



華南師範大學

## 本科学生实验（实践）报告

院 系：计 算 机 学 院

实验课程：编译原理课程项目

实验项目：Oberon-0 程序流程图生成器

指导老师：王欣明

开课时间：2018 ~ 2019年度第二学期

专 业：网络工程

班 级：七班

学 生：曹博凯 林瑞儿 卓伟洲

学 号：20162382018

20162180014      20162180063

华南师范大学教务处

华南师范大学实验报告

学生姓名：曹博凯 学号：20162382018

专业：网络工程 年级班级：2016级七班

课程名称：编译原理课程项目 实验项目：Oberon-0 程序流程图生成器

实验时间 2019 年 6 月

实验指导老师 王欣明 实验评分

## 1、实验名称：

子实验三：生成语法分析程序（JavaCUP）

## 2、实验主要硬件软件环境：

Windows10操作系统

## 三、实验任务：

子实验三要求你下载一个语法分析程序自动生成工具 JavaCUP，利用该工具自动产生一个Oberon-0 语言的语法分析和语法制导翻译程序。

### 1. 下载自动生成工具 JavaCUP

实验三选用由美国卡内基·梅隆大学的 Scott E. Hudson 开发的一个语法分析程序自动生成工具 JavaCUP，它是一个 LALR Parser Generator。JavaCUP 是一个类似 Unix 平台上 yacc 程序的开源（Open Source）软件工具，遵循 GNU General Public License（GPL）。JavaCUP 本身采用 Java 编写，并且生成 Java 语言的分析程序源代码。该软件工具经由美国普林斯顿大学计算机科学系 Andrew W. Appel 教授指导 Frank Flannery 等人改进，目前由 C. Scott Ananian 负责维护。

从 <http://www2.cs.tum.edu/projects/cup/> 可下载该软件工具的最新版本。该网站已包含了你在实验中所需的各类资源，包括该工具的 Java 源代码、已编译生成的字节码、简明的用户手册、以及一个简单的命令行计算器例子等。

### 2. 配置和试用 JavaCUP

成功下载并配置后，试运行 JavaCUP 附带的输入源文件例子（一个基于命令行的简单计算器应用），以保证你正确安装并配置了 JavaCUP。

### 3. 生成 Oberon-0 语法分析和语法制导翻译程序

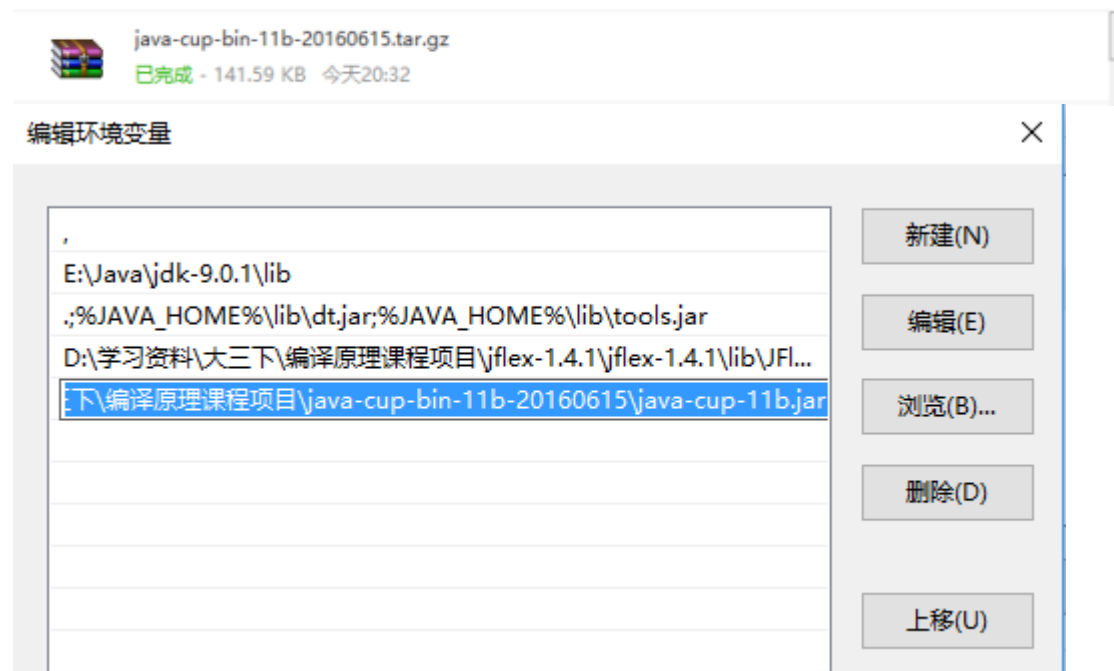
仔细阅读 JavaCUP 使用手册，根据 Oberon-0 语言的 BNF 定义编写一个 JavaCUP 输入源文件。根据你的 JavaCUP 输入源文件生成的语法分析程序须完成以下功能：

