目录

第 1 章 概述

1.1 本书的研制背景与目标

1.2 本书课程实验的构成

1.2.1 要实现的源语言

1.2.2 目标机的选择

1.2.3 中间表示

1.2.4 汇编代码的内部表示

1.2.5 编译器组件及本书的课程实验概览

1.3 开发环境与工具

1.3.1 开发环境与工具简介

1.3.2 环境变量设置

1.3.3 Eclipse 的安装和使用

1.3.4 XML 与 Ant 简介

1.4 实验软件包

1.5 课程实验开展建议

1.5.1 测试环境

1.5.2 课程设计提交要求

1.5.3 过程管理与控制

第 2 章 一个简单的程序解释器

2.1 SIMPLEMINIJOOL 语言

2.2 课程设计 1：一个简单的程序解释器

2.3 实验运行平台

2.3.1 实验平台接口

2.3.2 实验运行平台的工作机制

2.3.3 实验运行平台的使用

2.4 ECLIPSE AST

2.4.1 AST 节点类

2.4.2 AST 类

2.4.3 ASTVisitor 类

2.4.4 SimpleMiniJOOL 语言涉及的 AST 节点类

2.4.5 Eclipse AST 使用示例

2.5 AST 的图形化显示包——ASTVIEW

2.5.1 ASTView 包中的类

2.5.2 AST 节点的输出属性及其定制
2.5.3 在 ASTViewer 中显示节点对应的低级中间表示
2.6 设计模式
2.6.1 工厂方法模式
2.6.2 访问者模式
2.7 课程设计 1 的开发和测试指南
2.7.1 主要开发任务
2.7.2 实现 InterpVisitor 类的一些指导
2.7.3 在 Eclipse 下开发
2.7.4 在控制台下编译和运行
2.7.5 测试要求
第 3 章 词法分析
3.1 本章课程设计概述
3.2 MINIJOOL 语言的词法
3.3 课程设计 2-1：用 JFLEX 为 MINIJOOL 语言生成一个词法分析器
3.3.1 示例
3.3.2 MiniJOOL 语言的词法分析器构造
3.4 课程设计 2-2：手工编写一个简单的词法分析器
3.4.1 Block 语言的词法
3.4.2 示例
3.4.3 课程设计任务
3.4.4 编译和运行指南
3.5 课程设计 2-3：编写一个 NFA 生成器
3.5.1 MLex 词法规范描述语言
3.5.2 课程设计指导
3.5.3 课程设计任务
3.6 课程设计 2-4：编写一个词法分析器的生成器
3.6.1 LexerCodeGenerator 的输入和输出示例
3.6.2 课程设计指导
3.7 JFLEX 词法规范
3.7.1 用户代码
3.7.2 选项和声明
3.7.3 词法规则
3.7.4 如何匹配输入流
第 4 章 语法分析
4.1 SKIPOOMINIJOOL 语言的语法

4.1.1 类型、常量和变量
4.1.2 语句
4.1.3 表达式
4.1.4 SkipOOMiniJOOL 程序的总体结构
4.1.5 一个 SkipOOMiniJOOL 程序示例
4.2 本章课程设计概述
4.3 课程设计 3-1：手工编写一个语法分析器
4.3.1 SimpleBlock 语言
4.3.2 如何引用课程设计 2-2 的词法分析器类
4.3.3 课程设计指导
4.3.4 课程设计任务
4.4 课程设计 3-2：用 CUP 生成一个能分析合法程序的语法分析器
4.4.1 示例 1：SimpleBlock 语言的语法分析器
4.4.2 SkipOOMiniJOOL 语言涉及的 AST 节点类
4.4.3 示例 2：Block 语言的语法分析器
4.4.4 课程设计任务
4.5 课程设计 3-3：用 JavaCC 生成一个语法分析器
4.5.1 示例：Block 语言及其子语言的分析器
4.5.2 课程设计任务
4.6 课程设计 3-4：用 CUP 生成一个有错误处理能力的语法分析器
4.6.1 错误类型与错误信息管理
4.6.2 错误恢复与处理机制
4.6.3 示例
4.6.4 课程设计任务
4.7 课程设计 3-5：用 JavaCC 生成一个有错误处理能力的语法分析器
4.7.1 JavaCC 的错误恢复机制
4.7.2 错误恢复与处理示例
4.7.3 课程设计任务
4.8 CUP 与 YACC
4.8.1 YACC 简介
4.8.2 CUP 与 YACC 的文法规范描述文件的结构
4.8.3 文法符号
4.8.4 一个简单的例子
4.8.5 错误恢复
第 5 章 语义分析
5.1 SkipOOMiniJOOL 语言的静态语义

