

|  |   |
|--|---|
| <b>Subject:</b> - DATA STRUCTURE         | <b>Subject Code:</b> 313301                                     |
| <b>Semester:</b> - III                   | <b>Course:</b> COMPUTER ENGINEERING                             |
| Laboratory No: L003                      | <b>Name of Subject Teacher:</b> Prof. Imraan S.                 |
| <b>Name of Student:</b> Saad sharif kazi | <b>Roll Id:</b> - 24203A0013                                    |
| <b>Experiment No:</b>                    | 15  |
| <b>Title of Experiment</b>               | Write a 'C' Program to add Two Polynomials using a Linked List. |

**Aim:** Write a 'C' Program to add Two Polynomials using a Linked List.

### Algorithm:

Step 1: Start

Step 2: Define a structure Node with members: coeff (int), power (int), and next (pointer to Node)

Step 3: Declare pointer poly1 to struct Node and initialize it to NULL

Step 4: Print "Enter the number of terms in the Polynomial"

Step 5: Read integer n from user

Step 6: Call function createlinkedlist(n) and assign result to poly1

Step 7: Inside createlinkedlist(n):

Step 7.1: If  $n \leq 0$ , print error and return NULL

Step 7.2: For  $i = 1$  to  $n$ , do:

Step 7.2.1: Read coeff and power from user

Step 7.2.2: Call createnode(coeff, power) to create a new node

Step 7.2.3: If  $i == 1$ , assign head and temp to new node

Step 7.2.4: Else, set  $\text{temp} \rightarrow \text{next} = \text{newnode}$  and move  $\text{temp} = \text{temp} \rightarrow \text{next}$

Step 7.3: Return head

Step 8: In main, call printlist(poly1)

Step 9: Inside printlist(head):

Step 9.1: While head is not NULL

Step 9.1.1: Print  $\text{head} \rightarrow \text{coeff}$  and  $\text{head} \rightarrow \text{power}$  in the form  $aX^b$

Step 9.1.2: If  $\text{head} \rightarrow \text{next}$  is not NULL, print " + "

Step 9.1.3: Move  $\text{head} = \text{head} \rightarrow \text{next}$

Step 10: End

### CODE:

```
≡ File Edit Search Run Compile Debug Project Options Window Help
[ ] SAAD15.C 1=[ ]
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct Node
{
int coeff;
int power;
struct Node* next;
};

struct Node* createlinkedlist(int);
struct Node* createnode(int,int);
void printList(struct Node*);
void add(struct Node*,struct Node*);

void main()
{
int n1,n2;
struct Node *poly1 = NULL, *poly2 = NULL;
clrscr();
printf("Enter the No. of Terms in 1st Polynomial: ");
1:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
≡ File Edit Search Run Compile Debug Project Options Window Help
[ ] SAAD15.C 1=[ ]
printf("Enter the No. of Terms in 1st Polynomial: ");
scanf("%d",&n1);
printf("Enter the No. of Terms in 2nd Polynomial: ");
scanf("%d",&n2);
printf("\nPolynomial 1: \n");
poly1 = createlinkedlist(n1);
printf("\nPolynomial 2: \n");
poly2 = createlinkedlist(n2);
printf("\n1st Polynomial: \n");
printList(poly1);
printf("\n2nd Polynomial: \n");
printList(poly2);
printf("\nResult after Addition: \n");
add(poly1,poly2);
getch();
}

struct Node* createlinkedlist(int n)
{
int coeff,power,i;
struct Node *head = NULL, *temp = NULL, *newNode = NULL;
41:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

```
≡ File Edit Search Run Compile Debug Project Options Window Help
[■] SAAD15.C 1-[+]  
struct Node *head = NULL, *temp = NULL, *newNode = NULL;  
if(n<=0)  
{  
printf("Number of Nodes should be greater then 0...");  
return NULL;  
}  
printf("Enter Coefficient & Power for Term 1: ");  
scanf("%i %i",&coeff,&power);  
newNode = createnode(coeff,power);  
head = newNode;  
temp = newNode;  
for(i=2;i<=n;i++)  
{  
printf("Enter Coefficient & Power for Term %i: ",i);  
scanf("%i %i",&coeff,&power);  
newNode = createnode(coeff,power);  
temp->next = newNode;  
temp = temp->next;  
}  
return head;  
}  
61:1  
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu  
≡ File Edit Search Run Compile Debug Project Options Window Help  
[■] SAAD15.C 1-[+]  
struct Node* createnode(int coeff, int power)  
{  
struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));  
newNode->coeff = coeff;  
newNode->power = power;  
newNode->next = NULL;  
return newNode;  
}  
  
