**Program to implement doubly Linked List.**

#include <stdio.h>

#include<stdlib.h>

void create();

void display();

void insertBegin();

void insertEnd();

void deleteByValue();

struct node

{

int data;

struct node \*prev;

struct node \*next;

};

struct node \*head=NULL;

int main()

{

int choice;

printf(" ====== MENU ====== \n");

printf(" 1.Create \n");

printf(" 2.Display \n");

printf(" 3.Insert at the beginning \n");

printf(" 4.Insert at the end \n");

printf(" 5.delete by value \n");

printf(" 6.Exit \n");

while(choice != 6){

printf("\nEnter your choice: ");

scanf("%d",&choice);

switch(choice){

case 1:

create();

break;

case 2:

display();

break;

case 3:

insertBegin();

break;

case 4:

insertEnd();

break;

case 5: deleteByValue();

break;

case 6: printf("Exiting....\n");

break;

default:

printf("Wrong Choice \n");

break;

}

}

return 0;

}

void create()

{

struct node \*newnode,\*temp;

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

printf("Enter the data value of the newnode:");

scanf("%d",&newnode->data);

newnode->next=NULL;

if(head==NULL)

{

head=newnode;

newnode->prev=NULL;

}

else

{

newnode->next=head;

head->prev=newnode;

newnode->prev=NULL;

head=newnode;

}

}

void display()

{ struct node \*ptr;

ptr = head;

if(head==NULL){

printf("Linked list is empty\n");

}

else{

while(ptr != NULL)

{

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

ptr=ptr->next;

}

}

}

void insertBegin()

{

struct node \*newnode;

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

printf("\nEnter the data value for the node: " );

scanf("%d",&newnode->data);

newnode->next =NULL;

if(head==NULL)

{

head=newnode;

newnode->prev=NULL;

}

else

{

newnode->next=head;

head->prev=newnode;

newnode->prev=NULL;

head=newnode;

}

}

void insertEnd()

{

struct node \*newnode,\*temp;

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

printf("\nEnter the data value for the node: " );

scanf("%d",&newnode->data );

newnode->next =NULL;

if(head==NULL)

{

head=newnode;

newnode->prev=NULL;

}

else

{

temp=head;

while(temp->next !=NULL)

{

temp=temp->next ;

}

temp->next =newnode;

newnode->prev=temp;

}

}

void deleteByValue()

{

struct node \*temp;

int del\_data;

printf("please enter data item which you wish to delete: ");

scanf("%d",&del\_data);

temp=head;

while(temp!=NULL)

{

if(temp->data==del\_data)

{

if(temp->prev==NULL && temp->next==NULL)//if only node

{

free(temp);

head=NULL;

break;

}

if(temp->prev==NULL)//if first node

{

temp->next->prev=NULL;

head=temp->next;

printf("element deleted!!!\n");

free(temp);

break;

}

if(temp->next==NULL)//if last node

{

temp->prev->next=NULL;

printf("element deleted!!!\n");

free(temp);

break;

}

temp->prev->next=temp->next;

temp->next->prev=temp->prev;

printf("element deleted!!!\n");

free(temp);

break;

}

temp=temp->next;

}}

