# MJava的语法规则和课程设计要求

## MJava的语法规则

Goal-> MainClass *{* ClassDeclaration *}* EOF

MainClass->**"class"** Identifier **"{"** **"public"** **"static"** **"void"** **"main"** **"("** **"String"** **"["** **"]"** Identifier **")"** **"{"** Statement **"}"** **"}"**

ClassDeclaration->**"class"** Identifier *[* **"extends"** Identifier *]* **"{"** *{* VarDeclaration *}* *{* MethodDeclaration *}* **"}"**

VarDeclaration ->TypeIdentifier **";"**

MethodDeclaration->**"public"** Type Identifier **"("** *[* Type Identifier *{* **","** Type Identifier *}* *]* **")"** **"{"** *{* VarDeclaration *}* *{* Statement *}* **"return"** Expression **";"** **"}"**

Type->**"int"** **"["** **"]"***|***"boolean"***|***"int"***|* Identifier

Statement->**"{"** *{* Statement *}* **"}"***|***"if"** **"("** Expression **")"** Statement **"else"** Statement *|***"while"** **"("** Expression **")"** Statement *|***"System.out.println"** **"("** Expression **")"** **";"***|* Identifier **"="** Expression **";"***|* Identifier **"["** Expression **"]"** **"="** Expression **";"**

Expression-> Expression *(* " &&" | "<" | "+" | "-" | "\*" *)* Expression *|* Expression **"["** Expression **"]"***|* Expression **"."** **"length"***|* Expression **"."** Identifier **"("** *[* Expression *{* **","** Expression *}* *]* **")"***|* IntegerLiteral *|***"true"***|***"false"***|* Identifier *|***"this"***|***"new"** **"int"** **"["** Expression **"]"***|***"new"** Identifier "(" ")"*|* **"!"** Expression *|***"("** Expression **")"**

注意事项

1. 双引号括起来的内容是终结符（即单词的类型）
2. 红色{ }，表示其中的内容可以重复出现0至n次
3. 红色[ ]，表示其中的内容可有可无
4. EOF 表示的是词法分析器在扫描到文件尾时返回的token
5. 红色字符表示范式中的符号，而非词法分析中解析的token。

3） 空白、换行符和制表符在词法分析时忽略掉。

## 语法分析程序课程设计要求

语法分析程序设计与实现需要按照以下要求完成。

1)为描述语法规则的文法给出相应的抽象语法树结构，提交相应的语法树定义的文件,语法树的定义尽量简洁易懂，必要时提供说明（25分）。

2)写出识别语法范畴的语法分析程序，提交相应的可执行源程序。该语法分析程序的具体要求如下（75分）：

(1)该语法分析程序需要读取词法分析的结果即读取文件tokenOut.txt中的单词序列进行语法分析。

(2)将tokenOut.txt中的表达式经语法分析后将结果表示为相应的抽象语法树。

(3)将语法分析的结果打印到文件SyntaxOut.txt中。

(4)给出测试方法描述,提交readme.doc,如果你还有其它需要说明的问题 须写在readme.doc中。

1、算法是什么

2、每一类语法结构 测试输入

3、看代码结构

4、必须有可执行文件 语言