void printList(struct Node* head)  
{  
struct Node* temp = head;  
while(temp->next != NULL)  
{  
printf("%iX%i + ",temp->coeff,temp->power);  
temp = temp->next;  
}  
printf("%iX%i\n",temp->coeff,temp->power);  
}  
82:1  
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

```
File Edit Search Run Compile Debug Project Options Window Help
SAAD15.C 1=[+]
void add(struct Node* poly1,struct Node* poly2)
{
    struct Node* result = NULL; // for new linked list
    struct Node *temp1 = poly1, *temp2 = poly2, *newNode = NULL, *t;
    int coeff,power;
    newNode = createnode(0,0);
    result = newNode;
    t = newNode;

    while((temp1!=NULL) || (temp2!=NULL))
    {
        if(temp1!=NULL && (temp2==NULL || temp1->power > temp2->power))
        {
            coeff = temp1->coeff;
            power = temp1->power;
            newNode = createnode(coeff,power);
            t->next = newNode;
            t = t->next;
            temp1 = temp1->next;
        }
        else if(temp2!=NULL && (temp1==NULL || temp1->power < temp2->power))
        {
            coeff = temp2->coeff;
            power = temp2->power;
            newNode = createnode(coeff,power);
            t->next = newNode;
            t = t->next;
            temp2 = temp2->next;
        }
        else if(temp1!=NULL && temp2!=NULL && temp1->power == temp2->power)
        {
            coeff = temp1->coeff + temp2->coeff;
            power = temp1->power;
            newNode = createnode(coeff,power);
            t->next = newNode;
            t = t->next;
            temp1 = temp1->next;
            temp2 = temp2->next;
        }
    }
    printList(result->next);
}
103:8
```

F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu  
File Edit Search Run Compile Debug Project Options Window Help

OUTPUT: -

Enter the No. of Terms in 1st Polynomial: 2  
Enter the No. of Terms in 2nd Polynomial: 3

Polynomial 1:

Enter Coefficient & Power for Term 1: 4 3  
Enter Coefficient & Power for Term 2: 3 2

Polynomial 2:

Enter Coefficient & Power for Term 1: 6 1  
Enter Coefficient & Power for Term 2: 2 3  
Enter Coefficient & Power for Term 3: 4 7

1st Polynomial:

$4X^3 + 3X^2$

2nd Polynomial:

$6X^1 + 2X^3 + 4X^7$

Result after Addition:

