**国家人才培养模式创新实验区、云南省教学团队、云南大学软件学院教育创新基金联合资助项目**

**实验一**

**实现词法分析器**

姓 名： 何志颖

学 号： 20151120237

专 业： 软件工程

任 课 教 师 ： 柳青

编 制 时 间 ： 2018年6月26日

成 绩：

云南大学软件学院

2018年6月制表

目录

[1、 实验目标 3](#_Toc54625727)

[2、 实现方案 4](#_Toc1022267402)

[2.1 本实验待分析的单词token： 4](#_Toc399540075)

[2.2 Token种别表： 5](#_Toc1369221236)

[3、 实现步骤 7](#_Toc140210619)

[4、 结构算法 8](#_Toc888055336)

[5、 测试用例 10](#_Toc55002427)

[6、 运行结果 10](#_Toc215487236)

[7、 实验体会 11](#_Toc651233494)

1. **实验目标**

设计并实现一个PL/0语言的词法分析器， 对读入的PL/0源程序，输出相应的Token二元式序列。

词法分析是编译的第一个阶段，它的主要任务是从左至右逐个字符地对源程序进行扫描，产生一个个单词序列，用以语法分析。

一般情况，常将词法分析程序设计成一个子程序。便于在总控程序中来调用该词法分析程序。本次实验就是设计这样一个词法分析器，可作为一个子程序被编译器总控程序调用。

1. **实现方案**

单词符号（TOKEN) 是一个程序设计语言的基本语法符号。程序设计语言的单词符号一般可分成下列5种：

1．基本字，也称关键字，如PL/0语言中的begin，end，if，while和var等。

2．标识符，用来表示各种名字，如常量名、变量名和过程名等。

3．常数，各种类型的常数，如25，3.1415，TRUE和"ABC"等。

4．运算符，如+，\*，<= 等。

5．界符，如逗点，分号，括号等。

**2.1 本实验待分析的单词token：**

1. 关键词

begin, const, end, odd, read, var, write, call, do, if, proc, then, while

1. 运算符和界符

+ \* ( = < > <= >= . ; - / ) , # :=

1. 标识符和字面量

变量i/j/k,数值1/2/3等

1. 空白符

空格、换行、制表符等

**2.2 Token种别表：**

|  |  |  |  |
| --- | --- | --- | --- |
| **单词符号** | **种别码** | **单词符号** | **种别码** |
| **Begin** | **Beginsym** | **(** | **Lparen** |
| **Const** | **Constsym** | **=** | **Equal** |
| **End** | **Endsym** | **<** | **Less\_than** |
| **Odd** | **Oddsym** | **>** | **Greater\_than** |
| **Read** | **Readsym** | **<=** | **Less\_equal** |
| **Var** | **Varsym** | **>=** | **Greater\_equal** |
| **Write** | **Writesym** | **.** | **Period** |
| **Call** | **Callsym** | **;** | **Semicolon** |
| **Do** | **Dosym** | **-** | **Minus** |
| **If** | **Ifsym** | **/** | **Divide** |
| **Proc** | **Procsym** | **)** | **Rparen** |
| **Then** | **Thensym** | **,** | **Comma** |
| **While** | **Whilesym** | **#** | **Neq** |
| **+** | **Plus** | **:=** | **becomes** |
| **\*** | **Times** |  |  |

**表2.2 Token对应的种别码**

1. **实现步骤**

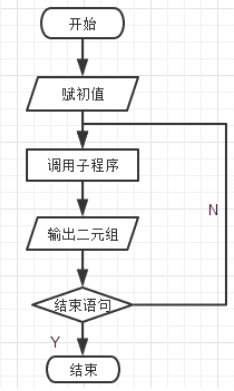


图3.1 主程序功能流程图

1. **结构算法**

PL/O语言的编译程序，是用高级语言PASCAL语言书写的。整个编译过程是由一些嵌套及并列的过程或函数完成。词法分析程序是独立的过程GETSYM完成，供语法分析读单词时使用。

语法分析是由过程BLOCK完成。采用自顶向下的递归子程序法。所产生的目标程序为假象栈式计算机的汇编语言。对目标程序的执行是由PASCAL语言书写的解释程序进行的。因此 PL/O语言可以在配备PASCAL语言的任何机器上实现。

由于PL/O语言编译程序是适合教学用的实例，它的数据类型只有整形数，数据运算只有四则运算。语句有复制语句、条件语句、While型循环语句、输入、输出语句和不带参数允许递归调用过程语句及复合语句。

词法分析程序GETSYM的功能包括：

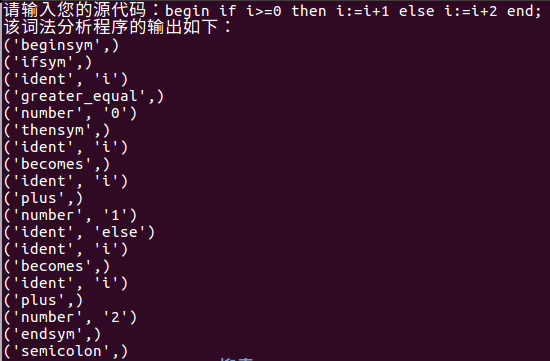
1. 滤空格,空格在词法分析时是一种不可缺少的界符,而在语法分析时是无用的,所以必须滤掉。
2. 识别保留子：设有一张保留字表。对每个字母打头的字母、数字字符串要查此表。若查着则为保留字，对应的类别放在SYM中。如IF对应值为THENSYM。
3. 识别标识符：对用户定义的标识符将IDENT放在SYM中，标识符本身的值放在ID中。
4. 拼数：当所取单词是数字时，将树的类别NUMBER放在SYM中，数值本身的值放在NUM中。
5. 拼复合词：对两个字符组成的算符如：>=、：=、<= 等单词，识别后将类别送SYM中。
6. 打印源程序：为边读入字符边打印。打印每个单词

的识别类别（如果是标识符或数字应该给出其值即id 和 num中的值。图2所示即为给定PL0源程序的一个可能输出）。

1. **测试用例**

**begin if i>=0 then i:=i+1 else i:=i+2 end;**

1. **运行结果**



1. **实验体会**

本次实验的目的在于实现一个简单的词法分析器，通过他得到的单词符号为后面的语法分析作准备，词法分析是编译过程中的一个阶段，在语法分析前进行。词法分析作为一遍，可以简化设计，改进编译效率，增加编译系统的可移植性。也可以和语法分析结合在一起作为一遍，由语法分析程序调用词法分析程序来获得当前单词提供给语法分析使用。

通过实验一，我对词法分析的主要任务和思想有了一定的认识，同时也让我了解了如何设计、编制并调试词法分析程序，并加深了我对词法分析器原理的理解。

在熟悉了直接构造词法分析器的方法和相关原理后，学会了使用编程语言编写词法分析器，并实现了一定的实际功能。在编译原理的指导下，顺利完成了此次实验，进一步加深了对理论原理的理解。