Polynomial Addition

Code Id	28
Date Updated	3/7/2010
Title	Polynomial addition

Description

This program uses linked list to add two polynomials.

Codes Snippet

```
#include
#include
void create (struct node **);
void add_node (struct node **, int, int);
void display (struct node *);
struct node (int coef, int deg, struct node *link);
void addpoly (struct node *, struct node *, struct node **);
main()
{
struct node *list = NULL, *list2 = NULL, *flist = NULL;
printf(�ttProgram to add two polynomialsnn�);
printf(�Enter the first list n�);
create(&list1);
printf(�nEnter the second list n�);
create(&list2);
addpoly(list1,list2,&flist);
printf(�n The resultant polynomial is : n�);
display(flist);
return 0;
}

void create(struct node **list)
{
        int degree, coeff;
        char ch;
        do
        {
                printf(�nEnter the degree of x:�);
                scanf(�%d�,&degree);
                printf(�nEnter its coefficient�);
                scanf(�%d�,&coeff);
                add_node(list,degree,coeff);
                printf(�nDo you want to add some other term[y/n]:�);
                fflush(stdin);
                scanf(�%c�,&ch);
        }while(ch==�y�);
        printf(�n The list created is :�);
        display(*list);
}
void add_node ( struct node **list, int degr, int coef)
{
        struct node *p, *q, *newr;
        p=*list;
        q=p;
        while(p != NULL &&  (p->deg > degr ))
        {
                q=p;
                p=p->link;
        }
        newr =  (struct node *) malloc (sizeof (struct node));
        newr->deg=degr;
        newr->coeff=coef;
        if (q==NULL)
        {
                *list=newr;
                newr->link=NULL;
        }

        else
        {
                if (q!=p)
```

```
        {
                printf(“%dx ^ %d”, list->coef,list->deg);
                list = list->link;
        }
        printf(“%d x ^ %d”, list->coef, list->deg);
}
void addpoly(struct node *p1, struct node *p2, struct node **p3)
{
        while(p1!=NULL && p2 != NULL)
        {
                if(p1->deg > p2->deg)
                {
                        add_node(p3,p1->deg,p1->coef);
                        p1= p1-> link;
                }
                else if(p1->deg < p2->deg)
                {
                        add_node(p3,p2->deg,p2->coef);
                        p2= p2-> link;
                }

                else
                {
                        if((p1->coef+p2->coef)!=0)
                        add_node(p3,p1->deg,(p1->coef+p2->coef));
                p1=p1->link;
                p2=p2->link;
        }
}
        if (p1==NULL)
        {
                while(p2 != NULL)
                {
                        add_node(p3,p2->deg,p2->coef);
                        p2=p2->link;
}
        }
else
{
        while(p1!=NULL)
        {
                        add_node(p3,p1->deg,p1->coef);
                        p1=p1->link;
}
        }
}
```