

DS Lab Assignment 3

Pranav Joshi

CS-B

Roll no: 43

Title: WAP to Implement following polynomial operations using Linked Lists.

a. Addition b. Subtraction c. Multiplication

Code:

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

typedef struct Node {
    int coeff;
    int exp;
    struct Node* next;
} node;

node* insertNode(node* head, int coeff, int exp) {
    node* newNode = (node*)malloc(sizeof(node));
    newNode->coeff = coeff;
    newNode->exp = exp;
    newNode->next = NULL;
    if(head == NULL) {
        head = newNode;
    } else {
        node* curr = head;
        while(curr->next != NULL) {
            curr = curr->next;
        }
        curr->next = newNode;
    }
    return head;
}

void printPolynomial(node* head) {
    if(head == NULL) {
        printf("0\n");
    } else {
        while(head != NULL) {
            printf("%dx^%d ", head->coeff, head->exp);
            head = head->next;
        }
    }
}
```

```

        if(head != NULL) {
            printf("+ ");
        }
    }
    printf("\n");
}

node* addPolynomial(node* poly1, node* poly2) {
    node* result = NULL;
    node* curr1 = poly1;
    node* curr2 = poly2;
    while(curr1 != NULL && curr2 != NULL) {
        if(curr1->exp == curr2->exp) {
            int coef = curr1->coeff + curr2->coeff;
            result = insertNode(result, coef, curr1->exp);
            curr1 = curr1->next;
            curr2 = curr2->next;
        } else if(curr1->exp > curr2->exp) {
            result = insertNode(result, curr1->coeff, curr1->exp);
            curr1 = curr1->next;
        } else {
            result = insertNode(result, curr2->coeff, curr2->exp);
            curr2 = curr2->next;
        }
    }
    while(curr1 != NULL) {
        result = insertNode(result, curr1->coeff, curr1->exp);
        curr1 = curr1->next;
    }
    while(curr2 != NULL) {
        result = insertNode(result, curr2->coeff, curr2->exp);
        curr2 = curr2->next;
    }
    return result;
}

node* subtractPolynomial(node* poly1, node* poly2) {
    node* result = NULL;
    node* curr1 = poly1;
    node* curr2 = poly2;
    while(curr1 != NULL && curr2 != NULL) {
        if(curr1->exp == curr2->exp) {
            int coef = curr1->coeff - curr2->coeff;
            result = insertNode(result, coef, curr1->exp);
            curr1 = curr1->next;
            curr2 = curr2->next;
        } else if(curr1->exp > curr2->exp) {

```

```

        result = insertNode(result, curr1->coeff, curr1->exp);
        curr1 = curr1->next;
    } else {
        result = insertNode(result, -curr2->coeff, curr2->exp);
        curr2 = curr2->next;
    }
}
while(curr1 != NULL) {
    result = insertNode(result, curr1->coeff, curr1->exp);
    curr1 = curr1->next;
}
while(curr2 != NULL) {
    result = insertNode(result, -curr2->coeff, curr2->exp);
    curr2 = curr2->next;
}
return result;
}

node* multiplyPolynomial(node* poly1, node* poly2) {
    node* result = NULL;
    node* curr1 = poly1;
    while(curr1 != NULL) {
        node* curr2 = poly2;
        while(curr2 != NULL) {
            int coef = curr1->coeff * curr2->coeff;
            int exp = curr1->exp + curr2->exp;
            result = insertNode(result, coef, exp);
            curr2 = curr2->next;
        }
        curr1 = curr1->next;
    }
    return result;
}

int main() {
    node* poly1 = NULL;
    node* poly2 = NULL;

    printf("Enter the number of terms in the first polynomial: ");
    int n1;
    scanf("%d", &n1);
    for(int i=0; i<n1; i++) {
        printf("Enter coefficient and exponent of term %d: ", i+1);
        int coeff, exp;
        scanf("%d %d", &coeff, &exp);
        poly1 = insertNode(poly1, coeff, exp);
    }
}

```

```

printf("Enter the number of terms in the second polynomial: ");
int n2;
scanf("%d", &n2);
for(int i=0; i<n2; i++) {
    printf("Enter coefficient and exponent of term %d: ", i+1);
    int coeff, exp;
    scanf("%d %d", &coeff, &exp);
    poly2 = insertNode(poly2, coeff, exp);
}

printf("First polynomial: ");
printPolynomial(poly1);
printf("Second polynomial: ");
printPolynomial(poly2);

printf("Sum of the two polynomials: ");
node* sum = addPolynomial(poly1, poly2);
printPolynomial(sum);

printf("Difference of the two polynomials: ");
node* diff = subtractPolynomial(poly1, poly2);
printPolynomial(diff);

printf("Product of the two polynomials: ");
node* product = multiplyPolynomial(poly1, poly2);
printPolynomial(product);

return 0;
}

```

Output:

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS C:\Code\C\Code> cd "c:\Code\C\Code\" ; if ($?) { gcc Sem4Assignment3.c -o Sem4Assignment3 } ; if ($?) { .\Sem4Assignment3 }
Enter the number of terms in the first polynomial: 3
Enter coefficient and exponent of term 1: 5 2
Enter coefficient and exponent of term 2: 4 1
Enter coefficient and exponent of term 3: 3 0
Enter the number of terms in the second polynomial: 3
Enter coefficient and exponent of term 1: 3 2
Enter coefficient and exponent of term 2: 4 1
Enter coefficient and exponent of term 3: 5 0
First polynomial: 5x^2 + 4x^1 + 3x^0
Second polynomial: 3x^2 + 4x^1 + 5x^0
Sum of the two polynomials: 8x^2 + 8x^1 + 8x^0
Difference of the two polynomials: 2x^2 + 0x^1 + -2x^0
Product of the two polynomials: 15x^4 + 20x^3 + 25x^2 + 12x^3 + 16x^2 + 20x^1 + 9x^2 + 12x^1 + 15x^0
PS C:\Code\C\Code> 

```