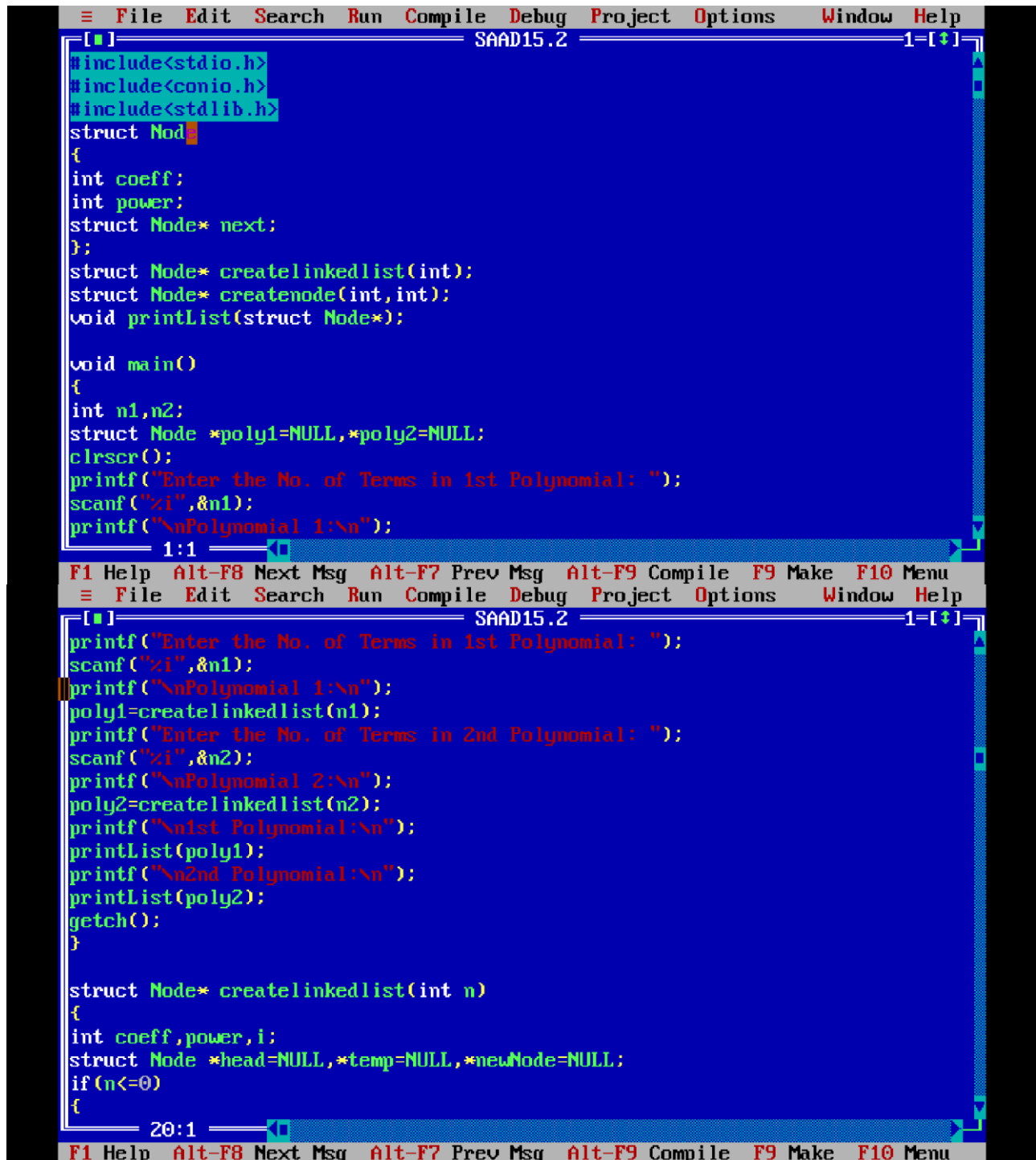
$4X^3 + 3X^2 + 6X^1 + 2X^3 + 4X^7$

-

### Practical Related Questions:

1. Write a C program to Create two polynomial  $P(x)=3x^4 + 2x^3 - 4x^2 + 7$  and  $Q(x) = 5x^3 + 4x^2 - 5$

CODE:



```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct Node
{
int coeff;
int power;
struct Node* next;
};
struct Node* createlinkedlist(int);
struct Node* createnode(int,int);
void printList(struct Node*);

void main()
{
int n1,n2;
struct Node *poly1=NULL,*poly2=NULL;
clrscr();
printf("Enter the No. of Terms in 1st Polynomial: ");
scanf("%i",&n1);
printf("\nPolynomial 1:\n");

printf("Enter the No. of Terms in 1st Polynomial: ");
scanf("%i",&n1);
printf("\nPolynomial 1:\n");
poly1=createlinkedlist(n1);
printf("Enter the No. of Terms in 2nd Polynomial: ");
scanf("%i",&n2);
printf("\nPolynomial 2:\n");
poly2=createlinkedlist(n2);
printf("\n1st Polynomial:\n");
printList(poly1);
printf("\n2nd Polynomial:\n");
printList(poly2);
getch();
}

struct Node* createlinkedlist(int n)
{
int coeff,power,i;
struct Node *head=NULL,*temp=NULL,*newNode=NULL;
if(n<=0)
{

```

```
File Edit Search Run Compile Debug Project Options Window Help
SAAD15.2 1-[+]
if(n<=0)
{
printf("Number of Nodes should be greater than 0...");
return NULL;
}
printf("Enter Coefficient & Power for Term 1: ");
scanf("%i %i",&coeff,&power);
newNode=createnode(coeff,power);
head=newNode;
temp=newNode;
for(i=2;i<=n;i++)
{
printf("Enter Coefficient & Power for Term %i: ",i);
scanf("%i %i",&coeff,&power);
newNode=createnode(coeff,power);
temp->next=newNode;
temp=temp->next;
}
return head;
}

58:2

F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
File Edit Search Run Compile Debug Project Options Window Help
SAAD15.2 1-[+]
struct Node* createnode(int coeff,int power)
{
struct Node* newNode=(struct Node*)malloc(sizeof(struct Node));
newNode->coeff=coeff;
newNode->power=power;
newNode->next=NULL;
return newNode;
}

void printList(struct Node* head)
{
struct Node* temp=head;
while(temp->next!=NULL)
{
printf("%iX^%i + ",temp->coeff,temp->power);
temp=temp->next;
}
printf("%iX^%i",temp->coeff,temp->power);
}

78:2

F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

## OUTPUT:

