

# Rajalakshmi Engineering College

Name: Dhanusri ramakrishnan suresh babu

Email: 241801051@rajalakshmi.edu.in

Roll no: 241801051

Phone: 9003627964

Branch: REC

Department: I AI & DS FB

Batch: 2028

Degree: B.E - AI & DS

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
// struct node(){
//   int coeff;
//   int expo;
//   struct node *next;
// }
// typedef struct node Node;
```

```
// Node *create(){
//   int c,e;
//   Node *head=NULL;
//   Node *tail=NULL;
//   scanf("%d",&n);
//   do{
//     scanf("%d",&c);
//     scanf("%d",&e);
//     Node*newnode=(Node*)malloc(sizeof(Node));
//     newnode->coeff=c;
//     newnode->expo=e;
//     newnode->next=NULL;
```

```

// if(head==NULL){
//     head=tail=newnode;
// }
// else{
//     tail->next=newnode;
//     tail=newnode;
// }
//     n=n-1;
// }while(n>0);
// return head;
// }

```

```

// Node *add(poly1,poly2){

```

```

// }

```

```

// int main(){
//     Node*poly1,*poly2;
//     poly1=create();
//     poly2=create();
//     Node*add(poly1,poly2);
//     printpoly(addpoly)
// }

```

```

#include<stdio.h>

```

```

#include<stdlib.h>

```

```

typedef struct node{
    int data1,data2;
    int exp;
    struct node*link;
}node;

```

```

int main(){
    int n,m;
    int sum=0;
    struct node no;
    scanf("%d",&n);
    for(int i=0;i<n;i++){
        scanf("%d %d",&no.data1,&no.data2);
        sum+=no.data1;
    }
    scanf("%d",&m);
    for(int i=0;i<m;i++){
        scanf("%d %d",&no.data1,&no.data2);
    }

```

```
        sum+=no.data1;  
    }  
    printf("%d",sum);  
  
}
```

**Status :** Correct

**Marks : 10/10**