

Rajalakshmi Engineering College

Name: Haripreeth CJ
Email: 241501065@rajalakshmi.edu.in
Roll no: 241501065
Phone: 9445359004
Branch: REC
Department: I AI & ML FA
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results



NeoColab_REC_CS23231_DATA STRUCTURES

REC_DS using C_Week 1_COD_Question 1

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Janani is a tech enthusiast who loves working with polynomials. She wants to create a program that can add polynomial coefficients and provide the sum of their coefficients.

The polynomials will be represented as a linked list, where each node of the linked list contains a coefficient and an exponent. The polynomial is represented in the standard form with 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 output prints the sum of the coefficients of the polynomials.

Sample Test Case

Input: 3

2 2

3 1

4 0

3

2 2

3 1

4 0

Output: 18

Answer

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

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

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

```
typedef struct Node
```

```
{
```

```
    int coeff,exp;
```

```
    struct Node*next;
```

```
}Node;
```

```
Node* createNode(int coeff,int exp){
```

```
    Node* newNode = (Node*)malloc(sizeof(Node));
```

```
    newNode->coeff=coeff;
```

```
    newNode->exp=exp;
```

```
    return newNode;
```

```
}
```

```
void insertTerm(Node** poly,int coeff,int exp){
```

```
    Node* newNode=createNode(coeff,exp);
```

```
    if(!*poly){
```

```
        *poly=newNode;
```

```
    }
```

```
    else {
```

```

    Node* temp=*poly;
    while(temp->next) temp=temp->next;
    temp->next=newNode;
}
}
int sumOfCoefficients(Node*poly)
{
    int sum=0;
    while(poly)
    {
        sum+=poly->coeff;
        poly=poly->next;
    }
    return sum;
}
Node* addPolynomials(Node* poly1, Node*poly2){
    Node* result= NULL;
    while(poly1 || poly2){
        if(poly1 &&(!poly2 || poly1->exp >poly2->exp)){
            insertTerm(&result,poly1->coeff,poly1->exp);
            poly1=poly1->next;
        }
        else if(poly2 &&(!poly1 || poly1->exp >poly2->exp)){
            insertTerm(&result,poly2->coeff,poly2->exp);
            poly2=poly2->next;
        }
        else
        {
            insertTerm(&result,poly1->coeff+poly2->coeff,poly1->exp);
            poly1=poly1->next;
            poly2=poly2->next;
        }
    }
    return result;
}
int main()
{
    int n,m;
    Node* poly1=NULL, *poly2=NULL;
    scanf("%d",&n);
    for(int i=0;i<n;i++){

```

```
    int coeff,exp;
    scanf("%d %d",&coeff,&exp);
    insertTerm(&poly1,coeff,exp);
}
scanf("%d",&m);
for(int i=0;i<m;i++){
    int coeff,exp;
    scanf("%d %d",&coeff,&exp);
    insertTerm(&poly2,coeff,exp);
}
Node*result= addPolynomials(poly1,poly2);
printf("%d\n",sumOfCoefficients(result));
return 0;
}
```

Status : Correct

Marks : 10/10