

Data Structures and Algorithms Laboratory Projects Report

Linear List

常家奇

Date: 2019-11-11

Chapter 1: Introduction

Use the stack to complete the evaluation of arithmetic expressions. Enter an arithmetic expression from the keyboard or file, calculate the result and display it.

Chapter 2: Algorithm Specification

主要数据结构设计说明

- 采用了两个栈来进行运算(数字栈, 符号栈), 通过一边读取表达式一边压栈来计算表达式的值

系统设计思想

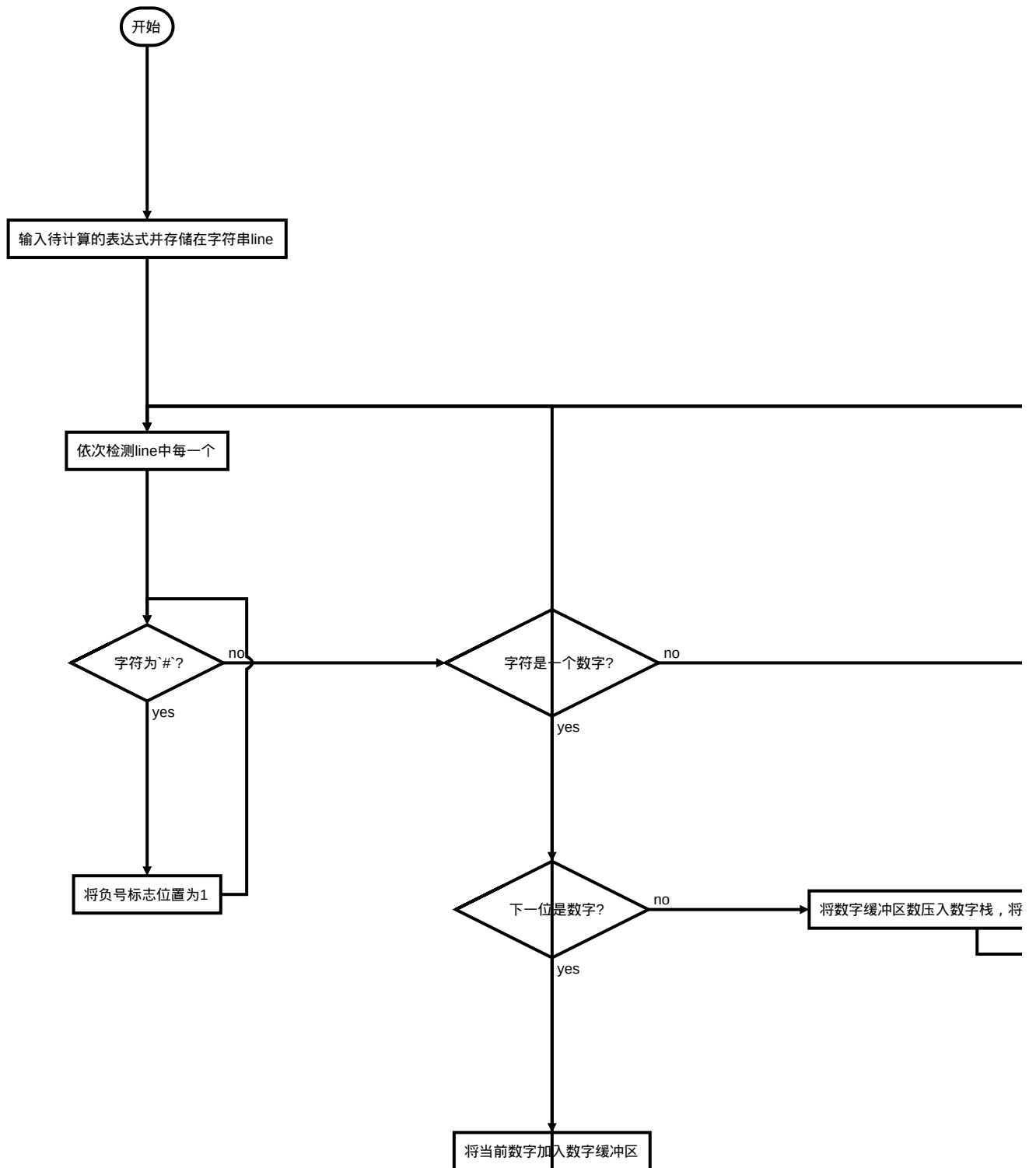
- 异常类
 - no_expr_exception 类
继承于基础异常类, 当输入的表达式非法时, 抛出此类。
 - divide_zero_exception 类
继承于基础异常类, 当运算过程中出现除数为0的情况时, 抛出此类。

- 流程图

首先, 输入一个待计算的中缀表达式, 将其读取到 string line中。检测表达式是否为退出表达式, 如果是, 则退出程序, 如果不是, 执行下一步。

然后, 依次检测 line 中的每一个字符:

- 如果是 # (代表负数), 则将负数标志位置为1, 重复此步。
- 如果是一个数字, 那么将其加入到数字缓冲区中同时如果这个字符的下一位不是数字那么将数字缓冲区中数压入数字栈 num_stack 中。
- 否则:
 - 如果字符为 (, 将其压入运算符栈 sys_stack 中。
 - 如果字符为):
 - 当符号栈顶元素不为 (时:
 - 如果 符号栈元素个数小于1 或 数字栈元素个数小于2, 抛出表达式错误异常。
 - 否则, 弹出符号栈栈顶元素和数字栈栈顶元素2个, 进行运算, 结果压入数字栈栈顶中。
 - 当符号栈顶元素为 (时, 弹栈。
 - 如果 数字栈为空 或 符号栈栈顶元素为 ' (' 或 符号栈栈顶符号优先级小于当前字符优先级, 将字符压入到符号栈。
 - 当符号栈不为空且栈顶符号优先级大于等于当前字符优先级时, 取出符号栈栈顶符号与数字栈栈顶两个元素进行运算:
 - 如果取出的符号为除且除数为0, 抛出除数为0异常。
 - 否则, 将运算结果压入数字栈。
- 输出数字栈栈顶元素, 计算完毕。
- 回到第一步。



Chapter 3: Testing Results

input	ouput	pass
1+2+3+4	10	yes
1-2*3+4	-1	yes

input	ouput	pass
(1+3)*3+2	14	yes
(#1+2)*5+1	6	yes
((2)-1)+1	error	yes
(4/2+1)*2	6	yes
((#1)+1))	error	failed

Chapter 4: Analysis and Comments

- 算法分析
算法使用了**两个栈**，采用**一边读一边处理**的方式，时间复杂度和空间复杂度均为 $O(n)$
- 算法特色
算法没有采用传统将表达式转换成后缀表达式再进行处理，而是直接一边读一边处理，优化了运行速度。
- 不足
精心设计的样例会造成程序崩溃。判断其逻辑复杂，故没有实现。(Ex:((#1)+1)))

Declaration

We hereby declare that all the work done in this project titled "Linear List" is of our independent effort as a group.

Duty Assignments:

Programmer: 常家奇

Tester: 常家奇

Report Writer: 常家奇