HÁSKÓLI ÍSLANDS

ÞÝÐENDUR

Scanner-NanoMorpho

Höfundar:
Hjalti Geir Garðarsson
Guðmundur Óli Norland
Egill Ragnarsson

Kennari:
Snorri Agnarsson

February 7, 2020

nanomorpho.jflex

```
/*
JFlex scanner for NanoMorpho

Based on Snorri Agnarssons NanoLisp scanner.

Authors:  Hjalti Geir Garðarsson
          Egill Ragnarsson
          Guðmundur Óli Norland

Running the program:
  Compile:
    java -jar JFlex-full-1.7.0.jar nanomorpho.jflex
    javac NanoMorpho.java
  Run:
    java NanoMorpho <input_file> > <output_file>

Use the makefile:
*/

import java.io.*;

%%

%public
%class NanoMorpho
%unicode
%byaccj

%{

// This part becomes a verbatim part of the program text inside
// the class, NanoMorpho.java, that is generated.

// Definitions of tokens:
static final int ERROR = -1;

static final int NAME = 1001;
static final int LITERAL = 1002;
static final int OPNAME = 1003;
```

```java
// Decleration <decl>
static final int VAR = 1010;

// Expression <expr>
static final int RETURN = 1020;
static final int WHILE = 1021;

// If Expression <ifexpr>
static final int IF = 1030;
static final int ELSIF = 1031;
static final int ELSE = 1032;

// A variable that will contain lexemes as they are recognized:
private static String lexeme;

// This runs the scanner:
public static void main( String[] args ) throws Exception
{
  NanoMorpho lexer = new NanoMorpho(new FileReader(args[0]));
  int token = lexer.yylex();

  while(token != 0) {
    System.out.println(""+token+": \'"+lexeme+"\'");
    token = lexer.yylex();
  }
}

%}

/* Regular definitions */
_DIGIT=[0-9]
_FLOAT={_DIGIT}+\.{_DIGIT}+([eE][+-]?{_DIGIT}+)?
_INT={_DIGIT}+
_BOOL=(true|false)
_ESCAPE=\\b|\\t|\\n|\\f|\\r|\\\"|\\\'|\\\\|(\\[0-3][0-7][0-7])|(\\[0-7↲
↪    ][0-7])|(\\[0-7])
_CHAR=\'([^\'\\]|{_ESCAPE})\'
_STRING=\"([^\"\\]|{_ESCAPE})*\"
_DELIM=[()\{\},;=]
_NAME=([:letter:]|{_DIGIT}|_)+
_OPNAME=([\+\-*/!%=><\:\^\~&|?])
_OPNAMETWO=(\=\=|\!\=|&&|\|\|)

%%

/* Scanning rules */
```

```
{_DELIM} {
  lexeme = yytext();
  return yycharat(0);
}

{_STRING} | {_FLOAT} | {_CHAR} | {_INT} | {_BOOL} | null {
  lexeme = yytext();
  return LITERAL;
}

"var" {
  lexeme = yytext();
  return VAR;
}

"return" {
  lexeme = yytext();
  return RETURN;
}

"while" {
  lexeme = yytext();
  return WHILE;
}

"if" {
  lexeme = yytext();
  return IF;
}

"elsif" {
  lexeme = yytext();
  return ELSIF;
}

"else" {
  lexeme = yytext();
  return ELSE;
}

{_NAME} {
  lexeme = yytext();
  return NAME;
}

{_OPNAME} | {_OPNAMETWO} {
```

```
  lexeme = yytext();
  return OPNAME;
}

// EOL character
";;;".*$ {
}

// White spaces are ignored
[ \t\r\n\f] {
}

// If all rules fail, return an error
. {
  lexeme = yytext();
  return ERROR;
}
```

makefile

```
NanoMorpho.class: NanoMorpho.java
  javac NanoMorpho.java
NanoMorpho.java: nanomorpho.jflex
  java -jar jflex-full-1.7.0.jar nanomorpho.jflex

clean:
  rm -rf *~ NanoMorpho.class NanoMorpho.java

test: test1 test2 test3

test1:
  java NanoMorpho tests/test1.s | ../util/test_output.py

test2:
  java NanoMorpho tests/test2.s | ../util/test_output.py

test3:
  java NanoMorpho tests/test3.s | ../util/test_output.py
```

