**DOUBLY LINKED LIST**

#include<stdio.h>

#include<stdlib.h>

struct node

{ int data;

struct node \*next;

struct node \*prev; };

struct node \*start=NULL;

struct node \*create(struct node \*);

struct node \*display(struct node \*);

struct node \*beginning(struct node \*);

struct node \*ending(struct node \*);

struct node \*before(struct node \*);

struct node \*after(struct node \*);

struct node \*display1(struct node \*);

struct node \*beginning1(struct node \*);

struct node \*ending1(struct node \*);

struct node\*before1(struct node \*);

struct node\*after1(struct node \*);

int main()

{

int c1,c2;

char ch='Y';

do{

printf("ENTER A CHOICE :\n");

printf("1)CREATE LINKED LIST \n");

printf("2)DISPLAY LINKED LIST \n");

printf("3)ENTER AN ELEMENT \n");

printf("4)DELETE AN ELEMENT \n");

scanf("%d",&c1);

switch(c1)

{ case 1:start=create(start);

break;

case 2:start=display(start);

break;

case 3:printf("ENTER A CHOICE :\n");

printf("1)BEGINNING \n2)END \n3)BEFORE A GIVEN NODE \n4)AFTER A GIVEN NODE\n");

scanf("%d",&c2);

switch(c2)

{ case 1: start=beginning(start);

break;

case 2: start=ending(start);

break;

case 3: start=before(start);

break;

case 4: start=after(start);

break;

}

break;

case 4:printf("ENTER A CHOICE :\n");

printf("1)BEGINNING \n2)END \n3)BEFORE A GIVEN NODE \n4)AFTER A GIVEN NODE\n");

scanf("%d",&c2);

switch(c2)

{

case 1: start=beginning1(start);

break;

case 2: start=ending1(start);

break;

case 3: start=before1(start);

break;

case 4: start=after1(start);

break;

}

break;

}

printf("DO YOU WANT TO CONTINUE(Y/N?");

scanf("%s",&ch);

}while(ch=='Y'||ch=='y');

}

struct node \*create(struct node \*start)

{

struct node \*new\_node,\*ptr;

int num;

char ch1='Y';

do

{

printf("ENTER DATA IN THE LIST");

scanf("%d",&num);

new\_node=(struct node\*)malloc(sizeof(struct node));

new\_node->data=num;

if(start==NULL)

{

new\_node->next=NULL;

new\_node->prev=NULL;

start=new\_node;

}

else

{

ptr=start;

while(ptr->next!=NULL)

{

ptr=ptr->next;

}

ptr->next=new\_node;

new\_node->prev=ptr;

new\_node->next=NULL;

}

printf("DO YOU WANT TO ENTER MORE(Y/N)?");

scanf("%s",&ch1);

}while(ch1=='Y'||ch1=='y');

return start; }

struct node \*display(struct node \*start)

{

struct node \*ptr;

ptr=start;

while(ptr->next!=start)

{

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

ptr=ptr->next;

}

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

return start;

}

struct node \*beginning(struct node \*start)

{

struct node \*new\_node,\*ptr;

int num;

printf("ENTER DATA : ");

scanf("%d",&num);

new\_node=(struct node\*)malloc(sizeof(struct node));

new\_node->data=num;

ptr=start;

while(ptr->next!=start)

{

ptr=ptr->next;

}

start->prev=new\_node;

ptr->next=new\_node;

new\_node->prev=NULL;

start=new\_node;

return start;

}

struct node \*ending(struct node \*start)

{

struct node \*new\_node,\*ptr;

int num;

printf("ENTER DATA : ");

scanf("%d",&num);

new\_node=(struct node\*)malloc(sizeof(struct node));

new\_node->data=num;

start=ptr;

while(ptr->next!=NULL)

{

ptr=ptr->next;

}

ptr->next=new\_node;

new\_node->next=NULL;

new\_node->prev=ptr;

return start;

}

struct node\*before(struct node \*start)

