

2025 年编译原理与技术实践课程实验报告

姓名: _____ 王海生 _____ 学号: _____ 10235101559 _____

课程名称: 编译原理与技术实践

指导教师: _____ 张敏 _____ 实验日期: _____ 2025 年 12 月 30 日

实践亮点

以下为本课程实践环节中的主要亮点（括号中为涉及到的实验的序号）：

1. 在四次编译器实验中，采用了**面向对象设计方法**，实现了词法分析器、语法分析器和翻译器的模块化封装与解耦。（1、2、3、4）
2. 在语法分析器实现中，采用了**自动化表生成方法**，实现了 FIRST/FOLLOW 集自动计算和 LL(1)/LR(1) 分析表自动构建，减少了人工推导错误。（2、3）
3. 在符号识别模块中，采用了**有限自动机 (DFA)**、**LL(1) 预测分析**和**LR(1) 移进-规约算法**，实现了从词法到语法的完整自动化分析功能。（1、2、3）
4. 在错误处理环节中，设计了**多层次错误恢复机制**（鲁棒性设计、同步恢复、恐慌模式、错误收集），实现了语法错误的精确定位、自动修复和继续解析功能。（1、2、3、4）
5. 在测试阶段，编写了**自动化测试脚本和测试用例生成器**，实现了批量测试用例的自动执行、结果比对验证和测试报告生成效果。（1、2、3、4）
6. 在数据存储模块中，采用了**STL 容器 (map、set)** 和**哈希表数据结构**，实现了 $O(1)$ 到 $O(\log n)$ 级别的符号查找、分析表查询和状态访问性能提升。（1、2、3、4）
7. 在词法分析阶段，采用了**最长匹配算法**和**扩展数字识别算法**，实现了多字符运算符、整数、浮点数、科学计数法的准确识别。（1）
8. 在错误定位环节，设计了**精确行号追踪机制**（非空行统计、行号传递），实现了词法、语法和语义错误的精确定位功能。（2、4）
9. 在项目管理中使用**Git 进行版本控制**，实现了多人协作开发。**GitHub 仓库**:
https://github.com/Hanson-Wang-chn/compiler_lab.git （1、2、3、4）
10. 系统整体采用**模块化设计**，便于后续功能扩展与维护。（1、2、3、4）

基础实验代码、补充实验代码、运行文档、补充说明文档，请见 GitHub 仓库。