LL(1)语法分析设计原理与实现

**16281208**

**熊超**

**2018.5.31**

目录

[LL(1)语法分析设计原理与实现 1](#_Toc10224500)

[1.实验目的 3](#_Toc10224501)

[2.设计要求 3](#_Toc10224502)

[3.实验分析 3](#_Toc10224503)

[3.1定义Vt，Vn，产生式集合 3](#_Toc10224504)

[3.2 求出first集、follow集 4](#_Toc10224505)

[3.3构造LL1分析表 5](#_Toc10224506)

[3.4 LL1分析 5](#_Toc10224507)

[4.实验测试 6](#_Toc10224508)

[5.实验心得与体会 8](#_Toc10224509)

## 1.实验目的

实现LL(1)分析中控制程序（表驱动程序）；完成以下描述赋值语句的LL(1)文法的LL(1)分析过程。

G[A]:A→V=E

E→TE′

E′→ATE′|ε

T→FT′

T′→MFT′|ε

F→ (E)|i

A→+|-

M→\*|/

V→i

## 2.设计要求

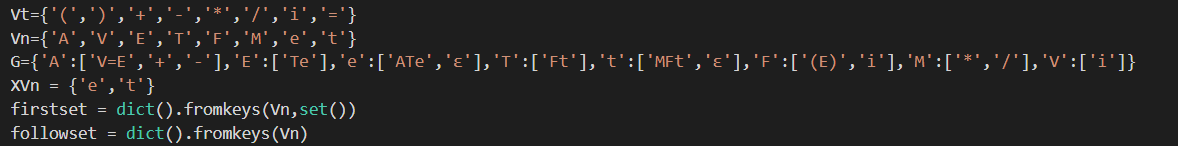
（1）输入串应是词法分析的输出二元式序列，即某算术表达式“专题1”的输出结果。输出为输入串是否为该文法定义的算术表达式的判断结果；

