一元多项式计算实验报告

1. 问题描述

从文件读入两个一元多项式，用链表数据结构表示；

数据文件格式为：#项数 #系数1 #指数1 #系数2 #指数2 …

分别对这两个多项式相加和相乘，生成结果链表，且结果链表要按幂级数排序，将结果按输入文件的格式输出到数据文件；

1. 设计思路

使用fscanf读入第一行第一个数据，作为项数；

使用for循环每次两个地读入第一行剩下的数据，构建第一个多项式链表；

对第二行重复上述步骤，构建第二个多项式链表；

对于加法操作，首先构建一个结果链表，头结点指数为两个多项式指数中最大的，尾结点指数为负无穷；使用双重循环，第一重循环遍历第一个多项式，第二重循环遍历第二个多项式，将两个多项式中指数相同的项相加，并按从大到小顺序插入到结果链表中，此时两个多项式中指数不同的项没有被加入到结果链表，将这些项按大小顺序插入到结果链表中，得到加法运算的结果链表；

对于乘法操作，首先构建一个结果链表，头结点指数为两个多项式指数中最大的，尾结点指数为负无穷；使用双重循环，将两个多项式中每一项分别相乘，插入结果链表，每次插入前遍历一遍结果链表，若有指数相同的项，则直接把系数相加，不必执行插入操作；

分别遍历两个结果链表，记录项数，将结果分行输出，并使用fprintf将其按格式输出到新文件中；

将方法封装到Polyno类中，方法声明分别为

void build(FILE \*file, Node \*head);

void input(Node \*first, Node \*second);

void print(Node \*head, int &cnt);

void add(Node \*first, Node \*second, Node \*ans);

void multiply(Node \*first, Node \*second, Node \*ans);

void output(Node \*ans1, Node \*ans2, int cnt1, int cnt2);

在Polynomial.cpp中创建Polyno对象，调用上述方法，完成问题要求。

1. 测试结果  
   输入文件和输出文件



