信 息 工 程 学 院

《编译原理》实验报告

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **学院**:信息工程学院 | **班级**:软工2001班 | **姓名**:张宇晨 | **学号**:2020012249 | **成绩**: |

# 实习目标

1. 熟悉Javacc 工作原理，掌握词法分析相关的类：

**Token**：表示单词的类。每个Token 对象的主要成员：int kind 表示单词的种别； String image 存储了token 所代表的内容；int beginLine 表示单词所处的行；int beginColumn 表示单词所处的列。

**TokenManager**：词法分析器，用来识别源程序中的单词序列。

**SimpleCharStream**：词法分析器的输入流。可以指定源程序来自于标准输入还是文件。

**TokenMgrError**：词法错误类，当识别到非法单词时，弹出的错误对象。

**Constants**：保存已定义所有单词的接口，针对每种单词定义其对应的种别码。

2. 使用以上的类，编制一个读单词过程，从输入的源程序中，识别出各个具有独立意义的单词，即基本保留字、标识符、常数、运算符、分隔符五大类。并依次输出各个单词的内部编码及单词符号自身值。

# 实验过程

1. 词法定义
   1. 关键字KEYWORD

if else while int return void main char等

* 1. 标识符IDENTIFIER

正则表达式定义，由字母、数字、下划线“\_”组成，且首字母非数字

* 1. 分隔符SEPARATER

, ; { } ( ) [ ] 还有单引号

* 1. 运算符OPERATOR

+ - \* / % > < =

* 1. 常数CONSTANT

简化为只表示整型常数

* 1. 过滤字符

空格“ ”、制表符“\t”、回车“\r”和换行符“\n”

1. 将词法定义写入jjt文件

默认文件里已经有了标识符、常数、和过滤字符的定义，故只需将剩余的关键字、分隔符、运算符的词法定义写入jjt文件。

关键字:

**TOKEN** : /\* LITERALS \*/

**{**

**<** INTEGER\_LITERAL :

< DECIMAL\_LITERAL > **([** "l", "L" **])?**

**|** < HEX\_LITERAL > **([** "l", "L" **])?**

**|** < OCTAL\_LITERAL > **([** "l", "L" **])?**

**>**

**|** **<** #DECIMAL\_LITERAL : **[** "1"**-**"9" **]** **([** "0"**-**"9" **])\*** **>**

**|** **<** #HEX\_LITERAL : "0" **[** "x", "X" **]** **([** "0"**-**"9", "a"**-**"f", "A"**-**"F" **])+** **>**

**|** **<** #OCTAL\_LITERAL : "0" **([** "0"**-**"7" **])\*** **>**

**}**

**TOKEN**:/\*KERWORDS\*/

**{**

**<** IF : "if"**>**

**|** **<** ELSE : "else"**>**

**|** **<** INT : "int"**>**

**|** **<** RETURN : "return"**>**

**|** **<** VOID : "void"**>**

**|** **<** WHILE : "while"**>**

**}**

**TOKEN** : /\* IDENTIFIERS \*/

**{**

**<** IDENTIFIER :

< LETTER >

**(**

< LETTER >

**|** < DIGIT >

**)\*** **>**

**|** **<** #LETTER : **[** "\_", "a"**-**"z", "A"**-**"Z" **]** **>**

**|** **<** #DIGIT : **[** "0"**-**"9" **]** **>**

**}**

**TOKEN**:/\*SPARATER\*/

**{**

**<** COMMA:"," **>**

**|** **<** SEM : ";"**>**

**|** **<** LFBRACKET:"{" **>**

**|** **<** RFBRACKET:"}" **>**

**|** **<** LBRACKET:"[" **>**

**|** **<** RBRACKET:"]" **>**

**|** **<** LRBRACKET:"(" **>**

**|** **<** RRBRACKET:")" **>**

**}**

**TOKEN** :/\*OPERATOR\*/

**{**

**<** ADD:"+" **>**

**|** **<** MUL:"\*" **>**

**|** **<** MINUS:"-" **>**

**|** **<** DIVISOR:"/" **>**

**|** **<** LESSTHAN:"<" **>**

**|** **<** GREATERTHAN:">" **>**

**|** **<** EQUAL:"=" **>**

**}**

1. 修改main
   1. 制定一个输入流，FileInputStream文件输入流
   2. 制定一个输出流，FileWriter 文件输出流
   3. 实例化一个词法分析器的输入流SimpleCharStream
   4. 实例化一个词法分析器LexicalTokenManager的对象
   5. 调用getnextToken()，获得Token对象
   6. 输出Token对象的属性(kind,image,beginLine,beginColumu)到文件输出流
   7. 关闭输入输出流

