#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

/\*structure containing a data part and link part\*/

struct node

{

int data;

struct node\*link;

};

int insert(struct node\*\*,int);

int begin(struct node\*\*,int);

int end(struct node\*,int,int);

void display(struct node\*);

int count(struct node\*);

int delet(struct node\*,int);

main()

{

struct node\*p;

p=NULL;/\*empty linked list\*/

printf("\nNo of elements in the linked list = %d",count(p));

insert(&p,14);

display(p);

begin(&p,999);

display(p);

end(p,7,0);

display(p);

printf("\nNo of elements in the linked list = %d",count(p));

}

/\*add a node at the end point of linked list\*/

int insert(struct node\*\*q,int num)

{

struct node\*temp,\*r;

if(\*q ==NULL)/\*if the list is empty,create frist node\*/

{

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

temp->data = num;

temp->link = NULL;

\*q = temp;

}

else

{

temp = \*q;

/\*go to last node\*/

while(temp->link!=NULL)

{

temp =temp->link;

}

/\* add node at te end\*/

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

r->data = num;

r->link=NULL;

temp->link=r;

}

}

/\*adds a new node at the beginning of the linked list\*/

int begin(struct node\*\*q,int num)

{

struct node\*temp;

/\*add new node\*/

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

temp->data = num;

temp->link = \*q;

\*q = temp;

}

/\* adds a new node after the specified number of nodes\*/

int end(struct node\*q,int loc,int num)

{

struct node\*temp,\*r;

int i;

temp =q;

/\* skip to desired portion\*/

for(i=0;i<loc;i++)

{

temp = temp->link;

/\* if end of linked list is encoutered\*/

if(temp ==NULL)

{

printf("\nThere are less than %d elements in list",loc);

return 0;

}

}

/\* insert new node\*/

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

r->data = num;

r->link = temp->link;

temp->link = r;

}

/\* displays the contents of the linked list\*/

void display(struct node\*q)

{

printf("\n");

/\* traverse the entire linkrd list\*/

while(q !=NULL)

{

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

q =q->link;

}

}

/\* count the number of nodes present in the linked list\*/

int count (struct node\*q)

{

int c=0;

/\* traverse the entire lunked list\*/

while(q!=NULL)

{

q = q->link;

c++;

}

return c;

}

/\* deletes the specified node from the linked list\*/

void delet(struct node\*\*q,int num)

{

struct node\*old, \*temp;

temp =\*q;

while(temp!=NULL)

{

if(temp->data == num)

{

/\* if node to be daleted is the first node in the linked list\*/

if(temp ==\*q)

{

\*q = temp->link;

}

/\*deletes the intermediate nodes in the linked list\*/

else

old->link = temp->link;

/\*free the memort occupied by the node\*/

free(temp);

return;

}

/\* traverse the linked list till the last node is reached\*/

else

{

old = temp; /\*old points to the previous node\*/

temp = temp->link; /\* go to the next node\*/

}

}

printf("\nElements %d not found",num);

}