Ex.No:1 Implementation of Singly Linked List

Program

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
typedef struct link
{
        int data;
        struct link *next;
}node;


void main()
{
        int f,no,loc,i,choice,loop;
        void insert_beg(int,node **);
        void insert_after(int,int,node **);
        void insert_end(node **,int);
        int find(int,node *);
        void display(node *);
        void delete_node(int,node **);
        int count_nodes(node *);
        void reverse_list(node **);
        node *head;
        clrscr();
        head=(node *)malloc(sizeof(node));
        head->data=0;
        head->next=NULL;

        do
        {
                printf("\nMenu\n1.Insert Beggining\n2.Insert in middle\n3.Insert at
                End\n4.Find\n5.Display List\n6.Delete Node\n7.Count Nodes\n8.Reverse
                List\n9.Exit");
                printf("\nEnter your Choice:");
                scanf("%d",&choice);
                switch(choice)
                {
                        case 1:
                                printf("Enter the element to be inserted:");
                                scanf("%d",&no);
                                insert_beg(no,&head);
                                break;
                        case 2:
                                printf("Enter the element to be inserted and location:");
```

```
                    scanf("%d%d",&no,&loc);
                    insert_after(no,loc,&head);
                    break;
                case 3:
                    printf("Enter the element to be inserted:");
                    scanf("%d",&no);
                    insert_end(&head,no);
                    break;
                case 4:
                    printf("Enter the element to be searched:");
                    scanf("%d",&no);
                    f=find(no,head);
                    if(f==1)
                        printf("Element found.");
                    else
                        printf("Element not found.");
                    break;
                case 5:
                    printf("The list of elements found in the LIst are:");
                    display(head);
                    break;
                case 6:
                    printf("Enter the element to be deleted:");
                    scanf("%d",&no);
                    delete_node(no,&head);
                    break;
                case 7:
                    printf("Number of elements present in the list are:");
                    count_nodes(head);
                    break;
                case 8:
                    printf("Reversing of the list is:");
                    reverse_list(&head);
                    display(head);
                    break;
                default:
                    exit(0);
            }
            printf("\nDo you wants to continue(Press 1 for YES):");
            scanf("%d",&loop);
        }while(loop==1);
getch();
}
```

```
void insert_end(node **head, int no)
{
	node *temp,*new1;
	temp=*head;
	if(temp==NULL)
	{
		temp=(node *)malloc(sizeof(node));
		temp->data=no;
		temp->next=NULL;
		*head=temp;
	}
	else
	{
		while(temp->next!=NULL)
			temp=temp->next;
		new1=(node*)malloc(sizeof(node));
		new1->data=no;
		new1->next=NULL;
		temp->next=new1;
	}
}

void display(node *head)
{
	if(head==NULL)
	{
		printf("LIst is empty");;
		return;
	}
	while(head!=NULL)
	{
		printf("%d\t",head->data);
		head=head->next;
		if(head==NULL)
			return;
	}
}

int find(int x,node *head)
{
	node *temp;
	temp=head;
	if(temp==NULL)
	{
		printf("LIst is empty");
		return NULL;
```

```
        }
        while(temp!=NULL&&temp->data!=x)
        {
                temp=temp->next;
        }
        return temp->data;
}

void insert_beg(int x,node **head)
{
        node *new1;
        new1=(node *)malloc(sizeof(node));
        new1->data=x;
        new1->next=*head;
        *head=new1;
}

void insert_after(int x,int loc,node **head)
{
        node *temp,*new1;
        int i;
        temp=*head;
        for(i=0;i<loc;i++)
        {
                temp=temp->next;
                if(temp==NULL)
                {
                        printf("There are less than %d elements in the list",loc);
                        return;
                }
        }
        new1=(node *)malloc(sizeof(node));
        new1->data=x;
        new1->next=temp->next;
        temp->next=new1;
}

void delete_node(int x,node **head)
{
        node *old,*temp;
        temp=*head;
        while(temp!=NULL)
        {
                if(temp->data==x)
                {
                        if(temp==*head)
                        *head=temp->next;
```

```
                else
                {
                        old->next=temp->next;
                        temp->next=NULL;
                }
                free(temp);
                return;
                }
                else
                {
                        old=temp;
                        temp=temp->next;
                }
        }
        printf("\nElement %d not found",x);
}

int count_nodes(node *head)
{
        int count=0;
        node *temp;
        if(head==NULL)
                return count;
        temp=head;
        while(temp!=NULL)
        {
                count++;
                temp=temp->next;
        }
        printf("\n%d",count);
}

void reverse_list(node **head)
{
        node *temp,*s,*rear;
        temp=*head;
        rear=NULL;
        while(temp!=NULL)
        {
                s=rear;
                rear=temp;
                temp=temp->next;
                rear->next=s;
        }
        *head=rear;
}
```

Output:

```
Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice:5
The list of elements found in the LIst are: 0
Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 7
Number of elements present in the list are:
1
Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit

Enter your Choice: 1
Enter the element to be inserted: 1

Do you wants to continue(Press 1 for YES): 1
```

```
Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice:3
Enter the element to be inserted: 2

Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 5
The list of elements found in the LIst are:  1    0       2

Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 2
Enter the element to be inserted and location: 1 1

Do you wants to continue(Press 1 for YES): 1
```

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 5
The list of elements found in the LIst are: 1 0 1 2
Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 3
Enter the element to be inserted: 3

Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit

Enter your Choice:5
The list of elements found in the LIst are: 1 0 1 2 3

Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice:4
Enter the element to be searched:1
Element found.

Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 6
Enter the element to be deleted: 1

Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit

Enter your Choice: 5
The list of elements found in the LIst are: 0 1 2 3
Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 7
Number of elements present in the list are:
4
Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 8
Reversing of the list is: 3 2 1 0
Do you wants to continue(Press 1 for YES): 1

Menu
1.Insert Beggining
2.Insert in middle
3.Insert at End
4.Find
5.Display List
6.Delete Node
7.Count Nodes
8.Reverse List
9.Exit
Enter your Choice: 9