

# 面向对象设计与构造——第一次作业

## JAVA 语言程序说明文档

2018/3/14

### 1. 项目功能

能够对至多 20 个符合输入要求的多项式进行加减计算，并按格式输出

### 2. 项目运行环境

建议项目运行环境：JRE 8

### 3. 程序输入说明

- 程序从控制台获取键盘输入
- 标准输入格式：本程序的合法输入格式如下

输入仅有一行，由“0-9 + - ( ) { } ”这几种半角字符和“空格”组成，通过键入回车键确认输入结束，标准的输入由多组代表多项式的符号集合组成，形式如下“{多项式 1}+{多项式 2}-.....+{多项式 n}”。每组花括号内为一个多项式字符串，所有花括弧前通过‘+’或‘-’代表多项式之间的加减关系，若第一组花括号前没有则默认为‘+’。

花括号内的多项式由数对(c,n)组成，其中 c 为系数，n 代表次数。一个多项式的标准格式为：

{(c1,n1),(c2,n2),...,(cm,nm)}，最末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组成，括号之间由一个逗号‘,’分隔，两个数之间也由一个‘,’分隔。一个多项式内的数对中的 n 都不相同

(如果出现重复的 n，则丢弃之前所输入的，例如，在某个多项式内有以下三项：(1,1)，(2,1)，(3,1)，则只保留最后一项)。

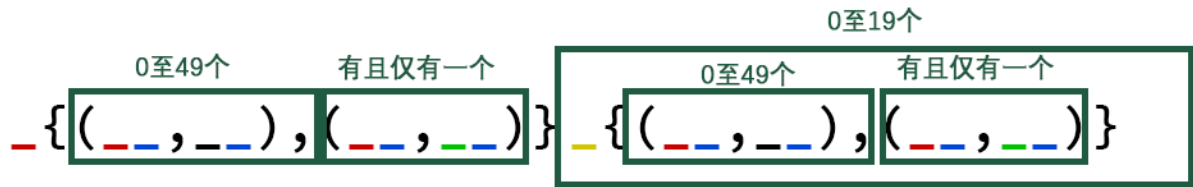
组成全部多项式的字符串中可以存在空格，但是所有空格应该被直接忽略。如{(c1,n1),(c2,n2)}这样的表达是被允许的，如果出现‘1 24’则被当成‘124’

在本次编程过程中，对于数对(c,n)，c 为系数，为十进制整数，取值范围为-999999 ≤ c ≤ 999999

(即 c 前面可以有‘-’)；n 为该项的幂，为十进制整数，有 0 ≤ n ≤ 999999。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。

输入时，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。

综合而言，合法输入的格式（去除所有空格后）可以概括为：



■ 一个可选的+或-

■ 1至6位数字

■ 一个可选的+或- (-仅限于后面所跟的数字全部为0时)

■ +或-

- 对于错误输入的响应：如果输入的格式不符合上方所定义的标准输入格式，将报错并退出程序，报错的格式以及定义的三种输入错误类型如下

- 对于某个多项式的格式输入有误，包括多项式内的非法字符、过多的项数等

```
ERROR
# 某个多项式输入有误
```

- 某两个多项式之间的符号有误，包括出现了+和-以外的字符以及字符重复等

```
ERROR
# 多项式之间的符号错误
```

- 输入了过多的多项式

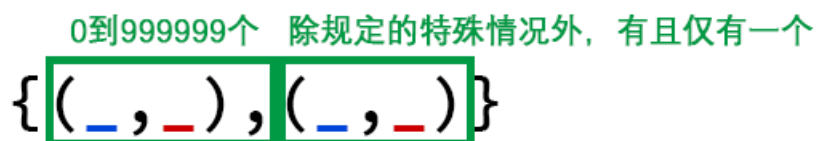
```
ERROR
# 多项式总数超出 20 项
```

- 去除所有空格之后，字符串为空

```
ERROR
# 没有输入多项式，请检查输入
```

## 4. 程序的输出结果

程序的输出格式如下：



■ 一个整数，取值范围为[-999999,999999]

■ 一个正整数，取值范围为[0,999999]，此部分按升序排列

输出样例：

$\{(c_1, n_1), (c_2, n_2), (c_3, n_3) \dots\}$

特殊规定：

若某一个指数对应的系数为 0，则不输出

如果所有的系数均为 0，则直接输出 0

如果输入格式有误，程序将报错，不会返回计算结果，参见上一节

## 5. 程序控制流程图

