#include <stdio.h>

#include <stdlib.h>

struct Node {

    int coefficient;

    int exponent;

    struct Node\* next;

} \*newNode, \*current;

struct Node\* createAndInsertNode(struct Node\*\* poly, int coefficient, int exponent)

{

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

    newNode->coefficient = coefficient;

    newNode->exponent = exponent;

    newNode->next = NULL;

    if (\*poly == NULL)

    {

        \*poly = newNode;

    }

    else

    {

        current = \*poly;

        while (current->next != NULL)

        {

            current = current->next;

        }

        current->next = newNode;

    }

    return newNode;

}

void displayPolynomial(struct Node\* poly) {

    if (poly == NULL) {

        printf("Polynomial is empty.\n");

        return;

    }

    current = poly;

    while (current != NULL)

    {

        printf("%dx^%d ", current->coefficient, current->exponent);

        if (current->next != NULL)

        {

            printf("+ ");

        }

        current = current->next;

    }

    printf("\n");

}

struct Node\* addPolynomials(struct Node\* poly1, struct Node\* poly2) {

    if (poly1 == NULL) {

        return poly2;

    }

    if (poly2 == NULL) {

        return poly1;

    }

    struct Node\* result = NULL;

    struct Node\* current = NULL;

    while (poly1 != NULL && poly2 != NULL) {

        if (poly1->exponent > poly2->exponent) {

            current = createAndInsertNode(&result, poly1->coefficient, poly1->exponent);

            poly1 = poly1->next;

        } else if (poly1->exponent < poly2->exponent) {

            current = createAndInsertNode(&result, poly2->coefficient, poly2->exponent);

            poly2 = poly2->next;

        } else

        {

            int sum = poly1->coefficient + poly2->coefficient;

            if (sum != 0) {

                current = createAndInsertNode(&result, sum, poly1->exponent);

            }

            poly1 = poly1->next;

            poly2 = poly2->next;

        }

    }

    while (poly1 != NULL) {

        current = createAndInsertNode(&result, poly1->coefficient, poly1->exponent);

        poly1 = poly1->next;

    }

    while (poly2 != NULL) {

        current = createAndInsertNode(&result, poly2->coefficient, poly2->exponent);

        poly2 = poly2->next;

    }

    return result;

}

int main() {

    struct Node\* poly1 = NULL;

    struct Node\* poly2 = NULL;

    struct Node\* sumPoly = NULL;

    int coefficient, exponent;

    int n;

    printf("Enter the number of terms in Polynomial 1: ");

    scanf("%d", &n);

    printf("Enter the terms of Polynomial 1:\n");

    for (int i = 0; i < n; i++) {

        printf("Enter coefficient and exponent of term %d: ", i + 1);

        scanf("%d %d", &coefficient, &exponent);

        createAndInsertNode(&poly1, coefficient, exponent);

    }

    printf("Enter the number of terms in Polynomial 2: ");

    scanf("%d", &n);

    printf("Enter the terms of Polynomial 2:\n");

    for (int i = 0; i < n; i++) {

        printf("Enter coefficient and exponent of term %d: ", i + 1);

        scanf("%d %d", &coefficient, &exponent);

        createAndInsertNode(&poly2, coefficient, exponent);

    }

    printf("Polynomial 1: ");

    displayPolynomial(poly1);

    printf("Polynomial 2: ");

    displayPolynomial(poly2);

    sumPoly = addPolynomials(poly1, poly2);

    printf("Sum of Polynomials: ");

    displayPolynomial(sumPoly);

    return 0;

}