**表达式计算实验报告**

班级：计科201 实验日期：10.24

姓名1：郭奋志 学号：20401010124

姓名2：胡大维 学号：20401010122

姓名3：郭飞 学号：20401010118

**一、实验目的**

 1、使用基本操作实现线性表的具体操作；

2、掌握文件操作的应用；

3、提高对链接存储结构数据结构的理解，逐步培养解决实际问题的编程能力。

**二、实验环境**

 一台使用visualc++6.0/CFree的计算机。

本实验共4学时。

**三、实验内容**

 设计一个自动计算器，描述如下：

（1） 需要计算的表达式存储在TXT文件中；

（2） 文本中的每一行都是一个表达式；

（3） 表达式包括操作数、加法、减法、乘法、除法和括号；

例如：（34-72.3）\*54.7-82.4

（4） “自动计算器”根据输入文件名计算文本文件中的每个表达式，并将每个结果存储到名为[原始文件名]\u new.txt的文件中。保存记录时最好覆盖该文件。文件中的每一行应采用以下格式编写：

例如：原始文件是A1.txt

计算出的输出文件为：A1\_out.txt

A1\_out.txt文本中的结果形式如下：

               （34-72.3）\*54.7-82.4 = -2177.41

如果计算结果为小数，则应保留小数点后4位。

（5） 计算完成后，应生成统计文件。其内容如下：

执行时间：xxxx xx xx hh:mm:ss

表达式的总数为：XXX

正确表达式的数目为：XXX

错误表达式的数量为：XXX

统计文件命名规则：[原始文件名]\u log.txt，文件以附加写入模式写入。

例如：如果原始文件名为A1.txt文件，则统计文档应为：A1\_log.txt

**四、重要的数据结构**

栈

**五、实施分析**

**算符分析：**

表达式包括操作数和运算符，需要用operandStack和operandStack分别储存操作数和操作符。操作数和操作符在处理之前要先压入栈。当运算符被处理时，首先从operatorStack弹出，用operandStack栈顶的两个数进行计算，计算结果压进operandStack中。计算表达式有两个步骤：

步骤一：

程序从走到右扫描表达式，抽取操作数和运算符、（）和[]。

1)如果抽取到的是操作数，则压进operatorStack中。

2）如果抽取的是“+”或“-”运算符，则处理 operandStack栈顶比“+”、“-”号优先级更高或相等的运算符“+”“-”“\*”“/”，最后把抽取到的运算符压进operandStack栈中。

3）如果抽取到的是“\*”或“/”运算符，则处理operatorStack栈顶比“\*”“/”运算符优先级更高的运算符（“\*”“/”），最后把抽取到的运算符压进operandStack中。

4）如果抽取到的是“（”或“[”，则直接压入operatorStack栈中。

5）如果抽取到的是“）”或“]”，则处理operandStack栈顶的所有操作符，直到遇到符号“）”或“]”。

步骤2：

重复以上操作，直到清空operandStack栈中的所有运算符。

**函数分析：**

1. 表达式中可能有一些空白符，需要过滤得到没有空白符的表达式。
2. 表达式是由数字字符、运算符字符、“【”、“】”、“（”、“）”、“.”其它不规范的字符组成。需要提取double型字符串并将其转化为double型数字压入operandStack中，将其他字符压入operatorStack中。
3. 在过滤的基础下判断表达式格式的真确。

4）将表达式算发包装成一个函数。

**六、调试问题分析**

1）在判断表达式正确时，不能判断一的带有“+”或“-”double型数字的表达式正确。

解决办法：对表达式开头和结尾的的字符进行了更加全面的条件检测，和在if-else语句中加入了正数和负数的判断条件。

**七、总结**

我们小组在三个课题中选中计算表表达式这个课题，我们做了充分的考量。计算表达式课题所要应用的知识点栈我们刚刚学过，而又不与上个学期的课题冲突。我们小组通过对课题的讨论分析，将程序拆解封装成几个函数，并赋之以实现，从中我们对程序设计的核心思想——函数抽象和逐步实现有了更深层次的了解。对数据结构中栈的运用更加得心应手。当然。小组形式的作业也让我们体会到了程序员间的分工合作的高效性，更为我们将来的程序员工作打下了夯实的基础。

**八、成员**

|  |  |  |
| --- | --- | --- |
| **分组** | | |
| **成员名称** | **完成的工作** | **完成情况** |
| **郭奋志** | **算法分析和核心算法包装** | **圆满完成** |
| **胡大维** | **算法分析和辅助函数包装** | **圆满完成** |
| **郭飞** | **算法分析和报告撰写以及测试工作** | **圆满完成** |