

华东师范大学软件学院实验报告

实验课程：编译原理实践

年级：2013 级 1 班 实验成绩：

实验名称：期末项目

姓名： 储德明

实验编号：No.1

学号：10132510126 实验日期：2016. 1. 21

指导教师：张敏

组号： 实验时间：

一、实验目的

开发词法分析器，语法分析器，并且有一定的界面功能。

二、实验内容与实验步骤

1. 学习 Lex, Yacc 工具
2. 编码，调试
3. 与界面结合起来再次调试。

三、实验环境

Linux Ubuntu 14.04, VIM

四、实验过程与分析

1. 通过《Lex 与 Yacc 2nd Editon》中文版了解了工具的基本使用（细读并实践了所有前几章节的源码，粗略了解了后半部分），通过 vimtutor 了解了 vim 的基本使用方法。
2. 建立词法分析器，同时阅读网上 blog 及 vimdoc 完成了 vim 下的文件高亮，初期的配置文件放在了 vim 目录下，之后由于 vim 库更新会影响到配置文件于是改到.vim 下。
3. 建立底层的数据结构，为了方便选择了使用 map（STL 内置的平衡二叉搜索树），并建立了工具头文件及实现。写好 Makefile。
4. 建立语法分析器，初期将所有的空格都放到了产生式中，直到构建表达式的时候产生了无法消去的冲突，于是把所有空格去掉成功解决了问题。语法中部分用*来表示的语法实际上无法直接使用，我用了类似自动机理论中学习的乔姆斯基范式的方式来解决这个问题，基本上就是将 $a \rightarrow b * cde$ 转化为 $a \rightarrow blist \quad finishB \mid finishB, blist \quad finishB$ 中用类似方法继续搞，然后把语法中所有的空都消去了，基本不会产生冲突了，我认为这是很基本的东西，并没有怎么说，不消除这个歧义那肯定无法处理出语法的。
5. 建立错误的分析处理，首先想到了行号列号，行号在 Lex 中对应的选项可以直接获得，然后列号通过保存一个变量简单解决。通过思考总结了一些常见的错误，写成了错误处理的函数，初期全部都是 yyerror 重载的，函数变多之后反而变得不方便了，于是重新命名了一遍。由于产生式中大量错误是相同的终结符产生的，我选择了多放入一个非终结符，在非终结符中产生非终结符及错误，然后进行错误处理，省去了很多事情。

6. 写完把程序打包丢给做界面的人，收工。

五、实验总结

总的来说，本次实验虽然完成了但是并不令人满意，个人觉得想要协调这么多人队伍比较困难，而实际的编码工作基本是只有两三人完成的，同时我也觉得三个人的队伍做起项目来最为有效率。人多了之后容易凑不齐人，想要找外国人的时候，他们总是不出现，甚至连项目汇报的时候也是不在的。

我独立开发了几乎所有的底层处理部分，学习 Lex & Yacc 花了比较久的时间，然后在两三天内完成了所有语法词法部分的功能。在项目中学习到的东西很多都能够利用在理论课的考试中。=_=本来是想分工合作来着，结果一是他们不回我的信息，二是开发到后头很有意思然后很快就写完了，结果底层基本就被我包掉了，除了数据结构的实现部分让 ck 实现了。界面本来是预定给外国人做的，结果他们根本不出现，然后吴炎，ck 他们就去做了。

做这个项目非常有助于加深对于词法语法分析的理解，比硬生生实现底层的算法有意思多了。