**PL/0 语言编译器分析实验报告**

**一、实验目的**

通过阅读与解析一个实际编译器（PL/0语言编译器）的源代码， 加深对编译阶段（包括词法分析、语法分析、语义分析、中间代 码生成等）和编译系统软件结构的理解，并达到提高学生学习兴趣的目的。

**二、实验要求**

(1)要求掌握基本的程序设计技巧（C语言）和阅读较大规模程序 源代码的能力；

(2)理解并掌握编译过程的逻辑阶段及各逻辑阶段的功能；

(3)要求能把握整个系统（PL/0语言编译器）的体系结构，各功能 模块的功能，各模块之间的接口；

(4)要求能总结出实现编译过程各逻辑阶段功能采用的具体算法与技术。

**三、实验步骤**

(1) 根据PL/0语言的语法图，理解PL/0语言各级语法单位的结构，掌握PL/0语言合法程序的结构；

(2)从总体上分析整个系统的体系结构、各功能模块的功能、各模块之间的调用关系、各模块之间的接口；

(3)详细分析各子程序和函数的代码结构、程序流程、采用的主要算法及实现的功能；

(4)撰写分析报告，主要内容包括系统结构框图、模块接口、主要算法、各模块程序流程图等。

**四、报告内容**

　 pl/0语言是pascal语言的一个子集，我们这里分析的pl/0的编译程序包括了对pl/0语言源程序进行分析处理、编译生成类pcode代码，并在虚拟机上解释运行生成的类pcode代码的功能。

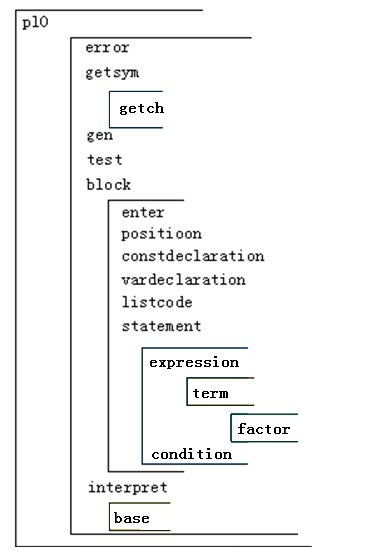
pl/0语言编译程序采用以语法分析为核心、一遍扫描的编译方法。词法分析和代码生成作为独立的子程序供语法分析程序调用。语法分析的同时，提供了出错报告和出错恢复的功能。在源程序没有错误编译通过的情况下，调用类pcode解释程序解释执行生成的类pcode代码。

**PL/0语言文法的EBNF表示**

EBNF表示的符号说明。

|  |  |
| --- | --- |
| 〈 〉 | 用左右尖括号括起来的中文字表示语法构造成分，或称语法单位，为非终结符。 |
| ∷= | 该符号的左部由右部定义，可读作“定义为”。 |
| | | 表示“或”，为左部可由多个右部定义。 |
| { } | 花括号表示其内的语法成分可以重复。在不加上下界时可重复0到任意次数，有上下界时为可重复次数的限制。 |

**PL/0编译程序过程与函数定义层次结构图**

****

**PL/0的解释执行结构**

|  |
| --- |
| PL/0语言解释执行程序 |

|  |
| --- |
| PL/0语言目标程序 |

|  |
| --- |
| 输入数据 |

|  |
| --- |
| 输出数据 |

**PL/0编译程序结构**

词法分析程序

语法语义分析程序

代码生成程序

表格管理程序

出错处理程序

PL/0源程序

目标程序

**编译程序总体流程图**

****

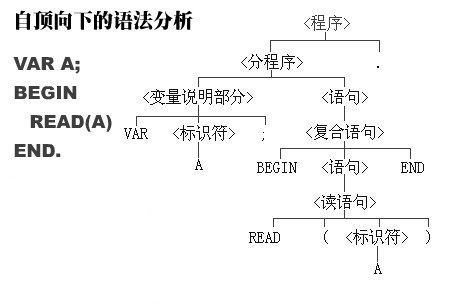
**PL/0编译程序的语法分析**

　　PL/0编译程序语法、语义分析是整个编译程序设计与实现的核心部分,要求学员努力学习掌握实现技术和方法。现分别说明语法分析实现的主要思想方法和语义分析的实现。

　　语法分析的任务是识别由词法分析给出的单词符号序列在结构上是否符合给定的文法规则。PL/0语言的文法规则已在2.1节中给出。本节将以语法图描述的语法形式为依据，给出语法分析过程的直观思想。

　　PL/0编译程序的语法分析采用了自顶向下的递归子程序法。

可用下面简单的PL/0程序为例构造其语法分析树



**语法调用关系图**

