**Project1实验报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | 24336146 | 姓名 | 姚竣耀 |
| 1. 程序功能简要说明 本程序是前一版多项式计算器的升级版本，新增了对浮点指数（包括负数指数）的支持，并优化了输入逻辑，使用户可以直接指定项数并逐项输入系数与指数。程序仍为命令行交互式系统，支持多项式 a(x) 与 b(x) 的输入、输出及运算。  主要功能如下：  1. 按项数交互式输入多项式 a 与 b，支持整数与小数系数、负数指数与小数指数。 2. 输入过程中支持撤回、替换与重新输入。 3. 多项式运算包括加法、减法、乘法、除法（返回商与余数）。 4. 支持在给定 x 值下的求值运算。 5. 输出支持两种模式：序列输出与人类可读形式输出（如 3x^(2.5) - x^(-1) + 7）。 6. 所有菜单均提供返回上级、撤回与确认操作提示。 | | | |
| 2. 程序运行示例 1. 主菜单界面     1. 输入多项式界面      1. 撤回功能演示      1. 浮点指数与负数指数输入演示      1. 运算功能演示：加、减、乘、除 运算基于输入多项式a:x^(1) b:0.2x^(4)+2.3x^(-1)+1.2x^(-3.2) 加法演示   减法演示  乘法(下面只演示按序列输出)  除法  6. 在 x 处求值功能演示  7. 输出形式对比演示（序列输出） | | | |
| 3. 关键代码及说明 关键代码示例 1：Polynomial::add（多项式加法）  说明：实现两个多项式的逐项相加操作。遍历多项式 a 与 b 的所有项，根据指数（exp）是否相同决定是否决定是否合并。若指数相同则将系数相加，否则保留原项。 1.使用 std::vector<Term> 存储项（Term 包含 coeff 与 exp）。  2.利用 fabs(a.exp - b.exp) < EPS 判断指数是否相等（浮点比较容差）。  3.加法完成后调用 normalize() 去除系数接近 0 的项并排序。  关键代码示例 2：Polynomial::mul（多项式乘法）  说明：实现两个多项式的乘法运算。采用双重循环方式，将第一个多项式的每一项与第二个多项式的每一项相乘，生成新项并累加至结果多项式中。 1.每一项乘积的指数为 exp = a.exp + b.exp，系数为 coeff = a.coeff \* b.coeff。  2.中间结果存储在 std::map<double, double> 结构中，自动按指数合并同类项。  3.最后将 map 转回 vector 并排序降序输出。  关键代码示例 3：Polynomial::div（多项式除法）  说明：使用长除法算法计算商与余数，并处理浮点指数的比较与归一化。 取被除式（dividend）的最高项与除式（divisor）的最高项相除，得到商项：  q.coeff = a.coeff / b.coeff; q.exp = a.exp - b.exp;  1.将 (divisor \* q) 从 dividend 中减去，更新余数。  2.循环执行直到余数的最高指数小于除式最高指数。  3.返回 std::pair<Polynomial, Polynomial> 作为（商，余数）。  关键代码示例 4：Polynomial::evaluate（在 x 处求值）  说明：计算多项式在给定实数 x 下的值。对每一项执行 coeff \* pow(x, exp) 并求和。 1.遍历所有项，利用 std::pow() 函数计算幂。  2.考虑负指数时自动转换为分数形式（如 x^-1 = 1/x）。  3.输出为 double 类型，支持浮点输入（例如 x=2.5）。 | | | |
| 4. 程序运行方式及简要说明 本程序可在 Visual Studio 或命令行环境下运行。建议使用 UTF-8 编码并启用 C++17 标准。  编译命令示例（命令行）：  g++ -std=c++17 Polynomial.cpp Menu.cpp main.cpp -o PolynomialCalculator  运行命令示例：  ./PolynomialCalculator # Linux 或 MinGW PolynomialCalculator.exe # Windows  注意：在 Visual Studio 中运行时，请确保控制台输出支持 UTF-8，否则指数括号可能显示异常。 | | | |
| 5. 实验总结 本次实验在前一版本的基础上扩展了对浮点指数和负数指数的支持，解决了多项式输入逻辑不兼容问题。实验过程中重点处理了指数比较的容差问题（EPS）与浮点输入的鲁棒性。通过此次实验，进一步掌握了类设计、菜单逻辑控制以及面向对象程序结构的调试方法。 | | | |