5.1.1 非形式描述部分
5.1.2 形式描述部分：类型系统
5.2 本章课程设计概述
5.3 课程设计 4-1 至课程设计 4-3 特征概述
5.4 课程设计 4-1：为源程序对应的 AST 构造符号表
5.4.1 Block 语言的语义特征
5.4.2 示例：为 Block 程序对应的 AST 构造符号表
5.4.3 课程设计指导和任务
5.5 课程设计 4-2：利用 AST 及其符号表信息开展语义检查
5.5.1 示例：利用 Block 程序对应的 AST 及其符号表信息开展语义检查
5.5.2 课程设计指导与任务
5.6 课程设计 4-3：对源程序关联的 AST 进行语义检查
5.6.1 示例：对 Block 程序对应的 AST 进行语义检查
5.6.2 课程设计注意事项
5.7 课程设计 4-4：在语法分析的同时构造符号表
5.7.1 课程设计 4-4 和 4-5 关联的文件
5.7.2 示例：带符号表构造的 Block 语言分析器
5.7.3 课程设计任务
5.8 课程设计 4-5：在语法分析的同时开展语义检查
5.8.1 示例：带语义检查的 Block 语言分析器
5.8.2 课程设计要点
第 6 章 中间表示的转换
6.1 本章课程设计概述
6.2 低级中间表示 LIR
6.2.1 LIR 设计特点简介
6.2.2 LIR 类型
6.2.3 LIR 操作数
6.2.4 LIR 语句
6.3 课程设计 5-1：SKIPOOMINIJOOL 语言的 AST 到 LIR 的转换器
6.3.1 课程设计指导
6.3.2 课程设计任务
6.4 课程设计 5-2 和课程设计 5-3
第 7 章 汇编语言及汇编代码的内部表示
7.1 汇编语言简介
7.2 MIPS 汇编语言
7.2.1 五阶段流水以及流水可见性

7.2.2 数据表示与对齐.....
7.2.3 寄存器.....
7.2.4 地址空间.....
7.2.5 寻址方式.....
7.2.6 过程调用约定.....
7.2.7 伪指令.....
7.2.8 常用 MIPS 汇编指令.....
7.3 x86 汇编语言.....
7.3.1 处理器执行周期和数据表示.....
7.3.2 寄存器.....
7.3.3 地址空间.....
7.3.4 寻址方式.....
7.3.5 过程调用.....
7.3.6 AT&T 汇编语法与 Intel 汇编语法的区别.....
7.3.7 伪指令.....
7.3.8 常用 x86 汇编指令.....
7.4 汇编代码的内部表示.....
7.4.1 AIR 库的组成及设计特点.....
7.4.2 使用 AIR 库编程的步骤.....
7.4.3 配置文件格式及示例.....
7.4.4 符号编码类的代码生成.....
7.4.5 汇编代码的表示、创建与输出.....
第 8 章 汇编代码生成.....
8.1 本章课程设计概述.....
8.2 汇编代码生成器设计中的问题.....
8.3 汇编代码生成器的开发步骤.....
8.4 EDU.USTC.CS.COMPILE.LIR.OPT 包及其使用.....
8.4.1 edu.ustc.cs.compile.opt.base 包：抽象层.....
8.4.2 edu.ustc.cs.compile.opt.lir 包：LIR 实现层.....
8.5 寄存器分配.....
8.5.1 线性扫描寄存器分配算法.....
8.5.2 实验软件包中的寄存器分配器.....
8.6 汇编代码生成的代码框架.....
8.6.1 基于 LIR 流图结构的汇编代码生成.....
8.6.2 AST 到汇编代码的生成.....
8.7 MIPS 汇编代码的生成与执行.....

8.7.1 MIPS 汇编语言与 SkipOOMiniJOOL 语言中部分结构的对应关系.....
8.7.2 利用 SPIM 解释执行 MIPS 汇编代码.....
8.8 x86 汇编代码的生成与执行.....
8.8.1 LIR 语句的转换.....
8.8.2 x86 汇编代码的生成.....
第 9 章 面向对象语言的编译.....
9.1 MINIJOOL 语言特征.....
9.1.1 一个 MiniJOOL 程序示例.....
9.1.2 MiniJOOL 程序的总体结构.....
9.1.3 类类型及其使用.....
9.2 本章课程设计概述.....
9.3 课程设计 8-1 构造 MINIJOOL 语言的词法语法分析器.....
9.4 课程设计 8-2 构造 MINIJOOL 语言的语义检查器.....
9.5 面向 MINIJOOL 语言的 LIR.....
9.5.1 对面向对象特征及其处理的设计考虑.....
9.5.2 LIR 中的类对象存储布局和类的类型信息表示.....
9.5.3 LIR 中虚方法表结构以及运行时类型判断.....
9.6 课程设计 8-3 构造 MINIJOOL-I 语言的 AST 到 LIR 的转换器.....
9.6.1 课程设计指导.....
9.6.2 课程设计任务.....
9.7 课程设计 8-4 构造 MINIJOOL-II 语言的 AST 到 LIR 的转换器.....
9.8 课程设计 8-5 构造 MINIJOOL-III 语言的 AST 到 LIR 的转换器.....
9.9 课程设计 8-6 构造 MINIJOOL 语言的 AST 到 LIR 的转换器.....
9.10 课程设计 8-7 构造 MINIJOOL 语言的汇编代码生成器.....
9.10.1 课程设计指导.....
9.10.2 课程设计任务.....
参考文献.....
附 录.....
附录 1 MINIJOOL 语言的词法记号类型及标识.....
附录 2 运算符的优先级与结合性.....
附录 3 MLEX 词法规范语言的 EBNF 表示.....
附录 4 SIMPLEMINIJOOL 语言语法的 EBNF 表示.....
附录 5 SKIPOOMINIJOOL 语言语法的 EBNF 表示.....
附录 6 MINIJOOL 语言语法的 EBNF 表示.....
附录 7 与本书有关的 ECLIPSE AST 节点类及其含义.....

附录 8　MIPS-SPIM 汇编语言的 EBNF 定义.....
附录 9　采用 AT&T 语法的 x86 汇编语言的 EBNF 定义 .....
附录 10　实验软件包提供的可用编译器组件.....