

中山大学移动信息工程学院本科生实验报告

(2014-2015 学年春季学期)

课程名称：计算机语言及其处理实验

任课教师

已拒绝

年级&班级		已拒绝	专业（方向）	软件工程（移动信息工程）
学号			姓名	
完成日期	2015-05-24		Email	已拒绝 .cn

一、实验目的

1. 学习使用 JFlex 词法分析程序；
2. 比较不同词法分析程序的差异。

二、实验内容

1. 总结 Oberon-0 语言词汇表

类型	实例
保留字	MODULE, TYPE, PROCEDURE, RECORD, ARRAY, OF, DIV, MOD, OR, VAR, Read, Write, Writeln
关键字	INTEGER, BOOLEAN, WHILE, IF, DO, ELSIF, THEN, ELSE, CONST, BEGIN, END
运算符	+ - * & ~ < <= > >= # =
标识符	letter(letter digit)*
标点符号	. ; () : := []
常量	[1-9] ⁺ [0-9]* 0[0-7]*

关键字主要是数据类型和程序控制流的相关词，保留字是程序本身已经定义了该词的含义，用户不可以将其作为其他变量或是标识符。按这样的归类，在 Oberon-0 中便可轻松的分开保留字和关键字了。

2. 抽取 Oberon-0 语言的词法规则

Digit——> [0-9]

Letter——> [a-z A-Z]

Integer = [1-9]⁺[0-9]* | 0[0-7]*

Number——> integer^{*}

Identifier——> letter(letter|digit)*

Comment——> (* ~ *)

Oberon-0 与 C/C++、Java 相比的异同点

异：①在 Oberon-0 中标识符不可以含有下划线

②数字只能是整数

③在取模，除法，等于，不等于，或，非等存在不同

	取模	除法	等于	不等于	或	非
C/C++	%	/	==	!=		!
Oberon-0	MOD	DIV	=	#	OR	~

④注释不同，C/C++是 /* */或//，而 Oberon-0 为 (**)

3. 生成 Oberon-0 语言的词法分析程序

4. 讨论不同词法分析程序生成工具的差异

在不同的 lex 工具中都是 “%%” 来划分用户代码、选项与声明、词法规则三个部分的，

如下

用户代码

%%

选项与声明

%%

词法规则

在用户代码部分，都是直接将这部分拷贝到此法分析器中。

选项与声明部分，GNU flex 生成的是 C 语言，而 JFlex 和 Jlex 生成的是 java 语言，由于 C 语言和 java 语言的不同，所以在词法规则方面是存在差异的。

词法规则部分，针对不同的语言有不同的词法规则，比如此实验中的 Oberon-0 语言的词法规则和 C/C++、Java 肯定是不相同的。

三、实验感想

在此实验中，了解了 JFlex、Jlex、GNU flex 三种不同的词法分析工具，对 JFlex 重点理解，了解了其中的不同元素的意义及作用，学会了编写 JFlex 的源文件，对词法生成的过程有了更深的理解。