

实验二 自顶向下语法分析器的设计与实现

【实验目的】

根据所给文法编制自顶向下语法分析器（递归下降分析程序、LL(1)分析程序），以便对任意输入的符号串进行自顶向下的语法分析，加深对自顶向下的语法分析法的理解。

【实验要求】

1.输入要求

- (1)任意符号串，以#结束。如， $i+i*i\#$ 。
- (2)测试的输入串须包含文法中所有的终结符：**i, +, *, (,)**。
- (3)测试数据（符号串）至少有三组，其中一组是能被接受，另外两组不被接受。

2.输出要求

- (1)程序编制人信息：**姓名、学号、日期**。
- (2)符号串分析结果，即所分析的符号串是否为文法的句子。
- (3)输出推导的全过程。

【注意】

- 1.表达式中允许使用运算符（+、*）、分隔符（括号）、字符 i，结束符#。
- 2.如果遇到错误的表达式，应**输出错误提示信息（该信息越详细越好）**。
- 3.对学有余力的同学，测试用的表达式事先放在文本文件中，一行存放一个表达式，同时以分号分隔。同时将预期的输出结果写在另一个文本文件中，以便和输出进行对照。还可以根据推导的过程，建立一棵语法树。

【实验内容】

用 LL(1)分析法编程实现对下列文法 $G[E]$ 进行语法分析的程序，并对任意输入的符号串进行语法分析测试，记录测试结果。

- (1) $E \rightarrow TG$
- (2) $G \rightarrow +TG$
- (3) $G \rightarrow \varepsilon$
- (4) $T \rightarrow FS$
- (5) $S \rightarrow *FS$
- (6) $S \rightarrow \varepsilon$
- (7) $F \rightarrow (E)$
- (8) $F \rightarrow i$

【实验步骤】

1.阅读教材有关章节，为实验做好知识准备。

应用自顶向下分析法的前提是，文法应该是 LL(1)文法。因此，对所给文法应先进行以下改造：消除二义性、消除左递归、提取左共因子，判断是否为 LL(1) 文法。

2.明确编程方案。

实验设计思想：利用 LL(1) 控制程序根据显示栈栈顶内容、向前看符号以及 LL(1) 分析表，对输入符号串自上而下的分析。

实验算法如图 1 所示。

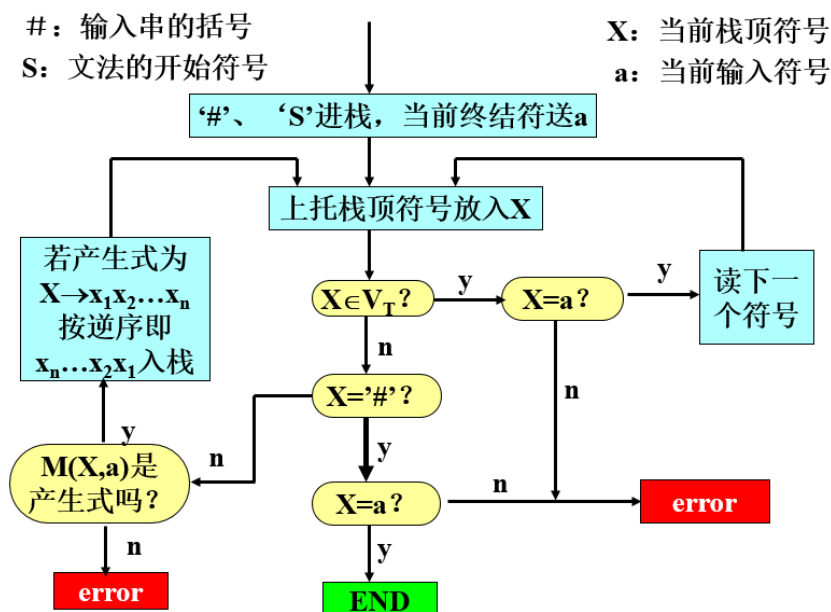


图 1 LL(1) 分析程序流程图

3.设计出模块结构、测试数据，初步编制好程序。

(1) 模块设计：将程序分成合理的多个模块（函数），每个模块做具体的同一事情。程序的模块结构可分成如下几部分。

- 定义部分：定义常量、变量、数据结构。
- 初始化：初始化变量空间（包括堆栈、结构体、数组、临时变量等）。

对于 LL(1)分析器，还要设立 LL(1)分析表。

- 控制部分：从键盘输入一个表达式符号串。利用 LL(1)分析算法进行表达式处理：流程控制按图 2 实现，并对表达式符号串进行分析，输出分析结果，如果遇到错误则显示错误信息。

(2) 写出（画出）设计方案：模块关系简图、流程图、全局变量、函数接口等。

(3) 编程时注意编程风格：空行的使用、注释的使用、缩进的使用等。

4.编辑源代码并调试，发现错误、再修改完善至调试通过。

【实验报告】

- 1.实验中使用的主要模块的算法流程图。
- 2.所实现的语法分析器的关键源代码。
- 3.实验结果截图打印粘贴在实验报告中。
- 4.记录实验过程中所出现的问题及解决方案