

编译原理实验大纲

1 实验目的

设计并实现一个 SPL-语言（或者类 C）的编译系统（SPL-语法见 SPL Specs Vxx.doc），掌握编译原理的各个环节：词法分析、语法分析、语义分析、代码生成，以及实现所需的数据结构：语法树、符号表等。通过这样的实验，提高学生协作编程的能力，加深对编译技术的理解，编译原理是一门综合各个学科知识的课程，编译系统设计让学生在实践中综合理解计算机学科知识。

2 实验环境

Windows 系统环境或者 Linux 环境。

1、Linux 环境下的编译和运行

- (1) Linux 2.6 以上版本
- (2) GCC3.4 以上版本
- (3) Bison 2.2 以上版本
- (4) Flex 2.5.33 以上版本

发行版可以采用 Ubuntu, Gentoo, Fedora Core 等。

2、Windows 环境下的编译和运行

- (1) Visual Studio 12
- (2) Masm 6.0 以上版本
- (3) Bison+Flex 或者 ParseGenerator 4.0（Lex 和 Yacc 的集成开发包）

3 实验要求：

(1) 实验内容

完成 SPL 语言（或者类 C 语言）编译器，输入为符合 SPL 语言规范的文本，输出是 MIPS 指令代码或者汇编代码。SPL 语言规范见 SPL Specs Vxx.doc。

编译器包含词法分析、语法分析、语义分析、代码生成、代码优化、运行环境等阶段和环节。报告中要包括词法分析部分的正规表达式描述、数据结构、原理以及实现，语法分析部分的文法描述、数据结构、原理以及实现，语义分析的方法描述，中间代码的格式、数据结构描述以及中间代码生成的实现及部分优化（目标代码的生成）。

要明确组员分工情况。

(2) 如果生成语法树，要求结果能用可视的方法表示出来（图或者采用数据结构中学过的方法）

(3) 提交压缩包内的文件要求，压缩文件名为“学号 1 姓名 1-学号 2 姓名 2-学号 3 姓名 3.zip”

包括文件：

原代码文件：包括 lex 文件，yacc 文件以及其他程序等等
可执行文件

实验报告：文件名为“report_学号 1 姓名 1-学号 2 姓名 2-学号 3 姓名 3.doc(x)”。

不包括的文件：

编译中间文件。

(4) 实验报告建议格式 (不限于该格式)

报告封面 (见后一页)

序言 (包括对整个编译器的描述, 对所提交的各个文件的说明, 组员的分工)

第一章 词法分析 (正规表达式, 实现原理和方法)

第二章 语法分析 (上下文无关文法, 实现原理和方法)

第三章 语义分析 (实现方法)

第四章 优化考虑 (每个阶段的优化考虑)

第五章 代码生成 (所有语句的代码生成的处理)

第六章 测试案例 (每个语句成分的测试案例, 至少两个复杂语句组合后的测试案例)

