### 实验4一元多项式相减

1.实验目的

（1）熟练掌握单链表的基本操作；

（2）会用单链表解决一些实际应用。

2.实验内容

（1）编写程序，通过键盘输入两个一元多项式headA和headB，要求能够按照指数升序排列，输出多项式。

（2）对两个一元多项式进行相减运算，结果存储到headA中并将结果输出。

3.程序分析

（1）在通常的应用中，一元多项式的次数可能很高并且可能变化很大。如按次数采用顺序存储结构存储多项式，使得顺序存储结构的最大长度很难确定，特别是在处理项数少且次数特别高的多项式时，对内存空间的浪费是相当大的。因此，一般情况下，都是采用链式存储结构来处理多项式的运算。一个线性链表表示一个一元多项式，每个结点表示多项式中的一项。

（2）在数学上，一个一元n次多项式Pn（x）可按降序写成：

Pn（x）=pnxn+pn -1xn -1+…+p1x1+p0

它是由n+1个系数唯一确定的。因此，在计算机里它可以用一个线性P来表示：

P=(pn ,pn -1 ,…p1 ,p0)

每一项的指数i隐含在其系数pi的序号里。

一个一元多项式的每一个子项都是由“系数—指数”两部分来组成的，所以可以将它抽象成一个由”系数—指数”对构成的线性表，结点的定义如下：

private class Term {

public double coef = 0; //系数

public double exp = 0; //指数

}

（3）根据建立的两个多项式ha和hb进行相减的运算，存放在ha中，并不改变hb链表。可以采用如下的方法进行设计：设指针p、q分别指向描述多项式的链表ha和hb的第一个结点， p0指向p的前驱。按程序要求，当p与q同时不为空时，p与q的指数域进行比较，如果p结点的指数大于q结点的指数，则新增一个结点s，插入到p0与p之间，q后移一位；如果p结点的指数小于q结点的指数，则p0与p同时后移一位；如果p结点的指数等于q结点的指数，则将两个结点的指数相减，如相减后的指数不为0，要修改p结点的指数，如相加后的指数为0，要删除p结点，然后p与q同时后移一位。最后，如果hb链表中有剩余结点，即q不为空，将q之后剩余的所有结点复制插入到ha链表中，注意结点的系数必须变号。

（4）多项式的显示，可以使用文本界面，用”系数—指数”对的形式表示表达式，如ha：(1，1)🡪（2，2）🡪（-3，100），代表的一元多项式为Pn（x）=x+2x2-3x100。

（5）介绍两个类

① import java.util.ArrayList;

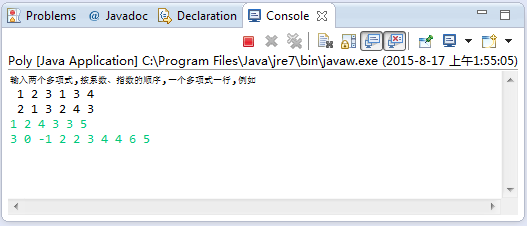
导入java.util包中的ArrayList类，底层采用数组结构（组长度是可变的百分之五十延长）进行存储，特点是查询很快，但增删代价较大。

②import java.util.List;

导入java.util包中的List类，List中元素是有序的(怎么存的就怎么取出来，顺序不会乱)，元素可以重复

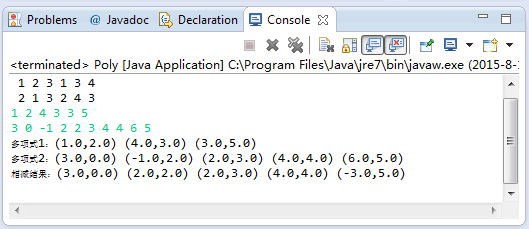
4.调试测试

（1）输入：按提示输入2个多项式的各个“系数—指数”对，以“回车”结束。如图4-1所示。



**图4-1** 输入

1. 输出：输出两个多项式相减后的结果。如图4-2所示。



**图4-2** 输出

5.思考题

如何保证在建立一元多项式时保证多项式链表按指数升序排列？