

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | 编译原理 |
| 姓 名： |  |
| 学 院： |  |
| 系： |  |
| 专 业： |  |
| 学 号： |  |
| 指导教师： |  |

标题

组员信息

序言 （包括对整个编译器的描述，对所提交的各个文件的说明，组员的分工）

1. 词法分析（正规表达式，实现原理和方法）
2. 语法分析 （上下文无关文法，实现原理和方法）
3. 语义分析 （实现方法）
4. 优化考虑 （每个阶段的优化考虑）
5. 代码生成 （所有语句的代码生成的处理）
6. 测试案例 （每个语句成分的测试案例，至少两个复杂语句组合后的测试案例）

**cp README**

1. **实验目的**

XXXXXXXXX（编）

1. **实验内容：**

XXXXXXXXX（编）

1. **词法分析**

正规表达式描述：

XXXXXXXXX（根据词法来写）

原理：

XXXXXXXXX（书上关于Scan的原理+我们实现的原理+错误恢复及报错原理）

实现：

XXXXXXXXX（lex代码）

1. **语法分析**

文法描述

XXXXXXXXX（直接把文法搬过来即可）

原理：

yacc is a LALR(1) parser generator. That is, it generates src code regarding

XXXXXXXXX（书上关于Yacc的底层原理+我们实现的原理+如何生成树和如何错误恢复和报错（尹幽潭））

实现：

XXXXXXXXX（yacc代码+可视化代码+语法生成树样例：包括代码及对应语法树图片）

1. **语义分析的方法描述**
2. **中间代码**

中间代码格式：

XXXXXXXXX

中间代码生成的实现：

XXXXXXXXX

中间代码优化：

XXXXXXXXX

**7.可选的目标代码生成**

1. **数据结构描述**

语法分析使用的语法生成树数据结构：

XXXXXXXXX（需要尹幽潭写一下）

后续工作中使用的数据结构：

1. **测试**

每个正常测试用例包括代码，对应生成树图片，对应中间代码及其说明。

异常（报错）测试用例仅报错截图及其说明即可

1. **组员分工情况**