控制输出代码如下：

System.out.println("Reading from file input...");

String inputpath ="src/shixi1/testinput3";

String outputpath ="src/shixi1/output";

InputStream fin = **null**;

FileWriter fw =**null**;

**try** {

fin =**new** FileInputStream(inputpath);

} **catch** (FileNotFoundException e1) {

// TODO 自动生成的 catch 块

e1.printStackTrace();

}

**try** {

fw = **new** FileWriter(outputpath, **true**);

} **catch** (IOException e1) {

// TODO 自动生成的 catch 块

e1.printStackTrace();

}

PrintWriter pw = **new** PrintWriter(fw);

pw.print(inputpath+"的词法分析结果：\n");

pw.print("TOKEN\t"+"KIND\t"+"LOCATION\n");

pw.flush();

**try**

{

SimpleCharStream stream = **new** SimpleCharStream(fin);

MyNewGrammarTokenManager mtk = **new** MyNewGrammarTokenManager(stream);

Token token = mtk.getNextToken();

**while**(token.kind != 0) {

pw.print(token.image+"\t\t"+token.kind+"\t\t"+"("+token.beginLine+","+token.beginColumn+")\n");

pw.flush();

token = mtk.getNextToken();

}

pw.close();

fin.close();

}

**catch** (Exception e)

{

System.out.println("Oops.");

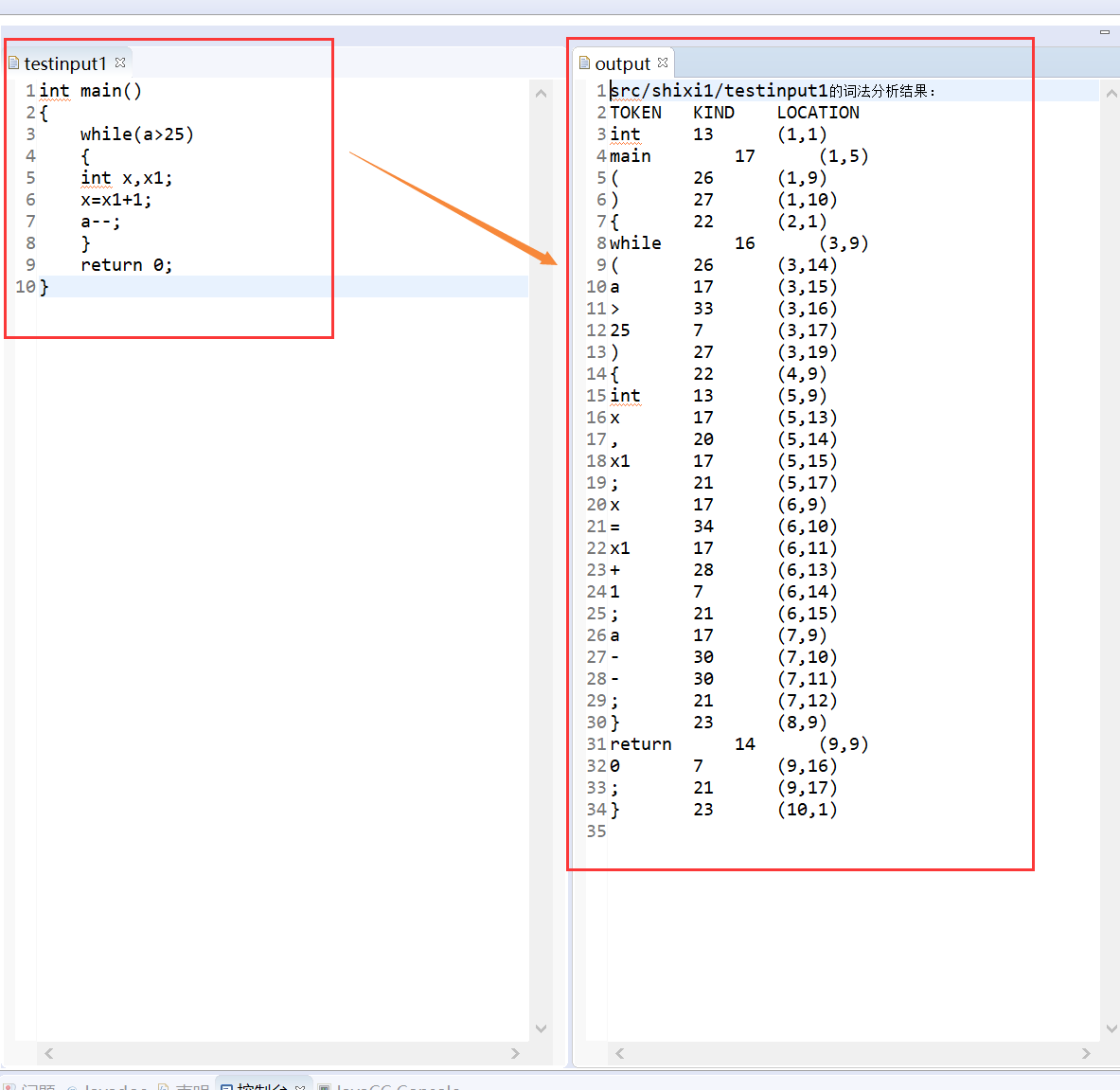
System.out.println(e.getMessage());

