Rajalakshmi Engineering College

Name: Sam Devaraja J
Email: 240701463@rajalakshmi.edu.in
Roll no: 2116240701463
Phone: 7395954829
Branch: REC
Department: I CSE FE
Batch: 2028
Degree: B.E - CSE

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>
struct node
{
    int co;
    int exp;
    struct node *next;
};
typedef struct node Node;
Node *create(int co,int exp)
{
    Node* newNode=(Node*)malloc(sizeof(Node));
    newNode->co=co;
    newNode->exp=exp;
    newNode->next=NULL;
    return newNode;
}
Node* insert(Node* head, int co,int exp)
{
```

```
    Node* newNode=create(co,exp);
    if(!head)
    return newNode;
    Node* temp=head;
    while(temp->next)
    temp=temp->next;
    temp->next=newNode;
    return head;
}
Node* add(Node* p1,Node* p2)
{
    Node* result=NULL;
    while(p1 && p2)
    {
        if(p1->exp > p2->exp)
        {
            result=insert(result,p1->co,p1->exp);
            p1=p1->next;
        }
        else if(p1->exp < p2->exp)
        {
            result=insert(result,p2->co,p2->exp);
            p2=p2->next;
        }
        else
        {
            int sum=p1->co+p2->co;
            if(sum!=0)
                result=insert(result,sum,p1->exp);
            p1=p1->next;
            p2=p2->next;
        }
    }
    while(p1)
    {
        result=insert(result,p1->co,p1->exp);
        p1=p1->next;
    }
    while(p2)
    {
        result=insert(result,p2->co,p2->exp);
        p2=p2->next;
```

```
    }
    return result;
}
int sum_co(Node* head)
{
    int sum=0;
    while(head)
    {
        sum+=head->co;
        head=head->next;
    }
    return sum;
}
int main()
{
    int n,m,co,exp;
    Node *p1=NULL,*p2=NULL;
    scanf("%d",&n);
    for (int i=0;i<n;i++)
    {
        scanf("%d %d",&co,&exp);
        p1=insert(p1,co,exp);
    }
    scanf("%d",&m);
    for (int i=0;i<m;i++)
    {
        scanf("%d %d",&co,&exp);
        p2=insert(p2,co,exp);
    }
    Node* result=add(p1,p2);
    printf("%d\n",sum_co(result));
    return 0;
}
```

Status : Correct ***Marks : 10/10***