**编译原理**

班级：22计算机科学与计算机留学生1

姓名：LOW REN HONG 学号：2021529620004

实验一 TINY语言的词法分析

1. 实验目

构造tiny语言的词法分析器（扫描器），要求利用第三方的lex工具进行构造。实验结果：构造出的扫描器，能够读入教材样例中给出的tiny语言的示例代码，分解成token输出。

1. 实验设计

（评价依据实验方案设计是否合理）

使用第三方的lex工具进行构造，根据tiny语言的词法规则，定义正则表达式模式来匹配不同的token，生成词法分析器（扫描器）代码。通过对教材样例中的tiny语言示例代码进行测试，验证词法分析器的功能是否正确。

1. 内容和步骤

1代码：

int x = 10;

float y = 3.14;

if (x > y) {

printf("x is greater than y\n");

} else {

printf("x is not greater than y\n");

}

2、结果：

int x = 10;

TOKEN\_INT

TOKEN\_UNKNOWN(x)

TOKEN\_UNKNOWN(=)

TOKEN\_NUM(10)

TOKEN\_SEMICOLON

float y = 3.14;

TOKEN\_FLOAT

TOKEN\_UNKNOWN(y)

TOKEN\_UNKNOWN(=)

TOKEN\_NUM(3)

TOKEN\_UNKNOWN(.)

TOKEN\_NUM(14)

TOKEN\_SEMICOLON

if (x > y) {

TOKEN\_IF

TOKEN\_LPAREN

TOKEN\_UNKNOWN(x)

TOKEN\_UNKNOWN(>)

TOKEN\_UNKNOWN(y)

TOKEN\_RPAREN

TOKEN\_LBRACE

printf("x is greater than y\n");

TOKEN\_PRINTF

TOKEN\_LPAREN

TOKEN\_UNKNOWN(")

TOKEN\_UNKNOWN(x)

TOKEN\_UNKNOWN(is)

TOKEN\_UNKNOWN(greater)

TOKEN\_UNKNOWN(than)

TOKEN\_UNKNOWN(y)

TOKEN\_UNKNOWN(\)

TOKEN\_UNKNOWN(n)

TOKEN\_UNKNOWN(")

TOKEN\_RPAREN

TOKEN\_SEMICOLON

} else {

TOKEN\_RBRACE

TOKEN\_ELSE

TOKEN\_LBRACE

printf("x is not greater than y\n");

TOKEN\_PRINTF

TOKEN\_LPAREN

TOKEN\_UNKNOWN(")

TOKEN\_UNKNOWN(x)

TOKEN\_UNKNOWN(is)

TOKEN\_UNKNOWN(not)

TOKEN\_UNKNOWN(greater)

TOKEN\_UNKNOWN(than)

TOKEN\_UNKNOWN(y)

TOKEN\_UNKNOWN(\)

TOKEN\_UNKNOWN(n)

TOKEN\_UNKNOWN(")

TOKEN\_RPAREN

TOKEN\_SEMICOLON

}

TOKEN\_RBRACE

1. 实验结论:

1 理论基础（评价依据 理论知识非常清楚）

这个的实验里使用了lex工具来建立一个tiny语言的词语分析其，通过tiny语句的词法分析进行定义，lex工具能够根据这些规则将输入的示例代码分解成一系列的token来输出，

2、分析和总结（评价依据：是否能够对实验结果作出完整和准确的描述，是否能够捕捉到实验中的各种现象，是否有强的信息综合能力，是否能得出正确的结论。）

在实验中，通过lex的工具成功构造了一个能读入tiny语言的实列代码并输出成token的扫描器。扫描器能够根据预先定义的词法规则，将输入的代码识别到token中，输出相应的token类型和值。使用lex工具能够方便地构造复杂语言的词法分析器，从而为后续的语法分析和语义分析等步骤奠定了基础。

3、对工具的评价（优缺点及其局限性的总结）

优点的话有简单易懂，高效快速，灵活性等等。但其局限性就仅限于词法分析，正则表达式

的有限性，可读性比较低。