NAME : JANHAVI GATTANI

BATCH : 2

ROLL NO : 37

PRN : 12311291

**ASSIGNMENT 7**

**QUESTION : Implement a Polynomial addition and multiplication using Linked Lists.**

**CODE:**

#include <stdio.h>

#include <stdlib.h>

typedef struct Node {

int coeff;

int exp;

struct Node\* next;

} Node;

// Create a new term node

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

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

newNode->coeff = coeff;

newNode->exp = exp;

newNode->next = NULL;

return newNode;

}

// Append node at the end and return updated list

Node\* append(Node\* poly, int coeff, int exp) {

if (coeff == 0) return poly;

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

if (poly == NULL)

return newNode;

Node\* temp = poly;

while (temp->next)

temp = temp->next;

temp->next = newNode;

return poly;

}

// Display the polynomial

void display(Node\* poly) {

while (poly) {

printf("%dx^%d", poly->coeff, poly->exp);

if (poly->next) printf(" + ");

poly = poly->next;

}

printf("\n");

}

// Add two polynomials

Node\* add(Node\* p1, Node\* p2) {

Node\* result = NULL;

while (p1 && p2) {

if (p1->exp == p2->exp) {

result = append(result, p1->coeff + p2->coeff, p1->exp);

p1 = p1->next;

p2 = p2->next;

} else if (p1->exp > p2->exp) {

result = append(result, p1->coeff, p1->exp);

p1 = p1->next;

} else {

result = append(result, p2->coeff, p2->exp);

p2 = p2->next;

}

}

while (p1) {

result = append(result, p1->coeff, p1->exp);

p1 = p1->next;

}

while (p2) {

result = append(result, p2->coeff, p2->exp);

p2 = p2->next;

}

return result;

}

// Multiply two polynomials

Node\* multiply(Node\* p1, Node\* p2) {

Node\* result = NULL;

for (Node\* t1 = p1; t1 != NULL; t1 = t1->next) {

for (Node\* t2 = p2; t2 != NULL; t2 = t2->next) {

int coeff = t1->coeff \* t2->coeff;

int exp = t1->exp + t2->exp;

Node \*temp = result, \*prev = NULL;

while (temp && temp->exp > exp) {

prev = temp;

temp = temp->next;

}

if (temp && temp->exp == exp) {

temp->coeff += coeff;

} else {

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

if (!prev) {

newNode->next = result;

result = newNode;

} else {

newNode->next = temp;

prev->next = newNode;

}

}

}

}

return result;

}

// Main function to demonstrate

int main() {

Node \*poly1 = NULL, \*poly2 = NULL;

// poly1 = 3x^3 + 2x^2 + 5

poly1 = append(poly1, 3, 3);

poly1 = append(poly1, 2, 2);

poly1 = append(poly1, 5, 0);

// poly2 = 1x^2 + 2x + 4

poly2 = append(poly2, 1, 2);

poly2 = append(poly2, 2, 1);

poly2 = append(poly2, 4, 0);

printf("Polynomial 1: ");

display(poly1);

printf("Polynomial 2: ");

display(poly2);

Node\* sum = add(poly1, poly2);

printf("Sum: ");

display(sum);

Node\* product = multiply(poly1, poly2);

printf("Product: ");

display(product);

return 0;

}

OUTPUT:

