**Doubly Link List**

#include<stdio.h>

#include<stdlib.h>

struct node

{

struct node \*prev;

int data;

struct node \*next;

}\*head=NULL,\*tail=NULL;

void create\_D\_ll();

void display();

void f\_display();

void b\_display();

void f\_create\_D\_ll();

void b\_create\_D\_ll();

void f\_insert\_mid();

void b\_insert\_mid(int);

int count();

void main()

{

create\_D\_ll();

display();

f\_create\_D\_ll();

display();

b\_create\_D\_ll();

display();

/\*f\_insert\_mid();

display();\*/

int num;

num=count();

printf("\nNumber of nodes in the list is: %d\n",num);

b\_insert\_mid(num);

display();

}

void create\_D\_ll()

{

struct node \*newnode,\*current;

int i,n;

printf("Enter number of node:\n");

scanf("%d",&n);

for(i=1;i<=n;i++)

{

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

if(newnode==NULL)

{

printf("Error!Memory not allocated.");

exit(0);

}

printf("Enter data of node %d\n",i);

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

newnode->prev=NULL;

newnode->next=NULL;

if(head==NULL)

{

head=newnode;

current=newnode;

tail=newnode;

}

else

{

current->next=newnode;

newnode->prev=current;

current=newnode;

tail=newnode;

}

}

}

void f\_display()

{

struct node \*temp;

temp=head;

printf("\nForward Link List is:\n");

while(temp!=NULL)

{

printf("\t-->%d ",temp->data);

temp=temp->next;

}

}

void b\_display()

{

struct node \*temp;

temp=tail;

if(head==NULL)

{

printf("\nThere is no node in the list:\n");

}

printf("\nBackward Link List is:\n");

while(temp!=NULL)

{

printf("\t-->%d ",temp->data);

temp=temp->prev;

}

}

void display()

{

f\_display();

b\_display();

}

void f\_create\_D\_ll()

{

struct node \*newnode,\*current;

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

if(newnode==NULL)

{

printf("Error!Memory not allocated.");

exit(0);

}

printf("\nEnter data of newnode which will inserted at first in forward link list:\n");

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

newnode->next=NULL;

newnode->prev=NULL;

current=head;

newnode->next=current;

current->prev=newnode;

head=newnode;

}

void b\_create\_D\_ll()

{

struct node \*newnode,\*current;

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

if(newnode==NULL)

{

printf("Error!Memory not allocated.");

exit(0);

}

printf("\nEnter data of newnode which will inserted at first from backward link list:\n");

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

newnode->prev=NULL;

newnode->next=NULL;

current=tail;

while(current!=head)

{

current=current->prev;

}

newnode->next=current;

current->prev=newnode;

head=newnode;

}

void f\_insert\_mid()

{

struct node \*newnode,\*temp1,\*temp2;

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

if(newnode==NULL)

{

printf("Error!Memory not allocated.");

exit(0);

}

int i,pos;

printf("\nEnter position:\n");

scanf("%d",&pos);

printf("\nEnter data of midnode to insert at position %d in forward insert:\n",pos);

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

newnode->prev=NULL;

newnode->next=NULL;

temp1=head;

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

{

temp1=temp1->next;

}

temp2=temp1->next;

temp1->next=newnode;

newnode->prev=temp1;

newnode->next=temp2;

}

void b\_insert\_mid(int num)

{

struct node \*newnode,\*temp1,\*temp2;

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

if(newnode==NULL)

{

printf("Error!Memory not allocated.");

exit(0);

}

int i,pos;

printf("\nEnter position:\n");

scanf("%d",&pos);

printf("\nEnter data of midnode for at position %d in backward insert:\n",pos);

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

newnode->prev=NULL;

newnode->next=NULL;

temp1=tail;

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

{

temp1=temp1->prev;

}

temp2=temp1->prev;

temp2->next=newnode;

newnode->prev=temp2;

newnode->next=temp1;

temp1->prev=newnode;

}

int count()

{

struct node \*temp;

int count=0;

temp=head;

while(temp!=NULL)

{

count++;

temp=temp->next;

}

return count;

}