Report - Logo Project

Authors: Bürgi Christian burgc5@bfh.ch

Haring Patrick harip1@bfh.ch

Leiser Michael leism3@bfh.ch

Class: I3r

Location: Biel/Bienne

Date: 10.01.2014

Version: 1

Introduction

This mini-project represents a parser for Logo for Java using JavaCC. The parser was implemented during the module "Automata and formal languages".

Grammar

```
Program    = "LOGO" Identifier { Subroutine } { Statement } "END"

Subroutine = "TO" Identifier { Parameter } { Statement } "END"

Statement  = "CS" | "PD" | "PU" | "HT" | "ST"
           | "FD" NExpr | "BK" NExpr | "LT" NExpr | "RT" NExpr
           | "WAIT" NExpr
           | "REPEAT" NExpr "[" { Statement } "]"
           | "IF" BExpr "[" { Statement } "]"
           | "IFELSE" BExpr "[" { Statement } "]" "[" { Statement } "]"
           | Identifier { NExpr }

NExpr      = NTerm { ( "+" | "-" )  NTerm }

NTerm      = NFactor { ( "*" | "/" ) NFactor }

NFactor    = "-" ( Number | REPCOUNT | Parameter | "(" NExpr ")" ) |
         Number | REPCOUNT | Parameter | "(" NExpr ")"
```

```
BExpr     = BTerm { "OR" BTerm }

BTerm     = BFactor { "AND" BFactor }

BFactor   = "TRUE" | "FALSE" | "NOT" "(" BExpr ")"
    | NExpr ( "==" | "!=" | "<" | ">" | "<=" | ">=" )  NExpr

Comments start with "#" with scope until the newline

Numbers are real numbers

Identifiers start with a letter followed by letters or digits

Parameters are ":" followed by Identifier

Identifiers, parameters, keywords in uppercase only
```

Solution

The implementation of the parse tree was straight-forward by transerfing the given Grammar into a tree structure, where each node in the tree representing a java method call in the Logo.jj file. These mehods also contain the generated java code, which is return as a String in the return statements of the methods. On top of the call hierarchy they are finally printed out.

The "REPCOUNT" which is actually a "for-loop" in java, has no variable or counter name in logo but has one in java. We had to create a counter name which is not the same in a nested loop. So we have added an int variable in Logo.jj which is incremented when entering a "REPCOUNT" statement and decremented when leaving this statement. The int variable can be castet to a char variable which can be used as a unique counter variable name.

Test

We have created the pictures with the .logo files and "Microsoft Windows Logo". We have created the pictures with the .java files and "Eclipse" with the "LogoPrimitives.class". Then we have checked the generated pictures against each other and they're looking equally.

Limitations

The Direction of the Turtle in java is per default in the right direction whereas in logo is in the top direction.