9. Develop a Program in C for the following operationson Singly Circular Linked List (SCLL) with header nodes

   a. Represent and Evaluate a Polynomial $P(x, y, z) = 6x^2y^2z - 4yz^2 + 3x^2yz + 2xy^2z - 2xyz^3$

   b. 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

→
```c
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

struct node{
        int coef,x,y,z;
        struct node *link;
};

typedef struct node *NODE;
NODE temp,head,cur,x,a=NULL,b,c;

NODE getnode(){
        x=(NODE)malloc(sizeof(struct node));
        return x;
}

NODE readpoly(){
        char ch;
        head=getnode();
        head->coef=head->x=head->y=head->z=-1;
        head->link=head;
        loop:
        temp=getnode();
        printf("\nEnter the coefficient and exponents in decreasing order : ");
        scanf("%d%d%d%d",&temp->coef,&temp->x,&temp->y,&temp->z);
        cur=head;
        while(cur->link!=head)
                cur=cur->link;
        cur->link=temp;
        temp->link=head;
        printf("\nDo you want to enter more coefficients (Y/N) : ");
        fflush(stdin);
        scanf(" %c",&ch);
        if(ch=='y'||ch=='Y')
                goto loop;
        return head;
}

int compare(NODE a,NODE b){
        if((a->x>b->x)||(a->y>b->y)||(a->z>b->z))
                return 1;
        else if((a->x<b->x)||(a->y<b->y)||(a->z<b->z))
                return -1;
        return 0;
}

void attach(int cf,int x1,int y1,int z1,NODE *ptr){
        temp=getnode();
        temp->coef=cf;
        temp->x=x1;
        temp->y=y1;
        temp->z=z1;
        (*ptr)->link=temp;
        *ptr=temp;
}

NODE addpoly(NODE a,NODE b){
        NODE starta;
        int sum,done=0;
        starta=a;
        a=a->link;
        b=b->link;
        c=getnode();
        c->coef=c->x=c->y=c->z=-1;
        NODE lastc=c;
        do{
                switch(compare(a,b)){
                        case -1:
                                attach(b->coef,b->x,b->y,b->z,&lastc);
                                b=b->link;
                                break;
                        case 0:
                                if(starta==a)
                                        done=1;
                                else{
                                        sum=a->coef+b->coef;
                                        if(sum)
                                                attach(sum,a->x,a->y,a->z,&lastc);
                                        a=a->link;
                                        b=b->link;
                                }
                                break;
                        case 1:
                                if(starta==a)
                                        done=1;
                                attach(a->coef,a->x,a->y,a->z,&lastc);
                                a=a->link;
                                break;
                }
        }while(!done);
        lastc->link=c;
        return c;
}

void print(NODE ptr){
        cur=ptr->link;
        while(cur!=ptr){
                printf("%d*x^%d*y^%d*z^%d",cur->coef,cur->x,cur->y,cur->z);
                cur=cur->link;
                if(cur!=ptr)
                printf(" + ");
        }
}

void evaluate(NODE ptr){
        int res=0,x,y,z,ex,ey,ez,cof;

        printf("\nEnter the values of x,y,z : ");
        scanf("%d%d%d",&x,&y,&z);

        cur=ptr->link;

        while(cur!=ptr){
                ex=cur->x;
                ey=cur->y;
                ez=cur->z;
                cof=cur->coef;
                res+=cof*pow(x,ex)*pow(y,ey)*pow(z,ez);

                cur=cur->link;
        }
        printf("\nresult: %d",res);
}

void main(){
        int ch;
        printf("\n1. Represent first polynomial A \
                \n2. Represent Second polynomial B \
                \n3. Display the polynomial A \
                \n4. Display the polynomial B \
                \n5. Add A & B polynomials \
                \n6. Evaluate polynomial C \
                \n7. Exit"
        );

        while(1){
                printf("\n> ");
                scanf("%d",&ch);
                switch(ch){
                        case 1:
                                printf("\nEnter the elements of the polynomial A");
                                a=readpoly();
                                break;
                        case 2:
                                printf("\nEnter the elements of the polynomial B");
                                b=readpoly();
                                break;
                        case 3:
                                print(a);
                                break;
                        case 4:
                                print(b);
                                break;
                        case 5:
                                c=addpoly(a,b);
                                printf("\nThe sum of two polynomials is : ");
                                print(c);
                                printf("\n");
                                break;
                        case 6:
                                evaluate(c);
                                break;
                        case 7:
                                return;
                        default:
                                printf("\nInvalid choice !\n");
                }
        }
}
```