```
Enter the No. of Terms in 1st Polynomial: 2

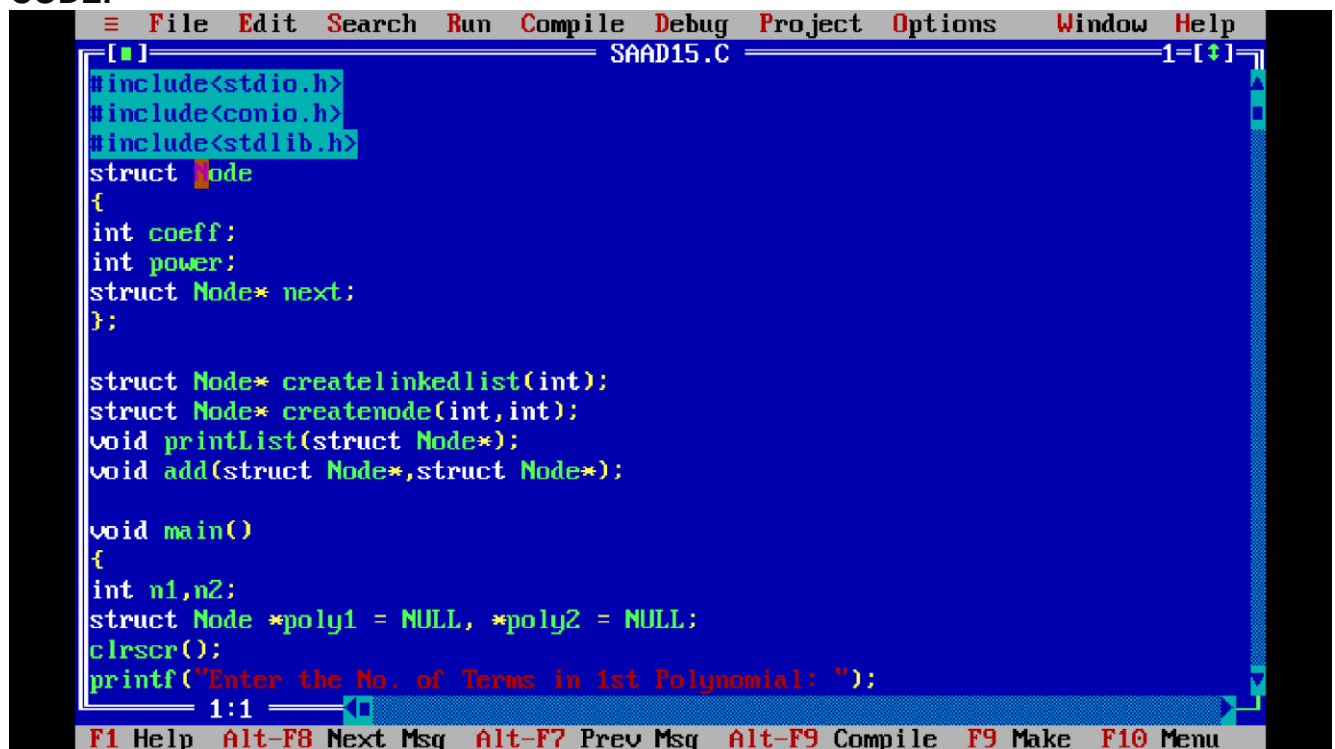
Polynomial 1:
Enter Coefficient & Power for Term 1: 5 6
Enter Coefficient & Power for Term 2: 6 7
Enter the No. of Terms in 2nd Polynomial: 3

Polynomial 2:
Enter Coefficient & Power for Term 1: 3 4
Enter Coefficient & Power for Term 2: 5 1
Enter Coefficient & Power for Term 3: 7 3

1st Polynomial:
5X^6 + 6X^7
2nd Polynomial:
3X^4 + 5X^1 + 7X^3_
```

2. Write a C program to display addition of two created polynomial.

## CODE:



```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct Node
{
    int coeff;
    int power;
    struct Node* next;
};

struct Node* createlinkedlist(int);
struct Node* createnode(int,int);
void printList(struct Node*);
void add(struct Node*,struct Node*);

void main()
{
    int n1,n2;
    struct Node *poly1 = NULL, *poly2 = NULL;
    clrscr();
    printf("Enter the No. of Terms in 1st Polynomial: ");
```



