Linked list

# include <iostream>

# include <process.h>

# include <conio.h>

# include <malloc.h>

using namespace std;

//Linked list structure

struct node

{

int data; // will store information

node \*next; // the reference to the next node

}\*start;

typedef struct node node;

node \*head=NULL;

void display();

void insertend();

void insertbeg();

void delend();

void delbeg();

void insertmid();

void delmid();

void modify();

int main()

{

int a;

//clrscr();

cout<<(" THIS PROGRAM GIVES YOU THE SINGLE LINKLIST\n");

do

{

cout<<("enter your choice\n");

cout<<("1.INSERT element at the END of linklist\n");

cout<<("2.INSERT element at the BEGINING of linklist\n3.DELETE from");

cout<<("END\n4.DELETE from BEGINING\n5.INSERT at MIDDLE\n");

cout<<("6.DELETE from MIDDLE\n7.MODIFY any element\n8.DISPLAY\n9.EXIT\n");

//fflush(stdin);

cin>>a;

switch(a)

{

case 1:

insertend();

break;

case 2:

insertbeg();

break;

case 3:

delend();

break;

case 4:

delbeg();

break;

case 5:

insertmid();

break;

case 6:

delmid();

break;

case 7:

modify();

break;

case 8:

display();

break;

case 9:

exit(0);

return 0;

}

}

while(a!=8);

getch();

}

void insertend()

{

node \*p,\*q;

int item;

cout<<("enter your elements in the stack\n");

cin>>item;

p=(node \*) malloc(sizeof(node));

p->data=item;

p->next=NULL;

if(start==NULL)

{

start=p;

}

else

{

q=start;

while(q->next!=NULL)

{

q=q->next;

}

q->next=p;

}

}

void display()

{

node \*temp;

temp=start;

cout<<(" THE LINKLIST IS AS FOLLOWS :");

while(temp->next!=NULL)

{

cout<<("%d->",temp->data);

temp=temp->next;

}

cout<<("%d->\n",temp->data);

}

void delend()

{

node \*q,\*p,\*k;

q=start;

if(start->data==0) //if we write here if(start==NULL)then it will not print

{ //the line.As here the rest portion of delend func delete the

cout<<("THERE IS NO ELEMENT IN THE LIST\n");//last value and remains it zero

}

else if(start->next==NULL)

{

k=start;

start=NULL;

free(k);

}

else

{

while(q->next->next!=NULL)

{

q=q->next;

}

p=q->next->next;

q->next=NULL;

free(p);

}

}

void insertbeg()

{

int item;

node \*p,\*q;

cout<<("enter the value which do you want to insert at begining\n");

cin>>item;

p=start;

q=(node \*)malloc(sizeof(node));

q->data=item;

q->next=p;

start=q;

}

void delbeg()

{

if(start==NULL)

{

cout<<("THERE IS NO ELEMENT IN THE LIST\n");

}

node \*p;

p=start;

start=p->next;

free(p);

}

void insertmid()

{

int item1,item2;

node \*p,\*q,\*k;

cout<<("enter the previous value after which you want to insert a new element\n");

cin>>item1;

cout<<("enter the value of new node\n");

cin>>item2;

q=(node \*)malloc(sizeof(node));

q->data=item2;

q->next=NULL;

p=start;

while(p->data!=item1)

{

p=p->next;

}

k=p->next;

p->next=q;

q->next=k;

}

void delmid()

{

int item;

node \*p,\*q,\*k;

cout<<("enter the previous value of that value which you want to delete\n");

cin>>item;

p=start;

while(p->data!=item)

{

p=p->next;

}

q=p->next->next;

k=p->next;

p->next=q;

free(k);

}

void modify()

{

int item1,item2;

node \*p,\*q;

cout<<("enter the value you want to modify\n");

cin>>item1;

cout<<("enter the new value\n");

cin>>item2;

p=start;

while(p->data!=item1)

{

p=p->next;

}

p->data=item2;

}