（2）LL(1)分析过程应能发现输入串出错；

（3）设计两个测试用例（尽可能完备，正确和出错），并给出测试结果；

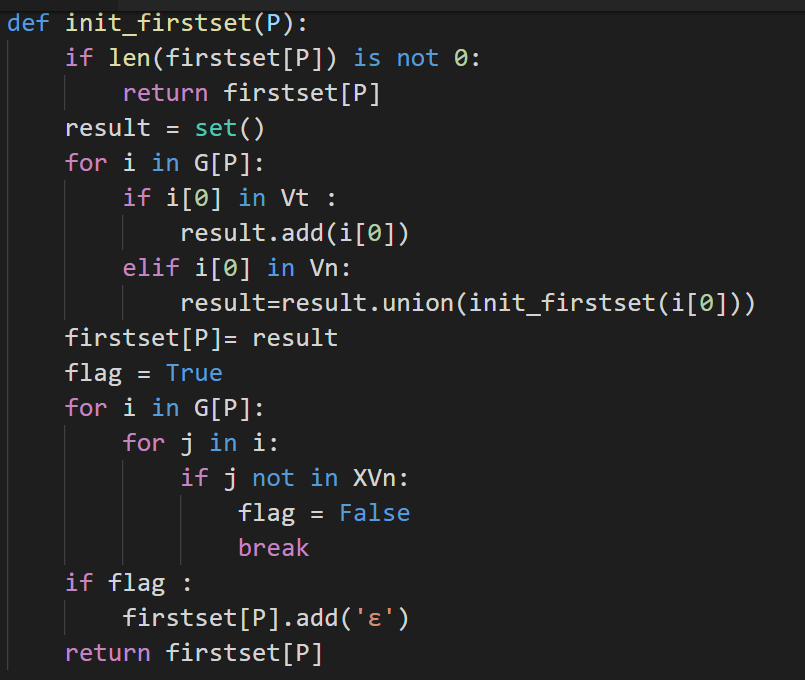
## 3.实验分析

### 3.1定义Vt，Vn，产生式集合

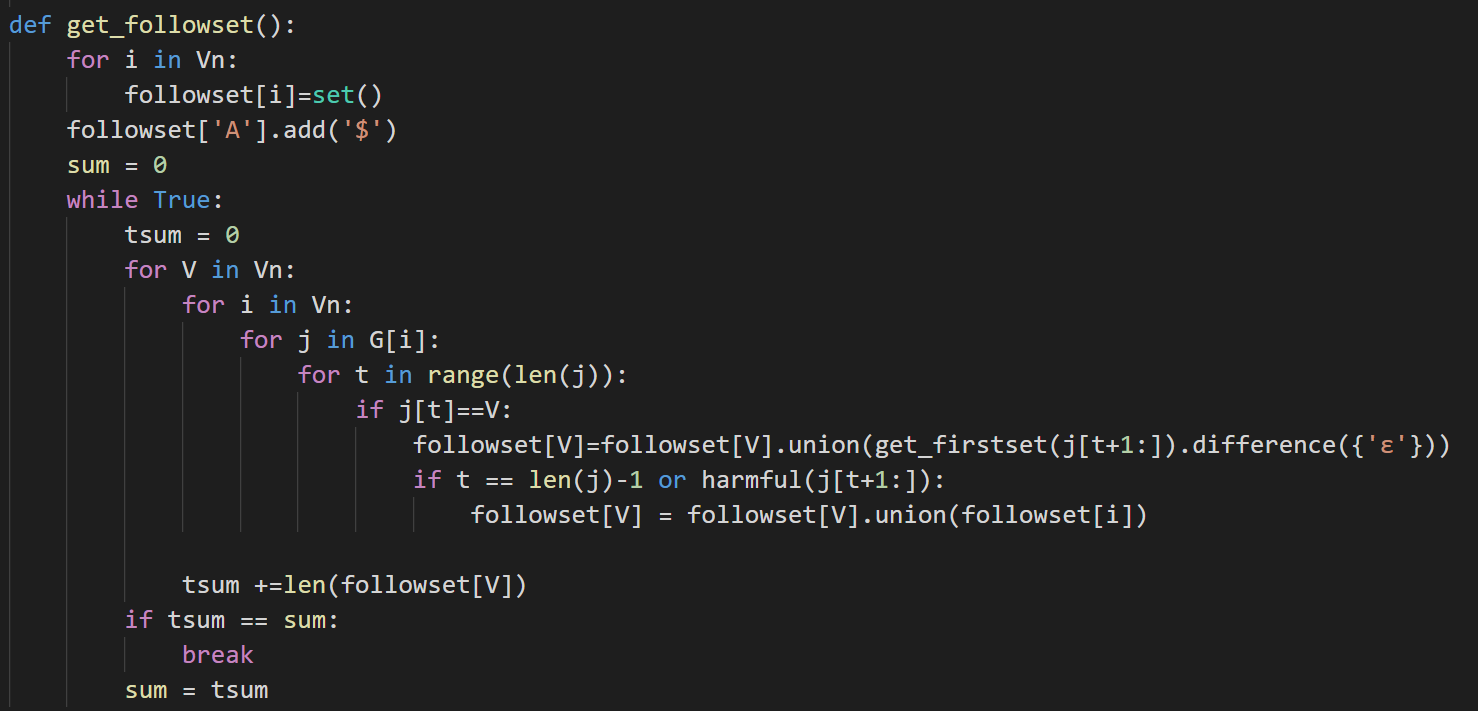


因为产生式较少，人工把能推导出空的产生式放入XVn，为方便归约，把E‘、T’改写为E,T单字符形式

