Ex 3: Polynomial Addition

PROGRAM:

#include<stdio.h>

#include<stdlib.h>

struct node

{

int coeff;

int expo;

struct node \*next;

};

struct node\* insert(struct node \*head,int co,int exp)

{

struct node \*temp;

struct node \*newnode=malloc(sizeof(struct node));

newnode->coeff=co;

newnode->expo=exp;

newnode->next=NULL;

if(head==NULL || exp>head->expo)

{

newnode->next=head;

head=newnode;

}

else

{

temp=head;

while(temp->next!=NULL &&temp->next->expo>=exp)

temp=temp->next;

newnode->next=temp->next;

temp->next=newnode;

}

return head;

}

struct node\* create(struct node \*head)

{

int n,i;

int coeff;

int expo;

printf("Enter the no of terms:");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("Enter the coeefficient for term %d:",i+1);

scanf("%d",&coeff);

printf("Enter the exponent for term %d:",i+1);

scanf("%d",&expo);

head=insert(head,coeff,expo);

}

return head;

}

void print(struct node\* head)

{

if(head==NULL)

printf("No Polynomial");

else

{

struct node \*temp=head;

while(temp!=NULL)

{

printf("%dx^%d",temp->coeff,temp->expo);

temp=temp->next;

if(temp!=NULL)

printf("+");

else

printf("\n");

}

}

}

void polyAdd(struct node \*head1, struct node \*head2)

{

struct node \*ptr1=head1;

struct node \*ptr2=head2;

struct node \*head3=NULL;

while(ptr1!=NULL && ptr2!=NULL)

{

if(ptr1->expo == ptr2->expo)

{

head3=insert(head3,ptr1->coeff+ptr2->coeff,ptr1->expo);

ptr1=ptr1->next;

ptr2=ptr2->next;

}

else if(ptr1->expo > ptr2->expo)

{

head3=insert(head3,ptr1->coeff,ptr1->expo);

ptr1=ptr1->next;

}

else if(ptr1->expo < ptr2->expo)

{

head3=insert(head3,ptr2->coeff,ptr2->expo);

ptr2=ptr2->next;

}

}

while(ptr1!=NULL)

{

head3=insert(head3,ptr1->coeff,ptr1->expo);

ptr1=ptr1->next;

}

while(ptr2!=NULL)

{

head3=insert(head3,ptr2->coeff,ptr2->expo);

ptr2=ptr2->next;

}

printf("Added Polynomial is: ") ;

print(head3);

}

int main()

{

struct node \*head1=NULL;

struct node \*head2=NULL;

printf("Enter first polynomial\n");

head1=create(head1);

printf("Enter second polynomial\n");

head2=create(head2);

polyAdd(head1,head2);

return 0;

}

OUTPUT:

