readme

1. 程序功能说明

本程序能够进行一元多项式加减运算。

2. 程序运行所需环境

编写时使用了Eclipse。

3. 输入说明

3.1 只能由控制台或命令行输入，输入仅有一行，不能使用文件输入。

3.2 由“0-9 + - , ( ) { } ”这几种字符和“空格”组成，输入者通过键入回车键确认输入结束。

3.3 标准的输入由若干多项式、多项式之前的正负号或两个多项式之间的一个正负号，以及若干空格组成，相邻两个多项式之间有且仅有一个‘+’或‘-’，第一个多项式之前可能存在正负号（若有，则仅有一个），每个多项式内的数对中的n都不相同。

3.4 一个多项式的标准格式为：{(c1,n1),(c2,n2),…,(cm,nm)}，最外端由花括号与其它多项式区分开，内部是若干由括号包含的数对，括号之间由一个逗号‘,’分隔，两个数之间也由一个‘,’分隔。

3.5 对于数对(c,n)，c为系数，有-10^6<c<10^6, n为幂，有0<=n<10^6，即c最大为6位十进制整数，n最大为6位十进制整数。所有数均可能出现前导0，但c和n除去符号位之外的长度分别不超过6和6。

3.6 每个多项式限制为最多50个数对，多项式的个数限制为最多20个。

3.7 输入中的空格和Tab将被直接忽略。

3.8 必须按照上述标准格式输入，即否则将显示错误信息“The Polys are not in the right state.”

输入样例：

{(3,0), (2,2), (12,3)} + {(3,1), (-5,3)} - {(-199,2), (29,3),(10,7)}

{(2,9),(1,5)}-{(2,9),(-3,10)}

4. 输出格式

4.1 程序的正常运行输出结果：第一行为“The answer:”，第二行为一个符合标准格式的多项式表达式，形如{(c1,n1),(c2,n2),…,(cm,nm)}，并按照多项式内的单项式次数进行升序排序。

标准输出样例：

The answer:

{(3,0), (3,1), (201,2), (-22,3), (10,7)}

4.2 如果答案为0，则输出“{}”

4.3 如果输入的多项式超过20个，则显示错误信息“You have input more than 20 Polys!”

4.4 如果某一多项式中的数对超过50个，则显示错误信息“There are more than 50 terms in one poly!”

4.5 如果某一多项式中出现了重复的指数，将显示错误信息“The degree in the Poly is repeated!”

注：若数据含有上述两种及以上的错误，只能显示一种错误信息。

5. 程序控制流程图

逐个解析字符串，

将数据存入对象Term，

构造对象Poly

按格式输出计算结果

将多项式逐项相加

Y

N

判断输入多项式和数对个数是否符合题意

将字符串在“{”处分割，

形成多项式

显示错误信息

Y

N

进行正则表达式匹配

去空格

读入一行字符串