S.No: 8	Exp. Name: ***Polynomial Operations - Adding Polynomials using Linked List***	Date:2024-06-02

ID: 92132321021

2023-2027-AIDS

PSNA College of Engineering and Technology

Aim:

Write a C program to **add** two polynomials using linked lists.

Note: Driver code is provided to you in the editor.

Source Code:

PolyLLMain1.c

```
#include <stdio.h>
#include <stdlib.h>
#include "AddPolyLL.c"

poly create(poly head) {
   poly temp;
   char ch;
   int coeff, exp;
   do {
      temp = (poly)malloc(sizeof(struct polynomial));
      printf("Coeff and Power of the term: ");
      scanf("%d%d", &coeff, &exp);
      temp -> coeff = coeff;
      temp -> exp = exp;
      temp -> next = NULL;
      head = addTerm(head, temp);
      printf("Want to add more terms?(y/n): ");
      scanf(" %c", &ch);
   } while(ch != 'n');
   return head;
}

void main() {
   poly head1=NULL, head2= NULL, result = NULL;
   int ch;
   printf("First polynomial: \n");
   head1 = create(head1);
   printf("Second polynomial: \n");
   head2 = create(head2);
   result = add(head1, head2);
   printf("First polynomial: ");
   print(head1);
   printf("Second polynomial: ");
   print(head2);
   printf("Addition: ");
   print(result);
}
```

AddPolyLL.c

```
struct polynomial {
    int coeff;
```

```
    int exp;
    struct polynomial* next;
};

typedef struct polynomial* poly;

// Function to add a term to the polynomial in a sorted manner
poly addTerm(poly head, poly term) {
    poly temp = head, prev = NULL;

    // Find the right place to insert the term
    while (temp != NULL && temp->exp > term->exp) {
        prev = temp;
        temp = temp->next;
    }

    // If a term with the same exponent is found, add the coefficients
    if (temp != NULL && temp->exp == term->exp) {
        temp->coeff += term->coeff;
        free(term);
    } else {
        // Insert the term in the correct position
        if (prev == NULL) {
            term->next = head;
            head = term;
        } else {
            prev->next = term;
            term->next = temp;
        }
    }
    return head;
}

// Function to add two polynomials
poly add(poly head1, poly head2) {
    poly result = NULL, temp;

    while (head1 && head2) {
        temp = (poly)malloc(sizeof(struct polynomial));

        if (head1->exp == head2->exp) {
            temp->coeff = head1->coeff + head2->coeff;
            temp->exp = head1->exp;
            head1 = head1->next;
            head2 = head2->next;
        } else if (head1->exp > head2->exp) {
            temp->coeff = head1->coeff;
            temp->exp = head1->exp;
            head1 = head1->next;
        } else {
            temp->coeff = head2->coeff;
            temp->exp = head2->exp;
            head2 = head2->next;
        }
        temp->next = NULL;
        result = addTerm(result, temp);
```

```
    }

    // Add remaining terms from head1
    while (head1) {
        temp = (poly)malloc(sizeof(struct polynomial));
        temp->coeff = head1->coeff;
        temp->exp = head1->exp;
        temp->next = NULL;
        result = addTerm(result, temp);
        head1 = head1->next;
    }

    // Add remaining terms from head2
    while (head2) {
        temp = (poly)malloc(sizeof(struct polynomial));
        temp->coeff = head2->coeff;
        temp->exp = head2->exp;
        temp->next = NULL;
        result = addTerm(result, temp);
        head2 = head2->next;
    }

    return result;
}

// Function to print the polynomial
void print(poly head) {
    while (head) {
        printf("%d X^%d", head->coeff, head->exp);
        head = head->next;
        if (head) {
            printf(" + ");
        }
    }
    printf("\n");
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
First polynomial: 2 3
Coeff and Power of the term: 2 3
Want to add more terms?(y/n): y
Coeff and Power of the term: 4 2
Want to add more terms?(y/n): y
Coeff and Power of the term: 6 1
Want to add more terms?(y/n): y
Coeff and Power of the term: 8 0
Want to add more terms?(y/n): n
Second polynomial: 1 3

ID: 92132321021
2023-2027-AIDS
PSNA College of Engineering and Technology

Coeff and Power of the term: 1 3
Want to add more terms?(y/n): y
Coeff and Power of the term: 3 2
Want to add more terms?(y/n): y
Coeff and Power of the term: 5 1
Want to add more terms?(y/n): y
Coeff and Power of the term: 7 0
Want to add more terms?(y/n): n
First polynomial: 2 X^3 + 4 X^2 + 6 X^1 + 8 X^0
Second polynomial: 1 X^3 + 3 X^2 + 5 X^1 + 7 X^0
Addition: 3 X^3 + 7 X^2 + 11 X^1 + 15 X^0

Test Case - 2
User Output
First polynomial: 1 3
Coeff and Power of the term: 1 3
Want to add more terms?(y/n): y
Coeff and Power of the term: 2 3
Want to add more terms?(y/n): n
Second polynomial: 3 4
Coeff and Power of the term: 3 4
Want to add more terms?(y/n): y
Coeff and Power of the term: 4 4
Want to add more terms?(y/n): n
First polynomial: 3 X^3
Second polynomial: 7 X^4
Addition: 7 X^4 + 3 X^3