4 评分标准

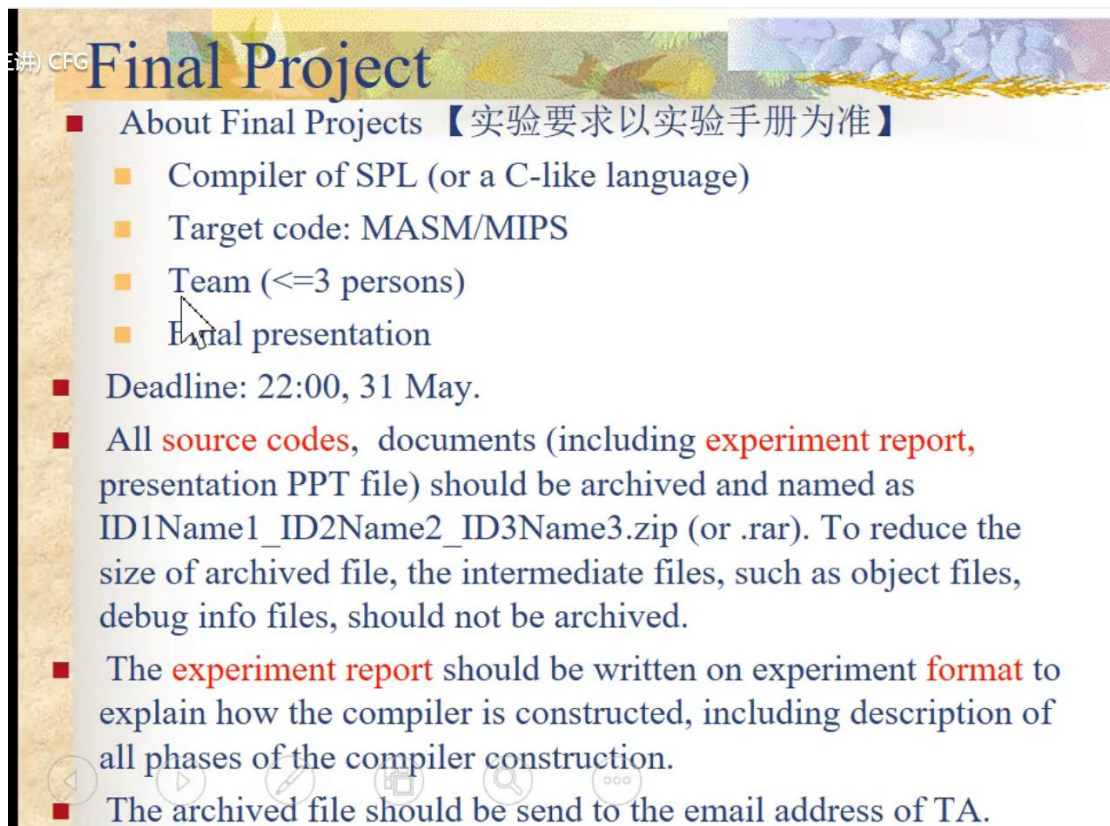
SPL-编译器实验具体的评分标准如下:

(1) 实验最后得分由两部分构成: 实验报告和文档的规范性(35%), 包括词法分析设计、语法分析设计、语义分析设计、运行环境设计、代码生成设计、符号表设计和其他设计文档; 实验的正确性和合理性(65%), 根据运行情况来判断实验的正确性和合理性, 包括词法分析、语法分析、语义分析和符号表、运行环境和代码生成等模块。

(2) 报告及时提交, 则根据报告的质量给相应得分; 未及时提交, 则在报告质量分级基础上降一级, 未提交报告或报告为抄袭, 相应的报告得分为“不及格”。

(3) 总体报告得分各小组成员相同, 在总体报告中应给出各成员负责的模块, 详细报告根据各人的任务单独给分。

(4) 若基本上未编写程序或程序纯属抄袭, 验收得分为“不及格”。



Final Project

- About Final Projects 【实验要求以实验手册为准】
 - Compiler of SPL (or a C-like language)
 - Target code: MASM/MIPS
 - Team (≤ 3 persons)
 - Final presentation
- Deadline: 22:00, 31 May.
- All source codes, documents (including experiment report, presentation PPT file) should be archived and named as ID1Name1_ID2Name2_ID3Name3.zip (or .rar). To reduce the size of archived file, the intermediate files, such as object files, debug info files, should not be archived.
- The experiment report should be written on experiment format to explain how the compiler is constructed, including description of all phases of the compiler construction.
- The archived file should be send to the email address of TA.

Points of Presentation

- Peoples, division of work
- Scanning. Using Lex?
- Parsing. Using Yacc?
- Semantic Analysis
 - Symbol table, type checking...
- Code generation
 - Use LLVM?
 - Target: MASM or MIPS
 - Runtime environment
- Optimization?
 - Constant folding/propagation, common expression elimination, register allocation...