

面向对象第一单元第一次作业指导书

摘要

本次作业，需要完成的任务为简单多项式导函数的求解。

问题

设定

首先是一些基本概念的声明：

- **带符号整数** 支持前导0的十进制带符号整数（若为正数，则正号可省略），无进制标识，如：
`+02`，`-16`，`19260817`。
- **因子**
 - **常数因子** 包含一个带符号整数，如：`233`。
 - **变量因子**
 - **幂函数**
 - **一般形式** 由自变量`x`，指数符号`**`和指数组成，指数为一个带符号整数，如：`x ** +2`。
 - **省略形式** 当指数为1时，可省略指数符号`**`和指数，如：`x`。
- **项**
 - **一般形式** 由乘法运算符连接若干因子组成，如：`2 * x ** 2 * 3 * x ** -2`。
 - **特殊形式**
 - 第一个因子为常数因子，值为1且**总共不止有一个因子**时，可省略该常数因子或表示为正号开头的形式，如：`x ** 2 * x ** -1`、`+ x ** 2`。
 - 第一个因子为常数因子，值为-1且**总共不止有一个因子**时，可表示为负号开头的形式，如：`-x ** 2`。
- **表达式** 由加法和减法运算符连接若干项组成，如：`-1 + x ** 233 * x ** -1 - x ** 06`。此外，**在第一项之前，可以带一个正号或者负号**，来表示第一项的正负，如：`- -1 + x ** 233`，`+ -2 + x ** 19260817`。
 - 项的个数至少为一，**空串不属于合法的表达式**。
- **空白字符** 在本次作业中，空白字符包含且仅包含空格 `<space>`（ASCII值32）和水平制表符 `\t`（ASCII值9）。其他空白字符，均属于非法字符。

对于空白字符，有以下几条规定：

- 带符号整数内不允许包含空白字符，注意带符号整数本身的**符号与整数之间**也不允许包含空白字符。
- 指数运算符内不允许包含空白字符，如 `* *` 不合法。
- 因子、项、表达式，在不与以上两条规定矛盾的前提下，可以在任意位置包含任意数量的空白字符。

设定的形式化表述

- 表达式 \rightarrow 空白项 [加减 空白项] 项 空白项 | 表达式 加减 空白项 项 空白项
- 项 \rightarrow [加减 空白项] 因子 | 项 空白项 * 空白项 因子
- 因子 \rightarrow 变量因子 | 常数因子
- 变量因子 \rightarrow 幂函数
- 常数因子 \rightarrow 带符号的整数
- 幂函数 \rightarrow x [空白项 指数]
- 指数 \rightarrow ** 空白项 带符号的整数
- 带符号的整数 \rightarrow [加减] 允许前导零的整数
- 允许前导零的整数 \rightarrow (0|1|2|...|9){0|1|2|...|9}
- 空白项 \rightarrow {空白字符}
- 空白字符 \rightarrow (空格) | \t
- 加减 \rightarrow + | -

