#include <stdio.h>

#include <stdlib.h>

struct node

{

int data;

struct node \*next;

};

struct node \*head = NULL;

void

insert\_tail (int x)

{

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

temp->data = x;

temp->next = NULL;

struct node \*ptr = head;

if (ptr == NULL)

{

head = temp;

}

else

{

while (ptr->next != NULL)

{

ptr = ptr->next;

}

ptr->next = temp;

}

}

void

display ()

{

struct node \*ptr = head;

while (ptr != NULL)

{

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

ptr = ptr->next;

}

}

int

len\_list (struct node \*ptr)

{

int len = 0;

while (ptr != NULL)

{

len++;

ptr = ptr->next;

}

return len;

}

void

k\_node\_from\_end\_naive (int k)

{

struct node \*ptr = head;

int n = len\_list (head);

for (int i = 1; i < ((n - k) + 1); i++)

{

ptr = ptr->next;

}

if (ptr != NULL)

{

printf ("\n %d is the kth node from end", ptr->data);

}

}

void

k\_node\_from\_end\_best (int k)

{

struct node \*p = head;

struct node \*q = head;

int i;

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

{

q = q->next;

}

while (q->next != NULL)

{

p = p->next;

q = q->next;

}

printf ("\n %d is the kth node from end", p->data);

}

int

main ()

{

insert\_tail (99);

insert\_tail (43);

insert\_tail (56);

insert\_tail (78);

insert\_tail (67);

display ();

k\_node\_from\_end\_naive (5);

k\_node\_from\_end\_best (4);

return 0;

}