大作业报告

一、选题分析

matlab blablabla

二、参考文献

Terence Parr·The Definitive ANTLR 4 Reference

<https://github.com/mattmcd/ParseMATLAB>

https://github.com/vinodkhare/MatlabAntlrGrammar/blob/master/MATLAB.g4

本作业是基于上述链接代码更改.上述两个版本的代码均是个人项目，均没有完成全部功能，切部分实现有很大的瑕疵。我们主要在此的基础上进行了很多改进，我们的主要工作如下

(1）对整体语法结构、词法结构进行了重构，使之更加清晰易懂

(2) 更改了浮点数词法的错误，原工程对于带e的浮点数不能支持新的matlab浮点数特性

(3) 重新修改了数组 元胞数组的语法结构，使每一行为一个单元

(4)

(last) 完整的分析了整个语法词法结构，给代码加入了注释

三、整体架构

整体分两个.g4文件，MATLABLexer.g4定义了所有词法结构的grammar。MATLABParser.g4定义了所有语法结构的grammar。利用antlr4自动生成词法分析器语法分析器，分析输入的字符串，并打印

public static void main( String[] args) throws Exception

{

ANTLRInputStream input = new ANTLRInputStream( System.in);

MATLABLexer lexer =new MATLABLexer(input);

CommonTokenStream tokens = new CommonTokenStream(lexer);

MATLABParser parser = new MATLABParser(tokens);

ParseTree tree = parser.fileDecl();//用fileDec rule生成语法分析树

System.out.println(tree.toStringTree(parser)); // print LISP-style tree

}

四、词法结构

词法包含关键字、符号、注释、基本常量（数字字符串）、标识符

antlrt4中以词法rule出现的顺序做为其优先级顺序，故应按照上述顺序来组织grammar结构

(1)关键字

// 关键字

*FUNCTION* : 'function' ;

*CLASSDEF* : 'classdef' ;

*PROPERTIES* : 'properties' ;

*METHODS* : 'methods' ;

*END* : 'end' ;

*IF* : 'if' ;

*ELSEIF* : 'elseif' ;

*ELSE* : 'else' ;

*WHILE* : 'while' ;

*FOR* : 'for' ;

*SWITCH* : 'switch' ;

*CASE* : 'case' ;

*OTHERWISE* : 'otherwise' ;

(2)符号略

包括算数运算符 关系运算符 逻辑运算符 特殊字符

(3)注释

COMMENT

: '%'.\*? NL -> skip ;

BLOCKCOMMENT : '%{' .\*? '%}' -> channel(HIDDEN);

示例 %123

%{ 123

}%

(4)基本常量

fragment

LETTER : [a-zA-Z] ;

fragment

DIGIT : [0-9] ;

fragment

ESC : '\'\'' ;

INT : DIGIT+; //普通int

FLOAT : DIGIT+ '.' DIGIT\*

| '.' DIGIT+

; //普通浮点数

SCI : (INT|FLOAT) 'e' INT ; //科学计数法

STRING : '\'' (ESC|.)\*? '\'' ; //字符串

(5)标示符

ID : LETTER (LETTER|DIGIT|'\_')\* ;//只能以字母开头的 字母数字下划线的组合

(6)特殊字符

五、语法结构

(1)总体语法结构

fileDecl

: (functionDecl | classDecl)? (functionDecl\* | partialFunctionDecl\*)

| partialFunctionDecl+

| statement+ // 普通语句

| *EOF*

;

functionDecl 普通函数 sequence

partialFunctionDecl是不加end的函数 sequence

statement是普通语句定义

对于fileDecl有三种替换项

只包含普通执行语句

只包含不加end的函数

包含带end的函数（主函数）/类 加一些包含或不包含end的函数

(2)函数

functionDecl

: partialFunctionDecl *END* NL+

;

函数由部分函数加end 加换行符

partialFunctionDecl

: *FUNCTION* outArgs? *ID* inArgs? endStat statement\*

;

部分函数由function 与返回值 标示符 输入参数 状态结束符endstat（回车换行或逗号分号，在matlab中以上均可以表明一条语句的结束类似于c语言的分号）

函数内部里面语句statement

（3）statement

statement

: (stat endStat)

;

stat

: ID *EQUALS* expr

| ifStat

| whileStat

| caseStat

| expr

| *NL*

;

基本的语句由如下几种基本格式

标示符 =xx表达式

if语句

while语句

switch case语句

表达式

(4)矩阵相关语法 注 该语法与下述表达式语法之间有递归关系

Array: 例子[1 2 3;3 4 5]

arrayExpr

: *LBRACK* matrix? *RBRACK*

;

cell: 例子{1 2 3;3 4 5}

cellExpr

: *LBRACE* matrix? *RBRACE*

;

row

: expr (*COMMA*? expr)\*

;

matrix

: row((*SEMI*|*NL*) row)\*

;

row 是由一个一个expr元素组成中间由，或空格分离。每一行之间用分号分离

每一个row由expr组成 而expr又可以由array组成 相互递归生成

例： [[1 2 3] [1 2 3];[1 2 3] [1 2 3]]

(5)表达式

expr

: expr *LPAREN* exprList *RPAREN*

| expr (*TRANS*|*CTRANS*)

| expr (*MPOW*|*POW*) expr

| (*PLUS*|*MINUS*|*NOT*) expr

| expr (*MTIMES*|*TIMES*|*MLDIVIDE*|*LDIVIDE*|*MRDIVIDE*|*RDIVIDE*) expr

| expr (*PLUS*|*MINUS*) expr

| expr *COLON* expr

| expr (*NOT*|*EQUALTO*|*GT*|*LT*|*GE*|*LE*) expr

| expr *VECAND* expr

| expr *VECOR* expr

| expr *SCALAND* expr

| expr *SCALOR* expr

| ID

| *INT* | *FLOAT* | *SCI*

| *STRING*

| arrayExpr

| cellExpr

| *LPAREN* expr *RPAREN*

;

表达式整体是递归的定义的语法结构，antlr支持左递归

· 表达式首先包含基本的数据结构 int float 科学计数发float 标示符 字符串 数组表达式 cell表达式。然后再次基础上递归的加上符号以此定义其他表达式

· expr *LPAREN* exprList *RPAREN*

引用数组的元素 例 a(1,1) 或者是函数调用 func(inputArg)

· expr (*TRANS*|*CTRANS*)

对矩阵的转置 例 a.’

六、成员分工

李文旭 词法分析 语法分析整体结构 函数 矩阵部分

邢桢 ifelse while for switch等控制语句

勾子懿 算数运算 布尔运算