**实 验 报 告**

|  |  |
| --- | --- |
| **课程名称：** | **编译技术** |
| **学生姓名：** | **葛子午** |
| **学生学号：** | **202330550381** |
| **学生专业：** | **软件工程** |
| **开课学期：** | **2024-2025第二学期** |

**软件学院**

**2025年06月**

# 实验四 中间代码生成

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **地 点：** | | | B7楼 | | 331房； | | **实验台号：** | |  | |
| **实验日期与时间：** | | | 2025.5.17 | | | | **评 分：** | |  | |
| **预习检查纪录：** | | |  | | | | **实验教师：** | |  | |
| **电子文档存放位置：** | | |  | | | | | | | |
| **电子文档文件名：** | | |  | | | | | | | |
| **批改意见：** | | |  | | | | | | | |
| 出勤情况（20%） | 课堂态度（20%） | | 实验报告质量（60%） | | | | | |
| 任务完成情况 | | 源代码注释 | | 报告内容 | |
| □ 出勤  □ 缺勤  □ 早退  □ 请假 | □ 认真  □ 不认真 | | □ 完成  □ 未完成 | | □ 优秀  □ 良好  □ 通过  □ 未通过 | | □ 优秀  □ 良好  □ 通过  □ 未通过 | |

# 报告内容

**1、实验目标**

（1）实现Decaf语言的递归下降语法分析器

1. 扩展支持for语句的语法分析（stmt -> for (forassign ; bool; forassign ) stmt）

（3）验证语法分析器对测试案例的解析能力

**2、实验环境**

（1）操作系统：Windows 11

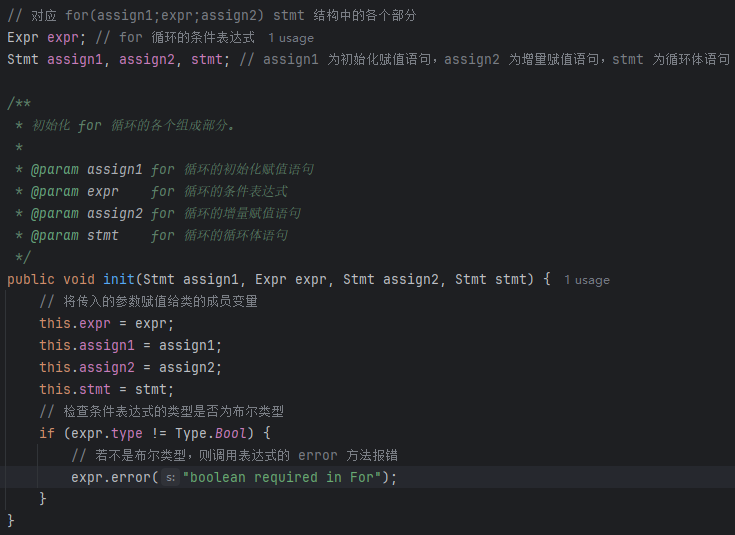
（2）开发工具：

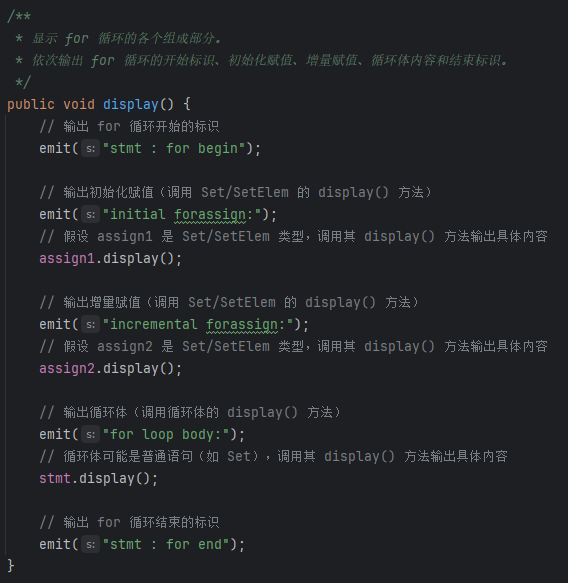
* Java开发环境：JDK 17
* 集成开发环境：IntelliJ IDEA
* 测试用例：包含for语句以及包含if语句的的Decaf程序

**3、关键数据结构和核心算法**

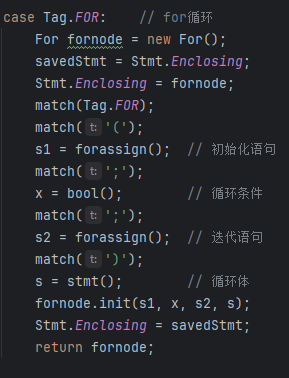
**3.1 数据结构**

（1）For语句节点类（inter/For.java）





1. Parser关键补充​



**3. 2 核心算法**

（1）递归下降分析算法​

算法特点：

每个非终结符对应一个解析方法

通过方法调用顺序隐式实现运算符优先级

使用lookahead token预测产生式

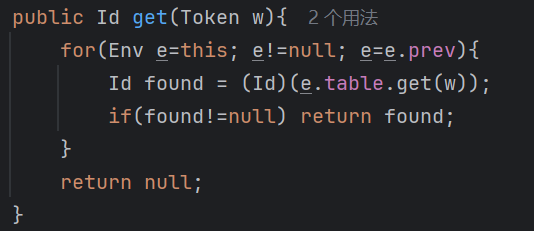




（2）符号表管理算法​

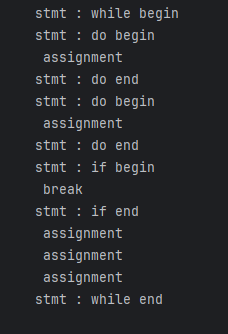
作用域链：通过prev指针实现嵌套作用域查询

类型系统：集成在symbols/Type中处理基础类型和数组类型

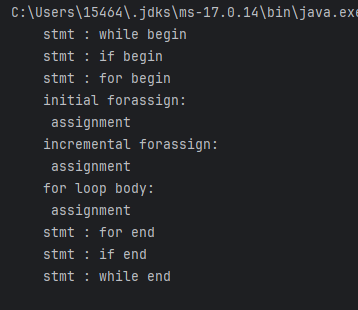


**3.4 运行结果**

（1）测试案例1



（2）测试案例2



**4、总结**

通过本次实验，我成功实现了Decaf语言的递归下降语法分析器，支持if、while、for等控制语句的解析，并构建了完整的语法树（AST）。在实现过程中，通过Env类管理嵌套作用域，确保语法错误的准确定位。实验验证了解析器的正确性，能够处理基础程序结构和多维数组访问。

未来可优化错误处理的精确性，并扩展支持++、--等语法糖。本实验加深了我对递归下降分析法、符号表管理以及编译器前端设计的理解，为后续语义分析阶段奠定了基础