Студент: Никита Сазанович
Группа: АУ 302

Домашнее задание #7

***Задача* 1.** *Предложить удобный конкретный синтаксис языка L для приведенного абстрактного.*

Доказательство.

Ничего не описывает синтаксис лучше, чем сам формат для генерации *Antlr* – (оставлены только правила для парсера – токены опущены):

```
file
    : (procedure)* block EOF
    ;

procedure
    : PROCEDURE IDENTIFIER L_BRACE parameterNames R_BRACE blockWithBraces
    ;

parameterNames
    : (IDENTIFIER (COMMA IDENTIFIER)*)?
    ;

block
    : (statement)*
    ;

blockWithBraces
    : L_CURLY_BRACE block R_CURLY_BRACE
    ;

statement
    : assignment SEMICOLON
    | readCall SEMICOLON
    | writeCall SEMICOLON
    | procedureCall SEMICOLON
    | whileBlock
    | ifStatement
    ;

assignment
    : IDENTIFIER ASSIGN expression
    ;

readCall
    : READ L_BRACE IDENTIFIER R_BRACE
    ;

writeCall
    : WRITE L_BRACE expression R_BRACE
    ;

procedureCall
    : IDENTIFIER L_BRACE arguments R_BRACE
    ;

arguments
    : (IDENTIFIER (COMMA IDENTIFIER)*)?
    ;

```

```
whileBlock
    : WHILE L_BRACE expression R_BRACE blockWithBraces
    ;

ifStatement
    : IF L_BRACE expression R_BRACE THEN blockWithBraces (ELSE blockWithBraces)?
    ;

expression
    : lorExpression
    ;

lorExpression
    : landExpression (LOR landExpression)*
    ;

landExpression
    : equivalenceExpression (LAND equivalenceExpression)*
    ;

equivalenceExpression
    : relationalExpression ((EQ | NQ) relationalExpression)*
    ;

relationalExpression
    : additiveExpression ((GT | LT | GTE | LTE) additiveExpression)*
    ;

additiveExpression
    : multiplicativeExpression ((PLUS | MINUS) multiplicativeExpression)*
    ;

multiplicativeExpression
    : atomicExpression ((MULTIPLY | DIVIDE | MODULUS) atomicExpression)*
    ;

bracedExpression
    : L_BRACE expression R_BRACE
    ;

atomicExpression
    : bracedExpression
    | IDENTIFIER
    | NUMBER
    ;
```

.

Собственно, это синтаксис предполагает, что язык будет игнорировать *whitespace*. Для ограничения блоков, используются фигурные скобки.

Также, в этом синтаксисе разделение выражений происходит с помощью точек с запятыми. Если statement содержит блок кода (if statement и while statement), то точка с запятой не нужна.

Также, были оставлены скобки и запятые для аргументов и параметров процедур.

Чтобы ознакомиться с полным набором правил для лексера и парсера, обратитесь к L.g4.

□