Test output

Test 1

```
main(i, j) {
  var r, s;
  var t = "Hallo";

  writeln("Hello World!");
}
```

Test 1

```
my_function(j, k) {

  if (j > k) {
    return j;
  }
  elsif (k > j) {
    return k;
  }
  else {
    return 0;
  }
}
```

Test 1

```
var bubbi_byggir = false;
if (!bubbi_byggir && bubbi_byggir == false) {
 writeln("Bubbi er ekki að byggja núna! Úps!");
};
var x = 5;
while (x==2 && x<-123 || (x==321 && x!=5)) {
 writeln("þetta gerist aldrei LOL! \b \t \n \r");
};
$@
```

Test_1

```
slowpoke  slowpoke-asus  ~/Desktop ... scanner (master) make test
java NanoMorpho tests/test1.s | ../util/test_output.py
----------------------------------------------------------------
1001: 'main'
40: '('
1001: 'i'
44: ','
1001: 'j'
41: ')'
123: '{'
1010: 'var'
1001: 'r'
44: ','
1001: 's'
59: ';'
1010: 'var'
1001: 't'
61: '='
1002: '"Hallo"'
59: ';'
1001: 'writeln'
40: '('
1002: '"Hello World!"'
41: ')'
59: ';'
125: '}'
```

Test_3

```
----------------------------------------------------------------
java NanoMorpho tests/test2.s | ../util/test_output.py
----------------------------------------------------------------
1001: 'my_function'
40: '('
1001: 'j'
44: ','
1001: 'k'
41: ')'
123: '{'
1030: 'if'
40: '('
1001: 'j'
1003: '>'
1001: 'k'
41: ')'
123: '{'
1020: 'return'
1001: 'j'
59: ';'
125: '}'
1031: 'elsif'
40: '('
1001: 'k'
1003: '>'
1001: 'j'
41: ')'
123: '{'
1020: 'return'
1001: 'k'
59: ';'
125: '}'
1032: 'else'
123: '{'
1020: 'return'
1002: '0'
59: ';'
125: '}'
125: '}'
```

Test_3

```
java NanoMorpho tests/test3.s | ../util/test_output.py
-----------------------------------------------------------------------------
1010: 'var'
1001: 'bubbi_byggir'
61: '='
1002: 'false'
59: ';'
1030: 'if'
40: '('
1003: '!'
1001: 'bubbi_byggir'
1003: '&&'
1001: 'bubbi_byggir'
1003: '=='
1002: 'false'
41: ')'
123: '{'
1001: 'writeln'
40: '('
1002: '"Bubbi er ekki að byggja núna! Úps!"'
41: ')'
59: ';'
125: '}'
59: ';'
1010: 'var'
1001: 'x'
61: '='
1002: '5'
59: ';'
1021: 'while'
40: '('
1001: 'x'
1003: '=='
1002: '2'
1003: '&&'
1001: 'x'
1003: '<'
1003: '-'
1002: '123'
1003: '||'
40: '('
1001: 'x'
1003: '=='
1002: '321'
1003: '&&'
1001: 'x'
1003: '!='
1002: '5'
41: ')'
41: ')'
123: '{'
1001: 'writeln'
40: '('
1002: '"þetta gerist aldrei LOL! \b \t \n \r"'
41: ')'
59: ';'
125: '}'
```

Test_3

```
1002: '"þetta gerist aldrei LOL! \b \t \n \r"'
41: ')'
59: ';'
125: '}'
59: ';'
-1: '$'
-1: '@'
-----------------------------------------------------------------------------
```

9