 

**实 验 报 告**

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

**软件学院**

**2025年05月**

# 实验四中间代码生成

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **地 点：** | | | B7 楼 | | 238房； | | **实验台号：** | | 186 | |
| **实验日期与时间：** | | | 2025/6/15 | | | | **评 分：** | |  | |
| **预习检查纪录：** | | |  | | | | **实验教师：** | | 应伟勤 | |
| **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
2. **理解中间代码生成原理**：深入理解三地址码的基本形式（x=y op z）及其在赋值语句、条件语句中的转换逻辑，掌握 Decaf 语言中if、while等控制语句转换为三地址码的实现方式。
3. **掌握代码生成实现**：通过分析inter包中Expr类的emitjumps函数、If类的gen函数等源码，理解布尔表达式和控制语句的中间代码生成逻辑。
4. **拓展编程能力**：参考while、if语句的生成逻辑，完成for语句中间代码生成的gen函数实现，提升对编译原理中中间代码生成阶段的实践能力。

2、实验环境

1. 硬件环境需求

Pentium4 2.8及以上，至少1G以上RAM

1. 软件环境需求

Linux或windowXP以后版本，使用 Java 开发环境（进行源码编辑与编译。针对 Decaf 语言的中间代码生成，需依赖已完成的 Decaf 语法分析阶段成果，在其基础上拓展中间代码生成功能。

3、关键数据结构和核心算法

3.1 数据结构

3. 2 核心算法

在以下地方添加了代码

For

    public void gen(int b, int a) {

        //add your implementation

        int label1=newlabel();

        int label2=newlabel();

        int label3=newlabel();

        stmt1.gen(b,label1);

        after=a;

        emitlabel(label1);

        expr.jumping(0,a);

        emitlabel(label2);stmt3.gen(label2,label3);

        emitlabel(label3);stmt2.gen(label3,label1);

        emit("goto L"+label1);

    }

3. 3 错误处理

3.4 运行结果

（1）对中间代码生成测试例1

    {

    int x;int y; int z;

    if( x < y )

     z = x;

    else

     z = y;

    z = z + z;

    }

得到合理结果

L1: iffalse x < y goto L5

L4: z = x

    goto L3

L5: z = y

L3: z = z + z

L2:

（2）对中间代码生成测试例2

    {

    int a;int b; int c; int min;

    if(a<b)

    {

        min=a;

    }

    while(min+a<c)

    {

        a=a+1;

    }

    }

得到合理结果

L1: iffalse a < b goto L3

L4: min = a

L3: t1 = min + a

    iffalse t1 < c goto L2

L5: a = a + 1

    goto L3

L2:

（3）对中间代码生成测试例3

{

    int a;int b; int c; int min; int i;

    if(a<b)

    {

        min=a;

    }

    for(i=1;i<100;i=i-1)

    {

        a=a+1;

    }

}

得到合理结果

L1: iffalse a < b goto L3

L4: min = a

L3: i = 1

L5: iffalse i < 100 goto L2

L6: a = a + 1

L7: i = i - 1

    goto L5

L2:

4、总结

1.理解三地址码（如x=y op z）在赋值、if等语句中的转换逻辑，分析inter包中emitjumps和gen函数的中间代码生成机制。

2.参考现有控制语句逻辑，完善for类的gen函数，实现for语句到三地址码的转换。

3.掌握Decaf语言中间代码生成流程，明确三地址码在编译阶段的桥梁作用，提升编译原理实践能力。