电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2024091203018

姓 名 龚陈陈

（实验） 课程名称 编译原理-词法分析

理论教师 朱嘉静

实验教师 朱嘉静

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：龚陈陈 学号：2024091203018指导教师：朱嘉静**

**组长姓名：龚陈陈 组长学号：2024091203018 组内排序：0**

**实验地点：信软大楼400 实验时间：2025-10-31 14:30 -- 2025-10-31 17:30**

1. **实验名称：**

利用Flex辅助遍历步骤中词法分析的代码实现

1. **实验学时：**

四小时

1. **实验目的：**

1.掌握词法分析的基本原理和功能；

2.熟悉正则表达式在词法单元识别中的应用；

3.学习并掌握 Flex 工具的使用方法；

4.实现一个针对 C 语言子集的词法分析器，能够正确识别关键字、运算符、分隔符、标识符和常数；

5.加深对编译器前端处理流程的理解。

1. **实验原理：**

词法分析是编译过程的第一阶段，其任务是从源程序字符流中识别出一个个具有独立意义的词法单元（Token）。每个 Token 包含类型（Type）和值（Value）两个属性。

本实验使用 Flex（Fast Lexical Analyzer Generator） 工具自动生成词法分析器。Flex 是一个词法分析器生成器，它根据用户定义的正则表达式规则，生成一个 C 语言函数 yylex()，该函数能够扫描输入字符串并返回识别出的 Token。

核心原理包括：

1.正则表达式匹配：为每类 Token 定义对应的正则表达式模式。

2.最长匹配原则（Longest Match）：当多个模式都能匹配时，选择能匹配最长子串的模式。

3.优先级原则：当多个模式匹配相同长度的子串时，按规则出现的先后顺序决定优先级。

4.DFA（确定性有限自动机）：Flex 将正则表达式转换为 DFA，高效地进行模式匹配。

1. **实验内容：**

设计 C 语言子集的关键字、运算符、分隔符、标识符和整数常量的正则表达式；

使用 Flex 工具编写 .l 文件，定义词法规则；

编写 token.h 头文件，定义所有 Token 类型的宏；

编译生成可执行的词法分析器；

测试分析器对各种输入的识别能力，验证其正确性。

1. **实验器材（设备、元器件）：**

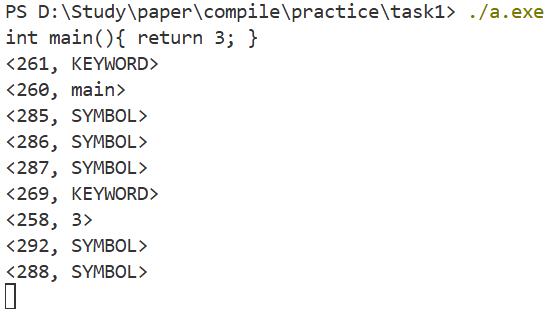
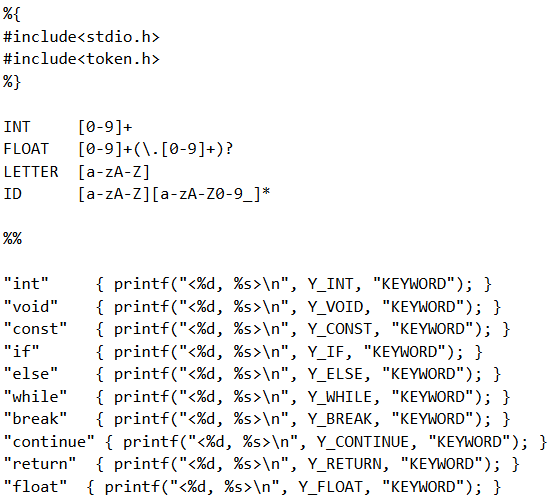
计算机一台

操作系统：Windows

开发工具：Flex、GCC 编译器、文本编辑器（VS Code）

头文件：token.h

1. **实验步骤：**
2. 编写token.h头文件
3. 编写task.lFlex文件
4. 编译并测试
5. **实验结果与分析（含重要数据结果分析或核心代码流程分析）**
6. token.h文件编写。
   1. Icoding中直接复制
7. lexer.l文件编写
   1. C代码声明区: 引入头文件，C语言定义常量，函数及变量声明
   2. 定义区: 对正则表达式进行重命名
   3. 规则区: 正则表达式成功匹配后执行的代码
   4. 子程序区: 存放额外的功能性函数或者覆写main函数

****lexer.l部分代码  
  
 3.编译并测试  
 输出符合预期结果

1. **总结及心得体会：**

通过本次实验，我成功实现了一个基于 Flex 的 C 语言子集词法分析器。我深入理解了词法分析的工作原理，掌握了正则表达式的设计方法，并熟练使用了 Flex 工具。

在实验过程中，我遇到了诸如 token.h 文件找不到、链接时报 WinMain 错误等问题，但通过查阅资料和调试，最终解决了这些问题，加深了对编译链接过程的理解。

本次实验让我认识到，词法分析虽然是编译的第一步，但其设计直接影响后续阶段的正确性。合理的 Token 分类和清晰的输出格式是构建完整编译器的基础。同时，自动化工具（如 Flex）极大地提高了开发效率，体现了编译技术的强大之处。

1. **对本实验过程及方法、手段的改进建议：**

增加错误处理机制：当前分析器对非法字符仅忽略，建议增加错误提示，如输出“未知字符”或报错位置。

支持更多 C 语言特性：可扩展支持浮点数、字符串常量、注释（//, /\* \*/）等。

输出到文件：将 Token 流输出到文件而非控制台，便于后续语法分析模块读取。

集成 Bison：下一步可与 Bison 结合，实现完整的语法分析，构建小型编译器原型。

使用更现代的工具链：考虑使用 CMake 管理项目，提高可移植性和可维护性。

**报告评分：**

**指导教师签字：**