}

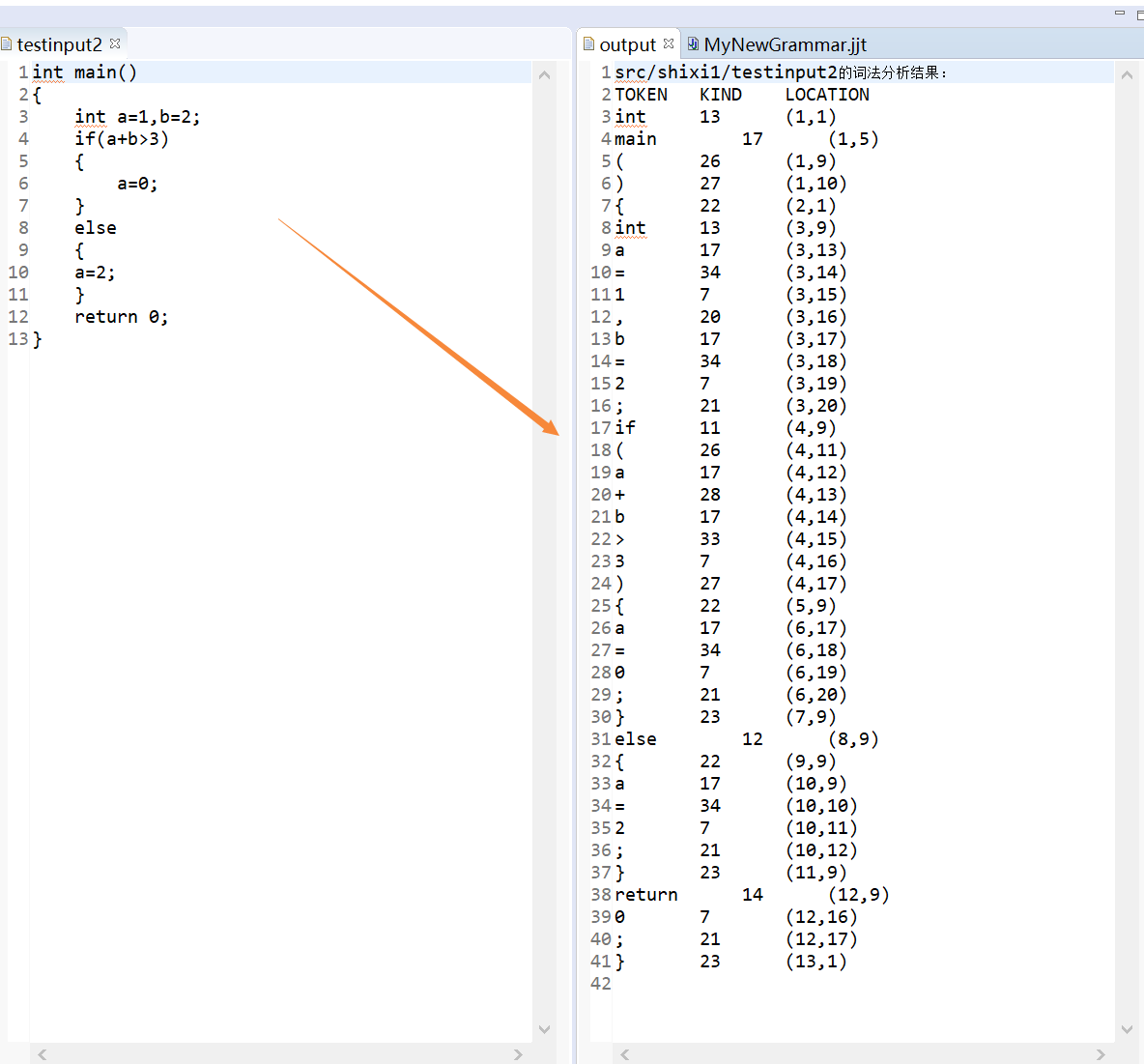
System.out.println("Analysis done! Please check it in file output.");