1、对于一个存在词法、语法或语义错误的 Oberon-0 源程序，必须至少指出一处错误，并判断错误的类别及产生错误的位置（错误产生的位置定位允许有偏差），并以相应的异常对象向客户程序报告找出的错误的类别。是否支持其他功能取决于你的时间、精力与能力，譬如你可尝试从错误中恢复并继续执行语法分析，也可生成设计图时立即中止程序的执行。在错误检查与错误恢复（指找到错误后继续执行分析过程的能力）方面做得优秀的实验可获得更高的评分。

2、对于一个词法、语法和语义完全正确的 Oberon-0 源程序，自动绘制出不同模块中每个函数（也叫过程）的流程图。本文档第 6 节部分详细介绍了流程图的定义和画图的 API。你在生成 Oberon-0 语言的语法分析程序时，可以直接使用子实验二由 JFlex 生成的 Oberon-0 语言的词法分析程序。

## 四、实验实现：

### 1. 下载自动生成工具 JavaCUP



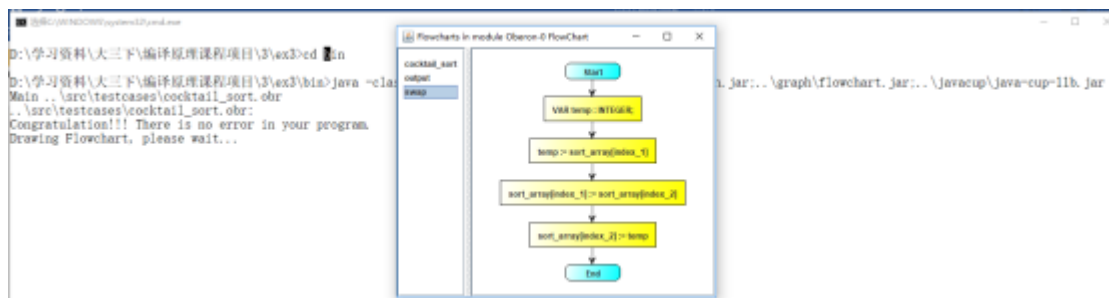
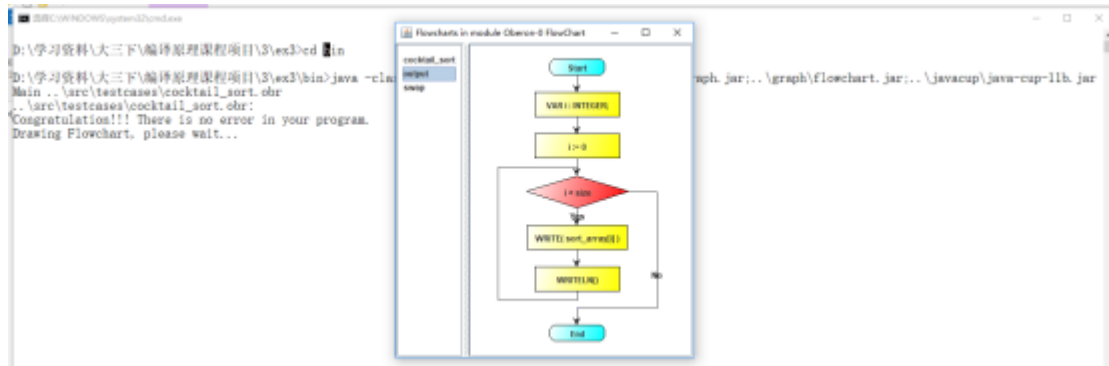
### 2. 配置和试用 JavaCUP

```
D:\学习资料\大三下\编译原理课程项目\jflex-1.4.1\jflex-1.4.1\examples\cup>java -jar java-cup-11b.jar -interface -parser Parser ycalc.cup
----- CUP v0.11b 20160615 (GIT 4ac7450) Parser Generation Summary -----
0 errors and 0 warnings
11 terminals, 6 non-terminals, and 14 productions declared,
producing 23 unique parse states.
0 terminals declared but not used.
0 non-terminals declared but not used.
0 productions never reduced.
0 conflicts detected (0 expected).
Code written to "Parser.java", and "sym.java".
----- (CUP v0.11b 20160615 (GIT 4ac7450))
```

```
2 + 4 = 6
5 * ( 6 - 3 ) + 1 = 16
6 / 3 * 5 + 20 = 30
4 * 76 / 31 = 9
1 - 1 - 1 = -1
```

### 3. 生成 Oberon-0 语法分析和语法制导翻译程序





## 五. 实验小结:

- (1) 本次实验使得对JavaCUP有了初步的了解
- (2) 对语法分析生成过程有了进一步的理解