**Project 1多项式与表达式求值计算器**

**【问题描述】**

设计一个一元稀疏多项式简单计算器和算数表达式求值计算器。

**【基本要求】**

编写一个程序，分别实现如下**2项**功能，并引导用户选择对应的功能：

* **一元稀疏多项式简单计算器的基本功能是：**

1. 输入并建立多项式。
2. 输出多项式，输出形式为整数序列: ，其中是多项式的项数，和分别是第项的系数和指数，序列按指数降序排列。
3. 多项式和相加，建立多项式+。
4. 多项式和相减，建立多项式-。
5. 计算多项式在处的值。
6. 求多项式的导函数(简单功能即可，不要求特殊类型函数)。
7. 多项式和相乘，建立乘积多项式。

* **算法表达式求值计算器的基本功能是：**

以字符序列的形式从终端输入语法正确的、不含变量的整数表达式。利用下表给出的算符优先关系，实现对算术混合运算表达式的求值，并仿照求值中运算符栈、运算数栈、输入字符和主要操作的变化过程。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 𝜃2  𝜃1 | + | - | \* | / | ( | ) | #（结束符） |
| + | > | > | < | < | < | > | > |
| - | > | > | < | < | < | > | > |
| \* | > | > | > | > | < | > | > |
| / | > | > | > | > | < | > | > |
| ( | < | < | < | < | < | = | --- |
| ) | > | > | > | > | --- | > | > |
| # | < | < | < | < | < | --- | = |

注：𝜃1<𝜃2表示 𝜃1的优先级低于 𝜃2

**【加分项】**

1. 对于算数表达式求值计算器，扩充运算符集，如增加乘方、单目减、赋值等运算。
2. 计算器的功能和仿真界面（可参考Windows / Mac计算器的高级功能），可以考虑使用QT，MFC等GUI支持库。

**【测试数据】**

**一元稀疏多项式简单计算器：**

1. 
2. 
3. 
4. 
5. 
6. 

**算法表达式求值计算器：**

3\*(7-2); 8; 1+2+3+4; 88-1\*5; 1024/4\*8; 1024/(4\*8); (20+2)\*(6/2);

3-3-3; 8/(9-9); 2\*(6+2\*(3+6\*(6+6))); (((6+6)\*6+3)\*2+6)\*2;

备注：**这里的测试数据仅是列举，同时不保证都是正确的表达式**，程序应具备处理异常情况的功能。可用文件将测试数据先预存好，演示时直接读入数据文件；也可以设计方便的输入界面，进行即时数据录入。

**【实现提示】**

1. 本练习重点在于掌握和理解线性表的朴素编码实现方式，**因此回避使用C++标准模版库（STL）提供的容器、迭代器和算法**（如可直接使用的查找、排序、交换等操作）。
2. 用线性表的方式存储多项式，并自定义输入和输出的表示形式，可尝试采用多种数据结构形式实现。
3. 设置运算符栈和运算数栈辅助分析算符优先关系，在读入表达式的字符序列的同时，完成运算符和运算数（整数）的识别处理，以及相应的运算。
4. 在识别出运算数的同时，要将其字符序列形式转换成整数形式。
5. 在程序的适当位置输出运算符栈、运算数栈、输入字符和主要操作的内容。
6. **代码要求使用C/C++语言进行编写**，但也可以探索多语言混合编程的方式，以获得更好的展示效果。操作系统Windows / Linux / Mac等环境皆可。

**【检查时间和要求】**

**2025学年秋季学期第7周实验课（2025年10月22日）**，检查演示可使用自己的笔记本电脑或实验室台式电脑。

评分要求：功能实现(50%)，程序运行界面(30%)，代码规范及注释(20%)。

将**源代码**（仅代码文件，zip格式压缩）和**实验报告**，在课程群对应的收集作业任务中提交。

ZIP文件命名格式：学号+姓名+Project1 (示例：24332001+张三+Project1)

-------------------------------------------------------------

附实验报告内容模板参考：

**Project1实验报告**

学号 姓名

1、程序功能简要说明。

2、程序运行截图，包括计算功能演示、部分实际运行结果展示、命令行或交互式界面效果等。

3、部分关键代码及其说明。

4、程序运行方式简要说明。