

Rajalakshmi Engineering College

Name: Lakshmi Narayanan S
Email: 241801133@rajalakshmi.edu.in
Roll no: 241801133
Phone: 9345832054
Branch: REC
Department: I AI & DS - AE
Batch: 2028
Degree: B.E - AI & DS

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_week 1_CY

Attempt : 1
Total Mark : 30
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Hayley loves studying polynomials, and she wants to write a program to compare two polynomials represented as linked lists and display whether they are equal or not.

The polynomials are expressed as a series of terms, where each term consists of a coefficient and an exponent. The program should read the polynomials from the user, compare them, and then display whether they are equal or not.

Input Format

The first line of input consists of an integer n , representing the number of terms in the first polynomial.

The following n lines of input consist of two integers, each representing the coefficient and the exponent of the term in the first polynomial.

The next line of input consists of an integer m , representing the number of terms in the second polynomial.

The following m lines of input consist of two integers, each representing the coefficient and the exponent of the term in the second polynomial.

Output Format

The first line of output prints "Polynomial 1: " followed by the first polynomial.

The second line prints "Polynomial 2: " followed by the second polynomial.

The polynomials should be displayed in the format ax^b , where a is the coefficient and b is the exponent.

If the two polynomials are equal, the third line prints "Polynomials are Equal."

If the two polynomials are not equal, the third line prints "Polynomials are Not Equal."

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 2

1 2

2 1

2

1 2

2 1

Output: Polynomial 1: $(1x^2) + (2x^1)$

Polynomial 2: $(1x^2) + (2x^1)$

Polynomials are Equal.

Answer

```
// You are using GCC
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```

typedef struct Term {
    int coeff;
    int expo;
    struct Term*next;
}Term;
Term*createTerm(int coeff,int expo){
    Term*newTerm = (Term*)malloc(sizeof(Term));
    newTerm->coeff = coeff;
    newTerm->expo = expo;
    newTerm->next = NULL;
    return newTerm;
}
void insertTerm(Term**poly,int coeff,int expo){
    Term*newTerm = createTerm(coeff,expo);
    if(*poly == NULL){
        *poly = newTerm;
    }
    else{
        Term*temp = *poly;
        while(temp->next != NULL){
            temp = temp->next;
        }
        temp->next = newTerm;
    }
}
void displayPolynomial(Term*poly){
    while(poly != NULL){
        printf("(%dx^%d)",poly->coeff,poly->expo);
        if(poly->next != NULL){
            printf(" + ");
        }
        poly = poly->next;
    }
    printf("\n");
}
int comparePolynomials(Term*poly1,Term*poly2){
    while(poly1 != NULL && poly2 != NULL){
        if(poly1->coeff != poly2->coeff || poly1->expo != poly2->expo){
            return 0;
        }
        poly1 = poly1->next;
        poly2 = poly2->next;
    }
}

```

```

    }
    return (poly1 == NULL && poly2 == NULL);
}

int main()
{
    Term* poly1 = NULL;
    Term* poly2 = NULL;
    int n,m,coeff,expo;
    scanf("%d",&n);
    for(int i = 0;i < n;i++)
    {
        scanf("%d %d",&coeff,&expo);
        insertTerm(&poly1,coeff,expo);
    }
    scanf("%d",&m);
    for(int i = 0;i < m;i++)
    {
        scanf("%d %d",&coeff,&expo);
        insertTerm(&poly2,coeff,expo);
    }
    printf("Polynomial 1:");
    displayPolynomial(poly1);
    printf("Polynomial 2:");
    displayPolynomial(poly2);
    if(comparePolynomials(poly1,poly2)){
        printf("Polynomials are Equal.\n");
    }
    else{
        printf("Polynomials are Not Equal.\n");
    }
    return 0;
}

```

Status : Correct

Marks : 10/10

2. Problem Statement

Hasini is studying polynomials in her class. Her teacher has introduced a new concept of two polynomials using linked lists.

The teacher provides Hasini with a program that takes two polynomials as input, represented as linked lists, and then displays them together. The polynomials are simplified and should be displayed in the format ax^b , where a is the coefficient and b is the exponent.

Input Format

The first line of input consists of an integer n , representing the number of terms in the first polynomial.

The following n lines of input consist of two integers each: the coefficient and the exponent of the term in the first polynomial.

The next line of input consists of an integer m , representing the number of terms in the second polynomial.

The following m lines of input consist of two integers each: the coefficient and the exponent of the term in the second polynomial.

Output Format

The first line of output prints the first polynomial.

The second line of output prints the second polynomial.

The polynomials should be displayed in the format ax^b , where a is the coefficient and b is the exponent.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3

1 2

2 1

3 0

3

2 2

1 1

4 0

Output: $1x^2 + 2x + 3$
 $2x^2 + 1x + 4$

Answer

-

Status : Skipped

Marks : 0/10

3. Problem Statement

Rani is studying polynomials in her class. She has learned about polynomial multiplication and is eager to try it out on her own. However, she finds the process of manually multiplying polynomials quite tedious. To make her task easier, she decides to write a program to multiply two polynomials represented as linked lists.

Help Rani by designing a program that takes two polynomials as input and outputs their product polynomial. Each polynomial is represented by a linked list of terms, where each term has a coefficient and an exponent. The terms are entered in descending order of exponents.

Input Format

The first line of input consists of an integer n , representing the number of terms in the first polynomial.

The following n lines of input consist of two integers each: the coefficient and the exponent of the term in the first polynomial.

The next line of input consists of an integer m , representing the number of terms in the second polynomial.

The following m lines of input consist of two integers each: the coefficient and the exponent of the term in the second polynomial.

Output Format

The first line of output prints the first polynomial.

The second line of output prints the second polynomial.

The third line of output prints the resulting polynomial after multiplying the given polynomials.

The polynomials should be displayed in the format, where each term is represented as ax^b , where a is the coefficient and b is the exponent.

Refer to the sample output for the exact format.

Sample Test Case

Input: 2

2 3

3 2

2

3 2

2 1

Output: $2x^3 + 3x^2$

$3x^2 + 2x$

$6x^5 + 13x^4 + 6x^3$

Answer

-

Status : Skipped

Marks : 0/10