**LAB PROGRAM 9(Evaluating Polynomial and Adding two polynomials using Circular linked list)**

#include<stdio.h>

#include<conio.h>

#include<math.h>

struct node

{

int cf,px,py,pz;

struct node \*next;

int flag;

};

typedef struct node NODE;

NODE \*getnode()

{

NODE \*x;

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

if(x==NULL)

{

printf("memory is not sufficient\n");

return 0;

}

else

return x;

}

NODE \*insert\_rear(int cf,int x,int y,int z, NODE \*head)

{

NODE \*temp,\*cur;

temp=getnode(head);

temp->cf=cf;

temp->px=x;

temp->py=y;

temp->pz=z;

cur=head->next;

while(cur->next!=head)

cur=cur->next;

cur->next=temp;

temp->next=head;

return head;

}

NODE \*read\_poly(NODE \*head)

{

int cf,px,py,pz,ch=1;

while(ch!=0)

{

printf("enter coeff\n");

scanf("%d",&cf);

printf("enter x,y,z powers(0 indicates no term)\n");

scanf("%d%d%d",&px,&py,&pz);

head=insert\_rear(cf,px,py,pz,head);

printf("do you want to enter one more term(if no please enter 0)\n");

scanf("%d",&ch);

}

return head;

}

NODE \*add\_poly(NODE \*h1,NODE \*h2,NODE \*h3)

{

NODE \*p1,\*p2;

int x1,y1,z1,cf1,x2,y2,z2,cf2,cf;

p1=h1->next;

while(p1!=h1)

{

x1=p1->px;

y1=p1->py;

z1=p1->pz;

cf1=p1->cf;

p2=h2->next;

while(p2!=h2)

{

x2=p2->px;

y2=p2->py;

z2=p2->pz;

cf2=p2->cf;

if(x1==x2 && y1==y2 && z1==z2)

break;

p2=p2->next;

}

if(p2!=h2)

{

cf=cf1+cf2;

p2->flag=1;

if(cf!=0)

h3=insert\_rear(cf,x1,y1,z1,h3);

}

else

h3=insert\_rear(cf1,x1,y1,z1,h3);

p1=p1->next;

}

p2=h2->next;

while(p2!=h2)

{

if(p2->flag==0)

h3=insert\_rear(p2->cf,p2->px,p2->py,p2->pz,h3);

p2=p2->next;

}

return h3;

}

void evaluate(NODE \*head)

{

int x,y,z;

float result=0;

NODE \*p;

p=head;

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

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

while(p->next!=head)

{

result=result+((p->cf)\*

(pow(x,p->px)) \*(pow(y,p->py)) \*(pow(z,p->pz)));

p=p->next;

}

result=result+((p->cf) \*(pow(x,p->px)) \*(pow(y,p->py)) \*(pow(z,p->pz)));

printf("result after evaluation%f\n",result);

}

void display(NODE \*head)

{

NODE \*temp;

if(head->next==head)

{

printf("polynomial doesn't exit\n");

return;

}

else

{

temp=head->next;

printf("\n");

while(temp!=head)

{

printf("%dx^%dy^%dz^%d\n",temp->cf,temp->px,temp->py,temp->pz);

if(temp->next!=head)

printf("+");

temp=temp->next;

}

printf("\n");

}

}

void main()

{

NODE \*h1,\*h2,\*h3;int choice;

clrscr();

h1=getnode();

h2=getnode();

h3=getnode();

h1->next=h1;

h2->next=h2;

h3->next=h3;

do

{

printf("1.evaluating polynomial\n2.sum of two polynomials\n3.exit\n");

printf("enter your choice\n");

scanf("%d",&choice);

switch(choice)

{

case 1:printf("enter polynomial for evaluation\n");

h1=read\_poly(h1);

display(h1);

evaluate(h1);

break;

case 2:printf("enter polynomial1 polynomial2 in the order\n");

h1=read\_poly(h1);

h2=read\_poly(h2);

h3=add\_poly(h1,h2,h3);

printf("polynomial 1 is\n");

display(h1);

printf("polynomial 2 is\n");

display(h2);

printf("result of polynomial after addition\n");

display(h3);

break;

case 3:exit(0);

default:printf("invalid choice");

}

}

while(choice!=3);

getch();

}