Lehrstuhl für Informatik 2
Modellierung und Verifikation von Software

Compiler Construction SoSe 2012
Exercise 2 (Hand in until 07.05.2012)

AOR Priv.-Doz. Dr. Thomas Noll

Friedrich Gretz, Christina Jansen

Exercise 1 (Soundness of the NTA): (3 Points)

Prove soundness of the top down analysis automaton $NTA(G)$ for a grammar $G = \langle N, \Sigma, P, S\rangle$, i.e. show that for all $w \in \Sigma^*$ and all $z \in \{1, \ldots, |P|\}^*$:

$$(w, S, \varepsilon) \vdash^* (\varepsilon, \varepsilon, z) \quad \text{implies} \quad S \xRightarrow[l]{z} w$$

Exercise 2 (Nondeterministic Top-Down Parsing of WHILE): (2+1+1+2+1 Points)

In the lecture you have seen that context-free grammars are needed for the syntax analysis. Let's carry that over to our WHILE language.

a) Specify a context-free grammar $G = (N, \Sigma, P, S)$ that captures programs written in our WHILE-programming language.

Allow for the following:

- Statements are separated by a semicolon.
- Support at least: variable declarations (ints only), assignments, arithmetic operations, conditional branches, loops, basic I/O
- Assume your lexical analyser supports tokens:
  - ID for identifiers
  - NUM for integers
  - INT, WHILE, IF, READ, WRITE for keywords int, while, if, read and write
  - ASSGN for $=$ and SEM for ;
  - LBRAC, RBRAC, LCBRAC, RCBRAC for (,), {, }
  - PLUS, TIMES for $+$ and $\cdot$
  - LEQ, EQ, NEQ for $<=$, $==$, $!=$
  - etc.
- You may ignore comments (multi- as well as single-line).
- Programs should be bracketed correctly.
- Your grammar should be able to generate the tokens corresponding to following program: $x = 3 \cdot x - y;$

b) Give the syntax tree for the input: ID ASSGN NUM TIMES ID MINUS ID SEM ($x = 3 \cdot x - y;$).

c) Is G unambiguous? Prove your answer!

d) Provide the corresponding NTA for a subgrammar G' of G, where G' generates programs consisting of sequences of assignments to integer variables only.

e) Provide an accepting NTA-run for the input ID ASSGN NUM TIMES ID MINUS ID SEM ($x = 3 \cdot x - y;$) by giving the corresponding NTA configurations.