#include <stdio.h>

#include <stdlib.h>

#include "fun\_proto.h"

#include "fun\_def.h"

int main(){

Node \*hd1,\*hd2;

hd1=CreateEmptyList();

hd2=CreateEmptyList();

int num1,num2;

printf("Enter the number of terms for first polynomial: ");

scanf("%d", &num1);

for(int i=0;i<num1;i++){

int coe,exp;

printf("Enter the coe and exp: ");

scanf("%d %d", &coe,&exp);

insert\_node(hd1,coe,exp);

}

printf("Enter the number of terms for second polynomial: ");

scanf("%d", &num2);

for(int i=0;i<num2;i++){

int coe,exp;

printf("Enter the coe and exp: ");

scanf("%d %d", &coe,&exp);

insert\_node(hd2,coe,exp);

}

Node \*t1,\*t2,\*r;

for(t1=hd1->next;t1!=NULL;t1=t1->next){

printf("%d %d\t",t1->coeff,t1->power);

}

printf("\n");

for(t2=hd2->next;t2!=NULL;t2=t2->next){

printf("%d %d\t",t2->coeff,t2->power);

}

printf("\n");

Node \*result1,\*result2;

result1=add\_poly(hd1,hd2);

result2=multip\_poly(hd1,hd2,num1,num2);

printf("Added Polynomial\n");

for(r=result1->next;r!=NULL;r=r->next){

printf("%d %d\t",r->coeff,r->power);

}

printf("\n");

printf("Multiplied Polynomial\n");

for(r=result2->next;r!=NULL;r=r->next){

printf("%d %d\t",r->coeff,r->power);

}

printf("\n");

}

/\* fun\_proto.h

struct l\_list{

int coeff;

int power;

struct l\_list \*next;

};

typedef struct l\_list Node;

void insert\_node(Node \*hd, int coe,int exp);

Node \*CreateEmptyList();

Node \*add\_poly(Node \*hd1,Node \*hd2);

Node \*multip\_poly(Node \*hd1,Node \*hd2,int size1,int size2);

\*/

/\*fun\_def.h

Node \*CreateEmptyList(){

Node \*h;

h=(Node\*)malloc(sizeof(Node));

h->next=NULL;

return h;

}

void insert\_node(Node \*hd, int coe,int exp){

Node \*new,\*temp;

new=(Node \*)malloc(sizeof(Node));

new->coeff=coe;

new->power=exp;

new->next=hd->next;

hd->next=new;

}

Node \*add\_poly(Node \*hd1,Node \*hd2){

Node \*res,\*p,\*q;

res=CreateEmptyList();

p=hd1->next;

q=hd2->next;

while(p!=NULL & q!=NULL){

if(p->power>q->power){

insert\_node(res,p->coeff,p->power);

p=p->next;

}

else if(q->power>p->power){

insert\_node(res,q->coeff,q->power);

q=q->next;

}

else if(p->power==q->power){

if(p->coeff + q->coeff != 0)

insert\_node(res,p->coeff + q->coeff,p->power);

p=p->next;

q=q->next;

}

}

if(p != NULL){

while(p != NULL){

insert\_node(res,p->coeff,p->power);

p=p->next;

}

}

if(q != NULL){

while(q != NULL){

insert\_node(res,q->coeff,q->power);

q=q->next;

}

}

return res;

}

Node \*multip\_poly(Node \*hd1,Node \*hd2,int size1,int size2){

Node \*res,\*p,\*q,\*res\_h1,\*res\_h2;

int t\_size=size1 \* size2;

res=CreateEmptyList();

p=hd1->next;

while(p != NULL){

q=hd2->next;

while(q != NULL){

insert\_node(res,p->coeff \* q->coeff,p->power + q->power);

q=q->next;

}

p=p->next;

}

return res;

}

\*/

/\*OUTPUT

Enter the number of terms for first polynomial: 5

Enter the coe and exp: -7 0

Enter the coe and exp: 3 1

Enter the coe and exp: -22 4

Enter the coe and exp: 8 8

Enter the coe and exp: 3 12

Enter the number of terms for second polynomial: 5

Enter the coe and exp: -9 1

Enter the coe and exp: 6 5

Enter the coe and exp: -8 8

Enter the coe and exp: -10 9

Enter the coe and exp: 7 14

3 12 8 8 -22 4 3 1 -7 0

7 14 -10 9 -8 8 6 5 -9 1

Added Polynomial

-7 0 -6 1 -22 4 6 5 -10 9 3 12 7 14

Multiplied Polynomial

63 1 -42 5 56 8 70 9 -49 14 -27 2 18 6 -24 9 -30 10 21 15 198 5 -132 9 176 12 220 13 -154 18 -72 9

48 13 -64 16 -80 17 56 22 -27 13 18 17 -24 20 -30 21 21 26

\*/