

Assignment 2

Algorithms: Correctness and Complexity

Due Nov 1, 2012

The work that you turn in for this assignment must represent your individual effort. You are welcome to help your fellow students understand the material of the course and the meaning of the assignment questions, however, the answer that you submit must be created by you alone.

Q0[10]. Consider the the following augmented grammar where Block' is the start symbol

$$\begin{aligned}
 \text{Block}' &\rightarrow \text{Block } \$ \\
 \text{Block} &\rightarrow \epsilon \\
 \text{Block} &\rightarrow \text{Comm Semicolon Block} \\
 \text{Semicolon} &\rightarrow \epsilon \\
 \text{Semicolon} &\rightarrow ; \\
 \text{Comm} &\rightarrow \mathbf{if} \text{ Exp } \mathbf{then} \text{ Block } \mathbf{else} \text{ Block } \mathbf{end} \mathbf{if} \\
 \text{Comm} &\rightarrow \mathbf{while} \text{ Exp } \mathbf{do} \text{ Block } \mathbf{end} \mathbf{while} \\
 \text{Comm} &\rightarrow \mathbf{id} := \text{Exp} \\
 \text{Comm} &\rightarrow \mathbf{print} \text{ PrintItems} \\
 \text{PrintItems} &\rightarrow \text{PrintItem MorePrintItems} \\
 \text{MorePrintItems} &\rightarrow \epsilon \\
 \text{MorePrintItems} &\rightarrow , \text{PrintItems} \\
 \text{PrintItem} &\rightarrow \text{Exp} \\
 \text{PrintItem} &\rightarrow \mathbf{string} \\
 \text{Exp} &\rightarrow \text{Term Exp1} \\
 \text{Exp1} &\rightarrow \epsilon \\
 \text{Exp1} &\rightarrow + \text{Term Exp1} \\
 \text{Exp1} &\rightarrow - \text{Term Exp1} \\
 \text{Term} &\rightarrow \text{Factor Term1} \\
 \text{Term1} &\rightarrow \epsilon \\
 \text{Term1} &\rightarrow * \text{Factor Term1} \\
 \text{Term1} &\rightarrow / \text{Factor Term1} \\
 \text{Factor} &\rightarrow \mathbf{num} \\
 \text{Factor} &\rightarrow \mathbf{id} \text{ Args} \\
 \text{Factor} &\rightarrow (\text{Exp}) \\
 \text{Args} &\rightarrow \epsilon \\
 \text{Args} &\rightarrow (\text{ArgList}) \\
 \text{ArgList} &\rightarrow \text{Exp MoreList} \\
 \text{MoreList} &\rightarrow \epsilon \\
 \text{MoreList} &\rightarrow , \text{ArgList}
 \end{aligned}$$

(a) [5] For a string of terminals of length at least 8, give a derivation of the string.

(b) [5 \leq] Draw a parse tree for your derivation in part (a)

Q1. [20] Compiler.

I have posted a partially completed compiler for the language of question 0. (Available for download from the course website.) This compiler uses the JavaCC parser generator for the lexical analysis and a hand built recursive descent parser. Your job is to complete the compiler by connecting the parser to the code generator and reporting on errors. Your compiler should report all lexical analysis errors, all syntax errors and the following context errors.

- Variable names must be a single lower case letter ‘a’ ,..., ‘z’ .
- Function names must be one of the following: sin, cos, tan, asin, acos, atan, ln, exp, pow. Each takes one argument except for pow, which takes two. (Function calls are distinguished from variable references by the presence of a nonempty argument list.)

Guards for if and while are interpreted as true if the expression is not 0 and false if the expression is 0.

The machine code for the compiler is defined by its implementation in class `Machine`. You can output instructions as they are being emitted if you turn on the machine’s `verbose` flag. You can trace the execution of the machine code using the machine’s `tracing` flag.

Submitting this assignment. Use D2L drop box. Submit your `VSPLParser.java` file along with any other files you changed.

Information on JavaCC. You can download JavaCC from

<http://java.net/projects/javacc>

If you use Eclipse, you can instead install the Eclipse plugin for JavaCC from

<http://eclipse-javacc.sourceforge.net/>

There is a tutorial on JavaCC at

<http://www.engr.mun.ca/~theo/JavaCC-Tutorial/>

(Sorry about the missing pictures.) There is a FAQ at

<http://www.engr.mun.ca/~theo/JavaCC-FAQ/>