#include <stdio.h>

#include <stdlib.h>

struct node {

int data;

struct node \*next;

};

struct node \*head = NULL;

void insertNode(int data) {

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

if (newNode == NULL) {

printf("Memory allocation failed.\n");

exit(1);

}

newNode->data = data;

newNode->next = NULL;

if (head == NULL) {

head = newNode;

} else {

struct node \*temp = head;

while (temp->next != NULL) {

temp = temp->next;

}

temp->next = newNode;

}

}

void displayList() {

struct node \*temp = head;

while (temp != NULL) {

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

temp = temp->next;

}

printf("\n");

}

**void** removeDuplicate() {

  //Node current will point to head

**struct** node \*current = head, \*index = NULL, \*temp = NULL;

**if**(head == NULL) {

**return**;

    }

**else** {

**while**(current != NULL){

            //Node temp will point to previous node to index.

            temp = current;

            //Index will point to node next to current

             index = current->next;

**while**(index != NULL) {

                //If current node's data is equal to index node's data

**if**(current->data == index->data) {

                    //Here, index node is pointing to the node which is duplicate of current node

                    //Skips the duplicate node by pointing to next node

                    temp->next = index->next;

                }

**else** {

                    //Temp will point to previous node of index.

                    temp = index;

                }

                index = index->next;

            }

            current = current->next;

        }

    }

}

int main() {

int n, data;

printf("Enter the number of elements: ");

scanf("%d", &n);

for (int i = 0; i < n; i++) {

printf("Enter element %d: ", i + 1);

scanf("%d", &data);

insertNode(data);

}

printf("Original list: ");

displayList();

removeDuplicate();

printf("Final list: ");

displayList();

return 0;

}