**DS Lab Program**

**Program 9:**

**Design, Develop and Implement a Program in C for the following operations on Singly Circular Linked List (SCLL) with header nodes**

1. **Represent and Evaluate a Polynomial P(x,y,z) = 6x2y2z-4yz5+3x3yz+2xy5z-2xyz3**
2. **Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the result in POLYSUM(x,y,z)**

**Support the program with appropriate functions for each of the above operations**

#include<stdio.h>

#include<math.h>

typedef struct node

{

int cf;

int xex,yex,zex;

struct node \*link;

}NODE;

NODE \*p3=NULL,\*p3head=NULL;

NODE\* insert(NODE \*head,NODE \*last,int n)

{

int i;

NODE \*nn;

if(head==NULL)

{

nn=(NODE\*)malloc(sizeof(NODE));

head=nn;

last=nn;

nn->link=nn;

}

for(i=1;i<=n;i++)

{

nn=(NODE\*)malloc(sizeof(NODE));

printf("enter the coefficient \n");

scanf("%d",&nn->cf);

printf("enter the exponent values for the x, y and z \n");

scanf("%d%d%d",&nn->xex,&nn->yex,&nn->zex);

if(head->link==head)

{

nn->link=head;

head->link=nn;

last=nn;

}

else

{

nn->link=head;

last->link=nn;

last=nn;

}

}

return last;

}

void display(NODE\* last)

{

NODE\* temp=last->link->link;

if(temp==NULL)

{

printf("LIST is empty\n");

return;

}

while(temp!=last)

{

printf("(%dx^%dy^%dz^%d)+",temp->cf,temp->xex,temp->yex,temp->zex);

temp=temp->link;

}

printf("(%dx^%dy^%dz^%d)",temp->cf,temp->xex,temp->yex,temp->zex);

}

void add(NODE \*p1,NODE \*p2,int n1,int n2)

{

NODE \*nn=NULL,\*a,\*b;

int i=0,j=0,comp;

a=p1->link->link;

b =p2->link->link;

nn=(NODE\*)malloc(sizeof(NODE));

p3head=nn;

p3=nn;

nn->link=nn;

while(i<n1 && j<n2)

{

nn=(NODE\*)malloc(sizeof(NODE));

nn->link=p3head;

p3->link=nn;

p3=nn;

if(a->xex == b->xex )

{ nn->cf = a->cf + b->cf;

nn->xex=a->xex;

nn->yex=a->yex;

nn->zex=a->zex;

a=a->link;

b=b->link;

i++;

j++;

}

else if(a->xex > b->xex )

{ nn->cf=a->cf;

nn->xex=a->xex;

nn->yex=a->yex;

nn->zex=a->zex;

a=a->link;

i++;

}

else

{ nn->cf=b->cf;

nn->xex=b->xex;

nn->yex=b->yex;

nn->zex=b->zex;

b=b->link;

j++;

}

}

while(i<n1)

{

nn=(NODE\*)malloc(sizeof(NODE));

nn->link=p3head;

p3->link=nn;

p3=nn;

nn->cf=a->cf;

nn->xex=a->xex;

nn->yex=a->yex;

nn->zex=a->zex;

a=a->link;

i++;

}

while(j<n2)

{

nn=(NODE\*)malloc(sizeof(NODE));

nn->link=p3head;

p3->link=nn;

p3=nn;

nn->cf=b->cf;

nn->xex=b->xex;

nn->yex=b->yex;

nn->zex=b->zex;

j++;

b=b->link;

}

}

void eval(NODE \*p)

{

int x,y,z,res=0,xval,yval,zval;

NODE \*temp;

temp=p->link->link;

printf("\n enter the value of x,y and z\n");

scanf("%d%d%d",&x,&y,&z);

while(temp!=p)

{

xval= pow(x,temp->xex);

yval= pow(y,temp->yex);

zval= pow(z,temp->zex);

res= res+temp->cf\*xval\*yval\*zval;

temp=temp->link;

}

xval= pow(x,temp->xex);

yval= pow(y,temp->yex);

zval= pow(z,temp->zex);

res= res+temp->cf\*xval\*yval\*zval;

printf("\n the result of the polynomial is %d",res);

}

void main()

{

int n1,n2;

NODE \*head1=NULL,\*head2=NULL,\*p1=NULL,\*p2=NULL;

printf("\n enter the number of terms of first polynomial \n");

scanf("%d",&n1);

p1=insert(head1,p1,n1);

eval(p1);

printf("\ enter the number of terms of second polynomial\n");

scanf("%d",&n2);

p2=insert(head2,p2,n2);

eval(p2);

add(p1,p2,n1,n2);

printf("\n the first polynomial is \n");

display(p1);

printf("\n the second polynomial is \n");

display(p2);

printf("\n the resultant polynomial is \n");

display(p3);

}