### 3.2 求出first集、follow集

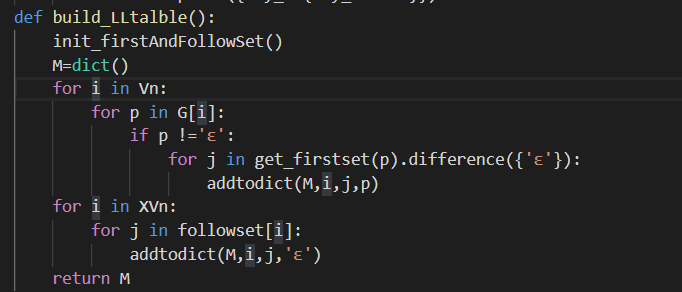


按照基本算法实现first集求解，因为涉及递归，所以那firstset减小递归层数



按照求解算法，整个followset都不再增大时，退出

### 3.3构造LL1分析表

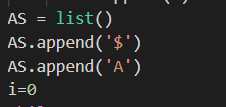


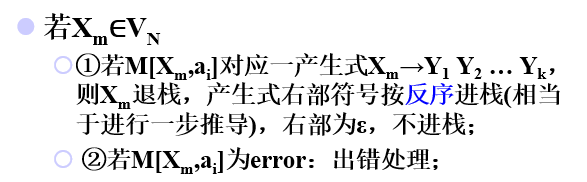
M为一个二维字典M[i][j]=p i为非终结符，j为i->p的first集中的元素,p为产生式右部

