



|  |  |  |  |
| --- | --- | --- | --- |
| **学号** | 20009200713 | **姓名** | 曾凡浩 |
| **班级** | **2003052** | **任课教师** | **张淑平** |
| **实验名称** | 第3次实验 | | |
| **实验学期** | **2021 – 2022 学年第2学期** | | |
| **实验日期** | 2022年4月18日 | **实验地点** |  |
| **报告成绩** |  | | |

西安电子科技大学 计算机科学与技术学院

# 实验目的

通过实现一个计算器程序，熟悉 C++为提供的模块与分别编译相关机制，掌握模块化程序设计范型的基本策略，能够灵活运用相应机制，提高编程水平

# 实验环境

操作系统：Window 10

开发工具：Visual Studio2020

# 实验内容

## 题目1名称 改造桌面计算器

本题要求 ：

采用多个头文件、多个源文件的方式实现完整程序，使得源代码的物理结构和逻辑结构保持一致；

输入中每遇到一个分号或回车(‘\n’)就认为一个完整表达式结束，并将该完整表达式记作一行。在处理过程中累计行数；

对于输入中存在错误时，除了提示错误现象外，还需提示相应的出错行号。

必须支持命令行参数，用以指明从何处读取数据。命令行参数中的第2项开始为输入文件之路径，可省略。

输入文件就是普通的文本文件，其中预先输入了若干表达式，内容和来自标准输入的完全相同。

若命令行参数未指明输入文件，则程序从标准输入设备(cin)读取表达式；

若命令行指明了1~N个输入文件，则程序依次从这些文件读入表达式。

# 数据结构与算法说明

## 改造桌面计算器

**模块结构及文件组织设计：**

模块1：主控模块，仅包括文件main.cpp ，定义了 main()函数。

模块2：词法分析模块，包括以下两个文件：

Lexer.hpp 定义了词法分析操作接口的声明，枚举类型以及全局变量的声明；

Lexer.cpp 实现了词法的具体操作，以及全局变量的定义。

模块3：语法分析模块，包括以下两个文件：

Paser.hpp 定义了语法分析操作的接口声明；

Paser.cpp 实现了语法分析的具体操作；

模块4：错误处理模块，包括以下两个文件：

Error.hpp 定义了错误分析操作的接口声明，以及错误处理时标志变量的声明；

Error.cpp 实现了错误分析的具体操作，以及标志变量的定义及操作；

**关键数据结构设计：**

数据结构1：定义了枚举类型 token\_type {

NUMBER,

PLUS = '+',

MINUS = '-',

MUL = '\*',

DIV = '/',

LP = '(',

RP = ')',

PRINT = ';',

ASS = '=',

ERR\_TOKEN,

NAME,

END

};

**算法1.1 int main()**

作 用：主控函数，也实现对题目所需其他内容的测试。

参 数：int argv, char \*argv[]

返回值：总是返回0。

计算过程：

通过判断argc是否为1来确定由键盘输入还是命令行文件输入；

调用算法get\_token（），获取到第一个字符代表什么；

循环调用算法expr（flase），计算各式的结果；

**算法2.1** token\_type get\_token()

作 用：对于输入的字符变量，获取对应的实际含义。

参 数：void

返回值：然后一个枚举类型的token\_type.

计算过程：

获取到输入流转化成char类型的ch。

运用switch--case来对ch进行分类判断返回实际含义；

**算法2.2** void skip()

作 用：当在表达式过程中发生错误时跳过该表达式；

参 数：void

返回值：void

计算过程：

循环获取输入流中的字符信息直到该行表达式结束；

**算法3.1** double expr(bool)

作 用：在计算过程中完成加减法操作；

参 数：bool

返回值：double

计算过程：

通过递归调用term来保留之前计算过的结果；

运用switch--case分别进行加减操作；

**算法3.1**  double term(bool)

作 用：在计算过程中完成乘除法操作；

参 数：bool

返回值：double

计算过程：

通过递归调用prim来保留之前计算过的结果；

通过一个无限循环以及switch--case的嵌套进行乘除操作；

**算法3.1** double prim(bool)

作 用：在计算过程中返回数值信息，以及括号等高优先级，有赋值等特殊运算；

参 数：bool

返回值：double

计算过程：

首先判断功能为读取字符信息，还是返回数值信息等操作；

若为前者则调用get\_token（）获取下一字符信息；

若为后者则通过传入的token\_type类型的变量判断进行什么操作；

**算法4.1** double error(string)

作 用：在计算过程出现错误时进行报错操作；

参 数：string

返回值：double

计算过程：

对传入的string信息判断发生什么错误并输出；

# 测试用例与测试结果

## 改造桌面计算器

|  |  |  |
| --- | --- | --- |
| 序号 | 测试数据 | 打印结果 |
| 1 | 2\*3+2/5;  -2+3\*3+2/5;  2/0;  3\*e; | 第1行结果：6.4  第2行结果：7.4  第3行发生除数为0错误  第4行结果：8.13 |
| 2 | A=1+3;  2+A;  3&3;  2\*p;  3+C; | 第1行结果：4  第2行结果：6  第3行发生非法输入错误  第4行结果：6.28  第5行有未赋值的字符存在 |

# 实验总结

初步理解如何将项目进行切分，来划分出不同的头文件以及.cpp程序。明白了在布局时结构体，枚举类型等应当放入对应的头文件中，而不是在头文件中用extern声明并在cpp文件中定义。对于数据流以及文件流有了新的认识，输出流一旦输出过了，则输出流中就不会在包含输出过的东西，所以一旦输出结束，输出流就相当于空了。更深入的理解了通过命令行参数来读取文件，处理文件信息等具体操作。通过本次实验自己的编程能力得到了长足的提升。