#include "linkedlist.h"

#include <stdio.h>

#include <stdlib.h>

static int l1, l2;

linkedlist::linkedlist()

{

}

linkedlist::~linkedlist()

{

}

struct node

{

int data;

struct node\* next;

};

void insert(struct node\*\* head, int newdata)

{

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

struct node\* last = \*head;

newnode->data = newdata;

newnode->next = NULL;

if (\*head == NULL)

{

\*head = newnode;

}

else

{

while (last->next != NULL)

{

last = last->next;

}

last->next = newnode;

}

}

void len(struct node\* head1, struct node\* head2)

{

while (head1 != NULL)

{

l1++;

head1 = head1->next;

}

while (head2 != NULL)

{

l2++;

head2 = head2->next;

}

}

struct node\* traverse(struct node\* head, int t)

{

while (t--)

{

head = head->next;

}

return head;

}

void merge(struct node\* head1, struct node\* head2)

{

struct node\* temp = head1;

if (temp->next == NULL)

{

temp->next = head2->next;

printf("address of pointer at the merging point:%p", temp);

printf("\nmerging point:%d\n", head2->data);

return;

}

else

{

temp = temp->next;

merge(temp, head2);

}

}

void display(struct node\* head)

{

struct node\* temp = head;

if (temp == NULL)

return;

else

{

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

temp = temp->next;

}

display(temp);

}

int main(void)

{

struct node\* head1 = NULL;

struct node\* head2 = NULL;

struct node\* temp = NULL;

int dif = 0;

insert(&head1, 1);

insert(&head1, 2);

insert(&head1, 3);

insert(&head1, 4);

insert(&head1, 5);

insert(&head2, 5);

insert(&head2, 6);

insert(&head2, 7);

printf("List1:");

display(head1);

printf("\nList2:");

display(head2);

printf("\n");

len(head1, head2);

dif = abs(l1 - l2);

temp = traverse(head1, dif);

merge(temp, head2);

printf("merged lists:");

display(head1);

printf("\n");

getchar();

return 0;

}