之后为可以推出空的终结符i和在其follow集的元素j添加M[i][j]= ε

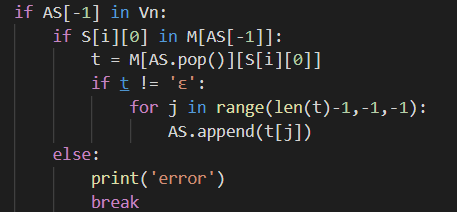
### 3.4 LL1分析

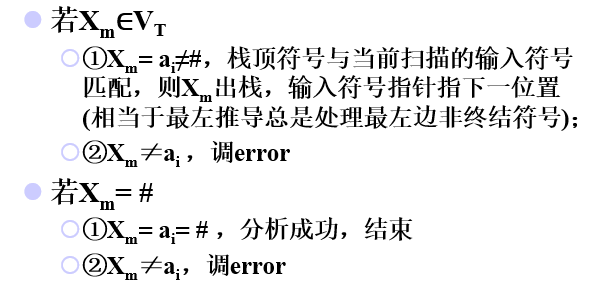
因为是自顶向下的分析，定义栈AS，将$和开始符号$入栈，i为输入指针

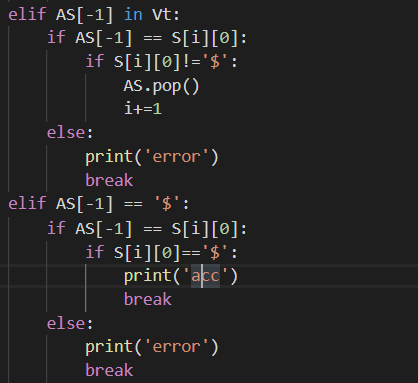




以下为上述算法的实现



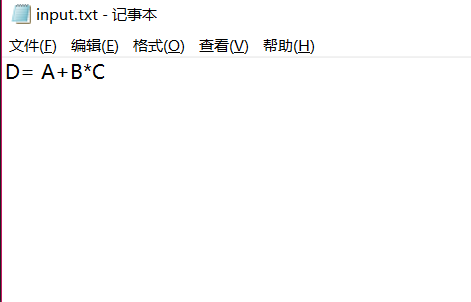




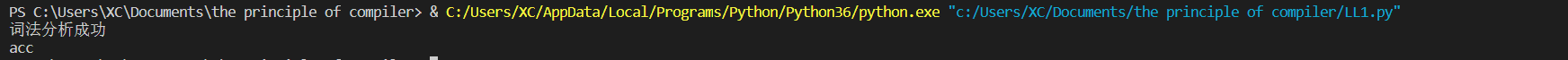
## 4.实验测试

**用例1：**

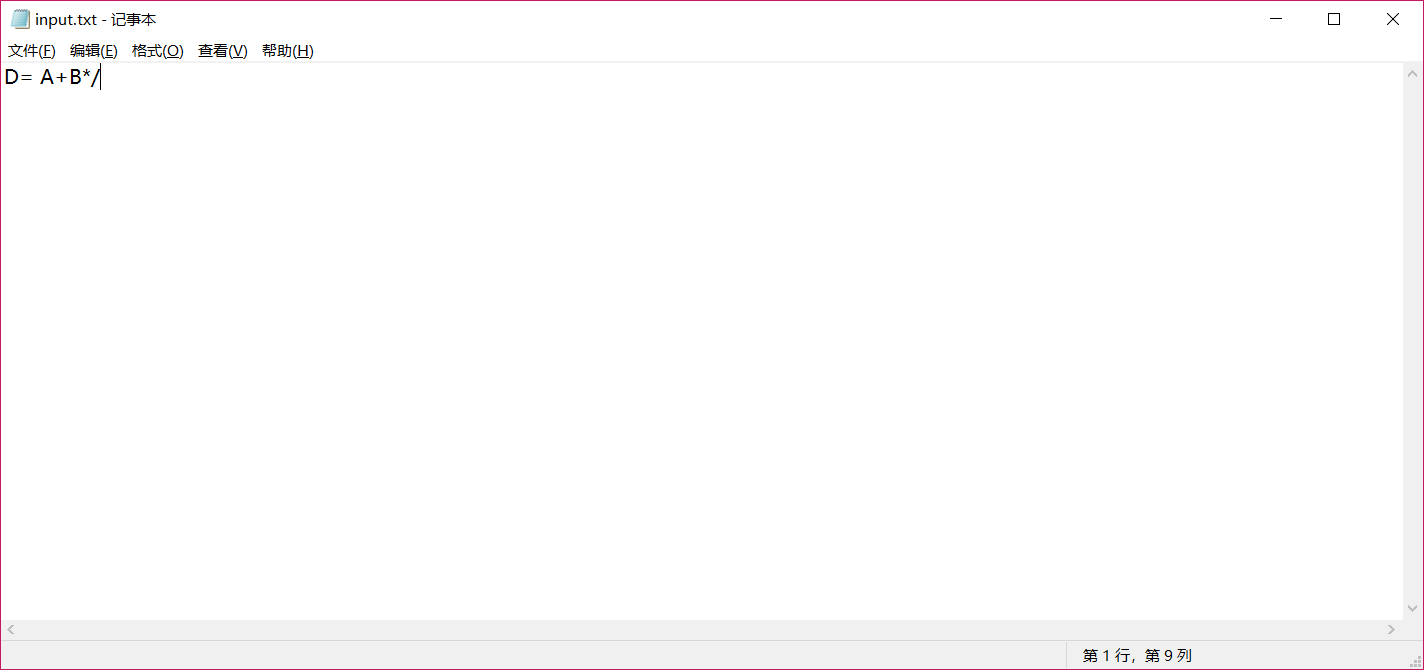
对于输入



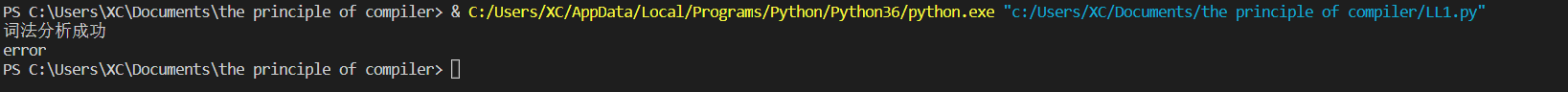
识别成功



**用例2：**



识别失败



## 5.实验心得与体会

通过本次专题实验，我更加深入的理解了LL1分析器的构造，锻炼了了自己的编程能力。