其中{}表示0个、1个或多个，[]表示0个或1个，|表示多个之中选择。

式子的具体含义参照其数学含义。

若输入字符串能够由“表达式”推导得出，则输入字符串合法。具体推导方法请参阅“第一单元形式化表述说明”文档。

描述

求导是数学计算中的一个计算方法，其定义为，当自变量的增量趋于零时，因变量的增量与自变量的增量之商的极限。

在本次作业中，需对输入表达式进行求导运算，化简并输出运算结果。

本次作业可能用到的求导公式有：

$$\text{I. 当 } f(x) = c \text{ (} c \text{ 为常数) 时, } f'(x) = 0$$

$$\text{II. 当 } f(x) = x^n \text{ (} n \neq 0 \text{) 时, } f'(x) = nx^{n-1}$$

例如：

$$\text{当 } f(x) = 2x^6 + 6x^4 \text{ 时, } f'(x) = 12x^5 + 24x^3$$

输入为 `2*x**6+6*x**4`，输出为 `12*x**5+24*x**3`。

一些规定

- 一个表达式可能有多个解释。因此，对于一个表达式，只要存在一条合法解释，该表达式即合法，我们保证这些解释在数学意义上均相等。

判定

输入格式

输入包含且仅包含一行，表示一个表达式。

本次作业保证输入数据全部为合法（格式正确）的表达式，不需要进行格式检查！！

数据限制

- 数据的最大长度为1000（请注意，这里不是有效长度，是去除右侧换行符后的总长度），并且是合法的表达式。

上述限制被定义为**数据基本限制**。在测试中，不会出现不符合该限制的数据。在此限制内，不作任何其他限制。

输出格式

由于本次作业输入数据全部为合法的表达式，因此程序不需要对输入数据的合法性进行判定。

关于输出：

- 应当输出一行，表示求得的导函数。格式同样需要符合前述表达式基本规则，其中要求带符号整数均为**十进制形式**。

判定模式

正确性判定

对于这次作业结果正确性的判定，在输出表达式符合格式要求的前提下，输出表达式与正确答案对应表达式**恒等**即为正确。其中，恒等定义如下。

设 $f(x)$ 的定义域为 D_1 ， D_1 包含于 R ， $g(x)$ 的定义域为 D_2 ， D_2 包含于 R ，对任意 $x \in D_1 \cap D_2$ ， $f(x) = g(x)$ 成立。

性能分判定

在本次作业中，性能分的唯一评判依据，是输出结果的有效长度。

有效长度定义为，输出结果去除所有的空白字符（<space>、\t）后的长度，设为 L 。

设某同学给出的正确答案的有效长度为 L_p ，所有人目前给出的正确答案的有效长度的最小值为 L_{min} 。

设 $x = \frac{L_p}{L_{min}}$ ，则该同学性能分百分比为：

$$r(x) = 100\% \cdot \begin{cases} 1 & x = 1 \\ -31.8239x^4 + 155.9038x^3 - 279.2180x^2 + 214.0743x - 57.9370 & 1 < x \leq 1.5 \\ 0 & x > 1.5 \end{cases}$$

简单来说，就是这样：

x	$r(x)$
1.0	100.0%
1.05	79.9%
1.1	60.5%
1.2	29.0%
1.3	10.9%
1.4	4.5%
1.5	0.0%

需要注意的是，**获得性能分的前提是，在正确性判定环节被判定为正确**。如果被判定为错误，则性能分部分为0分。

互测相关

在互测环节

- 数据须符合**数据基本限制**。

如果不满足上述数据限制，该数据将被系统忽略，不会对被测程序进行测试。

样例

#	输入	输出	解释
1	1	0	根据表达式定义可得。
2	$4x+x^2+x$	$2x+5$	根据表达式定义可得。
3	$4x+x^2+x$	$4+2x+1$	未合并同类项，但表达式依然等价。
4	$-4x + x^2 + x$	$2x+5$	$-4x$ 为合法项，且表达式第一项前也可以包含正负号。
5	$+4x - x^2 + x$	$2x+5$	$-x^2$ 为合法项。
6	$+19260817x$	19260817	根据表达式定义可得。
7	$+ 19260817x$	19260817	多项式第一项前可以带有正负号。
8	$+ +19260817x$	19260817	$+19260817x$ 为合法项，开头可以带有正负号。
9	$43+4x^3$	$12x^2$	根据表达式定义可得。
10	$2x^2^3$	$12x$	根据表达式定义可得。

注意：由于本作业可被判定为正确的答案不唯一，以上样例的输出**仅保证正确性，但并不一定为性能最优解**。

补充信息

关于评测

- 评测时，会自动忽略掉行末的空格以及文件末多余的回车。
- 对于输入，如果包含多行，则忽略第一行以后的内容即可。
- 类似地，对于输出结果，如果包含多行，则在评测时将忽略第一行以后的内容。（也就是说，你们可以在正文之后附加一些其他的信息以改善自己调试的体验）

一点点提示

- Java内的原生整数类型有 `long` 和 `int`，长度分别为64位和32位。
- 如果觉得上述数据类型不够用的话，可以百度一下Java内可以怎样快速处理这个问题，也可以回顾一下pre第2弹的指导书。
- 在Java内，不建议使用静态数组。推荐使用 `ArrayList`、`HashMap`、`HashSet` 一类的数据结构，快速管理和调配手中无序的数据。
- 关于输入字符串的处理，推荐使用**正则表达式**，相关的API可以了解 `Pattern` 和 `Matcher` 类。

一点点想说的话

- **不要重复造轮子！不要重复造轮子！不要重复造轮子！重要的事情说三遍**
- 我们鼓励大家通过Baidu、Google、Stackoverflow等方式自行学习和解决问题。
- 如果还有更多的问题，请到讨论区提问。但是**请善用讨论区**，并在此之前认真阅读包括但不限于课程要求文档、指导书、搜索引擎结果等的内容。