{

struct node \*new\_node,\*ptr,\*preptr;

int num,val;

printf("ENTER DATA : ");

scanf("%d",&num);

new\_node=(struct node\*)malloc(sizeof(struct node));

new\_node->data=num;

printf("ENTER THE VALUE BEFORE WHICH THE DATA HAS TO BE INSERTED: ");

scanf("%d",&val);

ptr=start;

while(ptr->data!=val)

{

preptr=ptr;

ptr=ptr->next;

}

preptr->next=new\_node;

new\_node->next=ptr;

new\_node->prev=ptr->prev;

ptr->prev->next=new\_node;

ptr->prev=new\_node;

return start;

}

struct node\*after(struct node \*start)

{

struct node \*new\_node,\*ptr;

int num,val;

printf("ENTER DATA : ");

scanf("%d",&num);

new\_node=(struct node\*)malloc(sizeof(struct node));

new\_node->data=num;

printf("ENTER THE VALUE AFTER WHICH THE DATA HAS TO BE INSERTED: ");

scanf("%d",&val);

ptr=start;

while(ptr->data!=val)

ptr=ptr->next;

new\_node->next=ptr->next;

new\_node->prev=ptr;

ptr->next->prev=new\_node;

ptr->next=new\_node;

return start;

}

struct node\*beginning1(struct node \*start)

{ struct node \*ptr;

ptr=start;

start=start->next;

start->prev=NULL;

free (ptr);

return start; }

struct node\*ending1(struct node \*start)

{ struct node \*ptr;

ptr=start;

while(ptr->next!=NULL)

ptr=ptr->next;

ptr->prev->next=NULL;

free (ptr);

return start; }

struct node\*before1(struct node \*start)

{ int val;

struct node \*ptr,\*preptr;

printf("ENTER THE VALUE BEFORE WHICH THE NODE HAS TO BE DELETED: ");

scanf("%d",&val);

ptr=start;

while(ptr->data!=val)

{ preptr=ptr;

ptr=ptr->next; }

preptr->prev->next=preptr->next;

ptr->prev=preptr->prev;

free (preptr);

return start; }

struct node\*after1(struct node \*start)

{ int val;

struct node \*ptr,\*preptr;

printf("ENTER THE VALUE AFTER WHICH THE NODE HAS TO BE DELETED: ");

scanf("%d",&val);

ptr=start;

preptr=ptr;

while(preptr->data!=val)

{

preptr=ptr;

ptr=ptr->next; }

preptr=ptr;

ptr=ptr->next;

preptr->next=ptr->next;

preptr->next->prev=preptr;

free (ptr);

return start; }

**OUTPUT**

ENTER A CHOICE :

1)CREATE LINKED LIST

2)DISPLAY LINKED LIST

3)ENTER AN ELEMENT

4)DELETE AN ELEMENT

1

ENTER DATA IN THE LIST1

DO YOU WANT TO ENTER MORE(Y/N?y

ENTER DATA IN THE LIST2

DO YOU WANT TO ENTER MORE(Y/N?y

ENTER DATA IN THE LIST3

DO YOU WANT TO ENTER MORE(Y/N?y

ENTER DATA IN THE LIST4

DO YOU WANT TO ENTER MORE(Y/N?n

DO YOU WANT TO CONTINUE(Y/N?y

ENTER A CHOICE :

1)CREATE LINKED LIST

2)DISPLAY LINKED LIST

3)ENTER AN ELEMENT

4)DELETE AN ELEMENT

3

ENTER A CHOICE :

1)BEGINNING

2)END

3)BEFORE A GIVEN NODE

4)AFTER A GIVEN NODE

1

ENTER DATA : 9

DO YOU WANT TO CONTINUE(Y/N?y

ENTER A CHOICE :

1)CREATE LINKED LIST

2)DISPLAY LINKED LIST

3)ENTER AN ELEMENT

4)DELETE AN ELEMENT

2

9

1

2

3

4

DO YOU WANT TO CONTINUE(Y/N?N