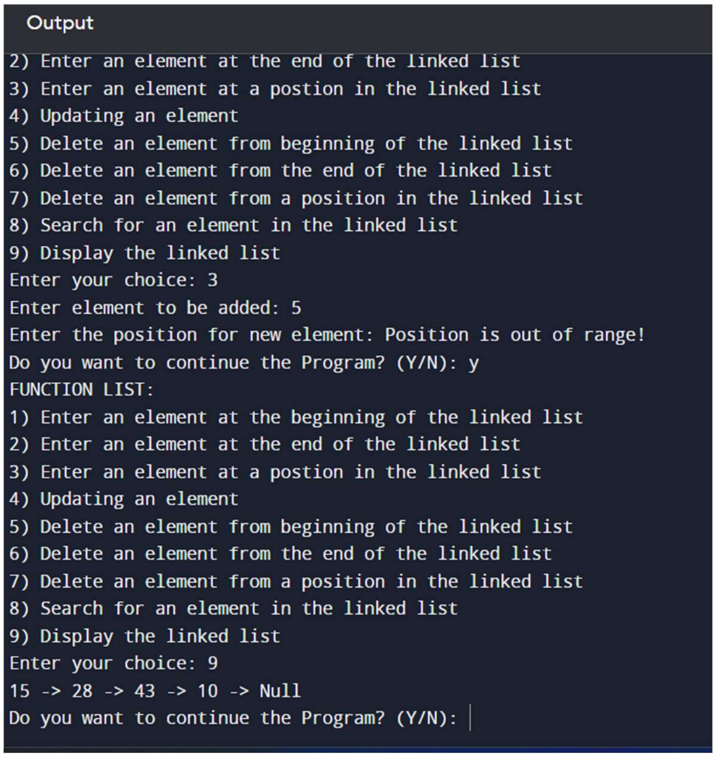
```
/tmp/IzF4KHwWMl.o
Program to implement the different functions of singly linked list:
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
9) Display the linked list
Enter your choice: 1
Enter element to be added: 28
Do you want to continue the Program? (Y/N): y
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
9) Display the linked list
```



```
9) Display the linked list
Enter your choice: 1
Enter element to be added: 15
Do you want to continue the Program? (Y/N): y
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
9) Display the linked list
Enter your choice: 2
Enter element to be added: 43
Do you want to continue the Program? (Y/N): y
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
```



```
8) Search for an element in the linked list
9) Display the linked list
Enter your choice: 2
Enter element to be added: 10
Do you want to continue the Program? (Y/N): y
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
9) Display the linked list
Enter your choice: 3
Enter element to be added: 5
Enter the position for new element: Position is out of range!
Do you want to continue the Program? (Y/N): y
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
```



```
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
9) Display the linked list
Enter your choice: 3
Enter element to be added: 5
Enter the position for new element: Position is out of range!
Do you want to continue the Program? (Y/N): y
FUNCTION LIST:
1) Enter an element at the beginning of the linked list
2) Enter an element at the end of the linked list
3) Enter an element at a postion in the linked list
4) Updating an element
5) Delete an element from beginning of the linked list
6) Delete an element from the end of the linked list
7) Delete an element from a position in the linked list
8) Search for an element in the linked list
9) Display the linked list
Enter your choice: 9
15 -> 28 -> 43 -> 10 -> Null
Do you want to continue the Program? (Y/N):
```