```
≡ File Edit Search Run Compile Debug Project Options Window Help
[■] SAAD15.C 1=[+]  
printf("Enter the No. of Terms in 1st Polynomial: ");  
scanf("%d",&n1);  
printf("Enter the No. of Terms in 2nd Polynomial: ");  
scanf("%d",&n2);  
printf("\nPolynomial 1: \n");  
poly1 = createlinkedlist(n1);  
printf("\nPolynomial 2: \n");  
poly2 = createlinkedlist(n2);  
printf("\n1st Polynomial: \n");  
printList(poly1);  
printf("\n2nd Polynomial: \n");  
printList(poly2);  
printf("\nResult after Addition: \n");  
add(poly1,poly2);  
getch();  
}  
  
struct Node* createlinkedlist(int n)  
{  
    int coeff,power,i;  
    struct Node *head = NULL, *temp = NULL, *newNode = NULL;  
    41:1
```

```
≡ File Edit Search Run Compile Debug Project Options Window Help
[■] SAAD15.C 1=[+]  
struct Node *head = NULL, *temp = NULL, *newNode = NULL;  
if(n<=0)  
{  
    printf("Number of Nodes should be greater than 0...");  
    return NULL;  
}  
printf("Enter Coefficient & Power for Term 1: ");  
scanf("%d %d",&coeff,&power);  
newNode = createnode(coeff,power);  
head = newNode;  
temp = newNode;  
for(i=2;i<=n;i++)  
{  
    printf("Enter Coefficient & Power for Term %d: ",i);  
    scanf("%d %d",&coeff,&power);  
    newNode = createnode(coeff,power);  
    temp->next = newNode;  
    temp = temp->next;  
}  
return head;  
}  
61:1
```

```
File Edit Search Run Compile Debug Project Options Window Help
SAAD15.C 1=1
struct Node* createnode(int coeff, int power)
{
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->coeff = coeff;
    newNode->power = power;
    newNode->next = NULL;
    return newNode;
}

void printList(struct Node* head)
{
    struct Node* temp = head;
    while(temp->next != NULL)
    {
        printf("%iX%i + ",temp->coeff,temp->power);
        temp = temp->next;
    }
    printf("%iX%i\n",temp->coeff,temp->power);
}

82:1

F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
File Edit Search Run Compile Debug Project Options Window Help
SAAD15.C 1=1
void add(struct Node* poly1,struct Node* poly2)
{
    struct Node* result = NULL; // for new linked list
    struct Node *temp1 = poly1, *temp2 = poly2, *newNode = NULL, *t;
    int coeff,power;
    newNode = createnode(0,0);
    result = newNode;
    t = newNode;

    while((temp1!=NULL) || (temp2!=NULL))
    {
        if(temp1!=NULL && (temp2==NULL || temp1->power > temp2->power))
        {
            coeff = temp1->coeff;
            power = temp1->power;
            newNode = createnode(coeff,power);
            t->next = newNode;
            t = t->next;
            temp1 = temp1->next;
        }
        else if(temp2!=NULL && (temp1==NULL || temp1->power < temp2->power))
        {
            coeff = temp2->coeff;
            power = temp2->power;
            newNode = createnode(coeff,power);
            t->next = newNode;
            t = t->next;
            temp2 = temp2->next;
        }
    }
    t->next = NULL;
    result = result->next;

    103:1

F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

```
File Edit Search Run Compile Debug Project Options Window Help
SAAD15.C
else if(temp2!=NULL && (temp1==NULL || temp1->power < temp2->power))
{
    coeff = temp2->coeff;
    power = temp2->power;
    newNode = createnode(coeff,power);
    t->next = newNode;
    t = t->next;
    temp2 = temp2->next;
}
else if(temp1!=NULL && temp2!=NULL && temp1->power == temp2->power)
{
    coeff = temp1->coeff + temp2->coeff;
    power = temp1->power;
    newNode = createnode(coeff,power);
    t->next = newNode;
    t = t->next;
    temp1 = temp1->next;
    temp2 = temp2->next;
}
}
printList(result->next);
103:8
```

## OUTPUT: -

```
Enter the No. of Terms in 1st Polynomial: 2
Enter the No. of Terms in 2nd Polynomial: 3

Polynomial 1:
Enter Coefficient & Power for Term 1: 4 3
Enter Coefficient & Power for Term 2: 3 2

Polynomial 2:
Enter Coefficient & Power for Term 1: 6 1
Enter Coefficient & Power for Term 2: 2 3
Enter Coefficient & Power for Term 3: 4 7

1st Polynomial:
4X^3 + 3X^2

2nd Polynomial:
6X^1 + 2X^3 + 4X^7

Result after Addition:
4X^3 + 3X^2 + 6X^1 + 2X^3 + 4X^7
-
```



| Marks Obtained             |                            |            | Dated signature of Teacher |
|----------------------------|----------------------------|------------|----------------------------|
| Process<br>Related<br>(35) | Product<br>Related<br>(15) | Total (50) |                            |
|                            |                            |            |                            |