***Aditya Ramakrishnan***

***Div:C-1***

***PRN:119A1004***

***Experiment no-6***

# Aim:: Implementation of Singly Linked List by the concept of ADT.

**Problem statement:**

Implement singly linked list and following operations- 1.

InsertBeg()- insert new node at the start and print message " inserted successfully" 2.

InsertEnd()- - insert new node at the end and print message " inserted successfully" 3

Deletenode()- delete a node with the given value.

Output: If list is empty, print "List empty" If node with given value found then print "deleted successfully, else print "node not found"

Display()- Print the data stored in the linked list.

**Program:**

#include<stdio.h>

#include<stdlib.h>

struct node

{

int data;

struct node \*next;

}\*s=0,\*q,\*t;

int main()

{

int ch;

void inst\_beg();

void inst\_end();

int del\_pos();

void display();

printf("\n---- Insert Menu ----");

printf("\n1.Insert at beginning of the node\n2.Insert at end of the noe\n3.Delete at specified position of the node\n4.Display data stored in linked list\n5.Exit the program");

while(1)

{

printf("\n\nEnter your choice:");

scanf("%d",&ch);

switch(ch)

{

case 1: inst\_beg();

break;

case 2: inst\_end();

break;

case 3: del\_pos();

break;

case 4: display();

break;

case 5: exit(0);

default: printf("Enter valid the choice!");

}

}

}

void inst\_beg()

{

int num;

t=(struct node\*)malloc(sizeof(struct node));

printf("Enter data:");

scanf("%d",&num);

t->data=num;

if(s==0)

{

t->next=0;

s=t;

}

else

{

t->next=s;

s=t;

}

printf("Inserted successfully");

}

void inst\_end()

{

int num;

t=(struct node\*)malloc(sizeof(struct node));

printf("Enter data:");

scanf("%d",&num);

t->data=num;

t->next=NULL;

if(s==0)

{

s=t;

}

else

{

q=s;

while(q->next!=NULL)

q=q->next;

q->next=t;

}

printf("Inserted successfully");

}

int del\_pos()

{

int pos,i;

if(s==0)

{

printf("List is empty!!");

return 0;

}

printf("Enter position to delete:");

scanf("%d",&pos);

q=s;

for(i=1;i<pos-1;i++)

{

if(q->next==NULL)

{

printf("There are less elements!!");

return 0;

}

q=q->next;

}

t=q->next;

q->next=t->next;

printf("Deleted element is %d",t->data);

return 0;

}

void display()

{

if(s==0)

{

printf("List is empty!!");

}

else

{

q=s;

printf("The linked list is:\n");

while(q!=0)

{

printf("%d->",q->data);

q=q->next;

}

}

}

***Output:***