# 实验结果

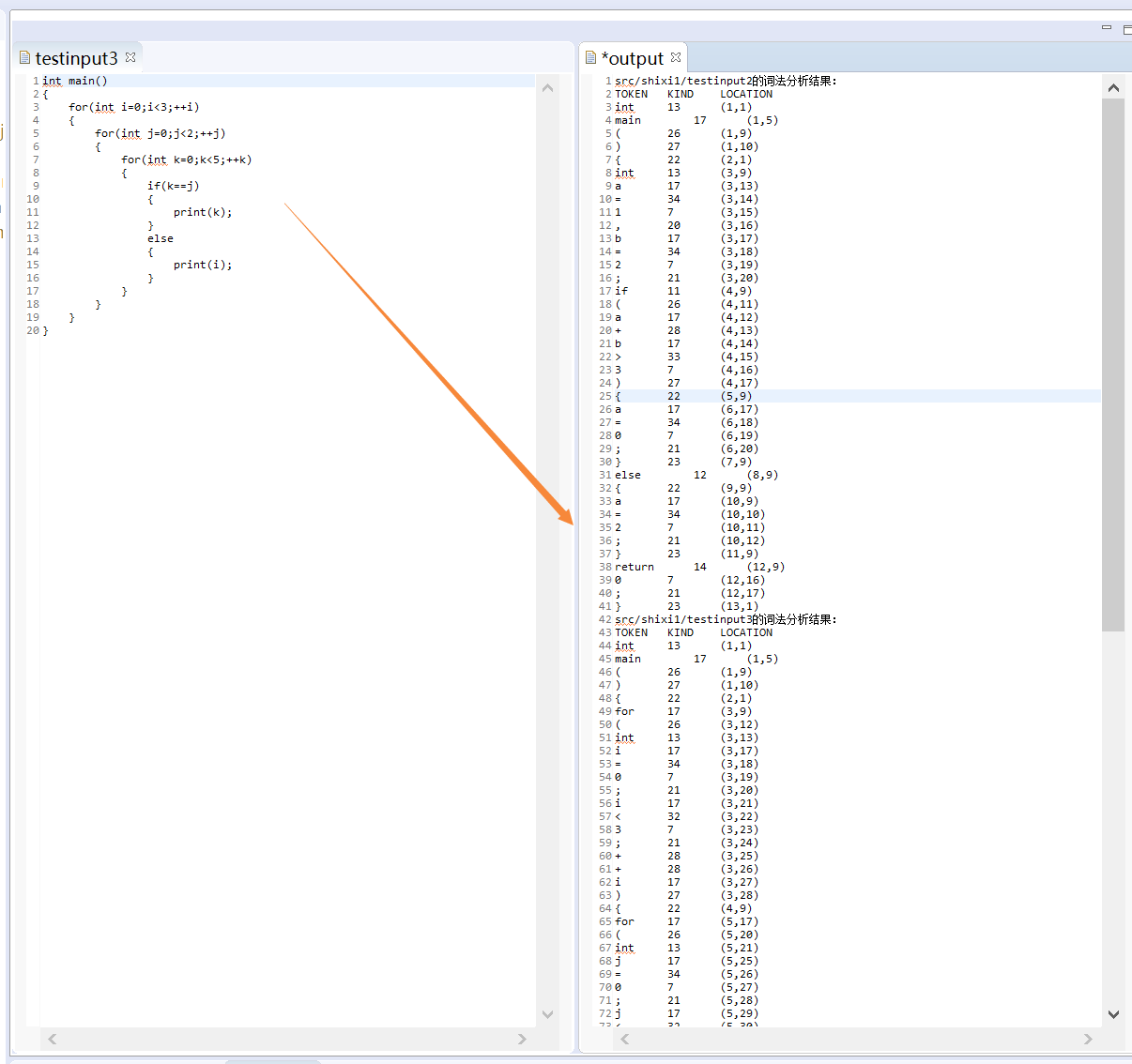
Testinput1:



Testinput2:



Testinput3:



# 疑问和建议

## 疑问

本次实习是词法分析的设计实现。javacc 的词法分析器 Tokenmanager 从左往右逐个字符地扫描源程序，根据定义好的字符规则分析产生一个个的单词符号Token即对输入的字符流进行处理，再输出单词流。本次实习比较简单，实习过程中遇到的问题和疑惑都已经通过请教老师和同学解决。

## 建议

希望老师可以装一个控制班级所有电脑屏幕的广播软件，这样在实习教学的时候可以不用看大屏幕，看自己面前的屏幕。大屏幕上的字确实是难以看清。

最后感谢老师在实习过程中的悉心指导与帮助！