AOR Priv.-Doz. Dr. Thomas Noll

Friedrich Gretz, Christina Jansen

Exercise 1 (Deterministic LR(0) parsing automaton): (4 Points)

Consider the following grammar G:

$$\begin{array}{lcl} S & \rightarrow & B \\ B & \rightarrow & (B \wedge B) \mid \neg B \mid true \mid false \end{array}$$

a) Define the deterministic $LR(0)$ parsing automaton of G. (Remark: If you provide the action- and goto-table, example-transitions suffice!)

b) Parse the input $((\neg true \wedge false) \wedge false)$. Provide the corresponding run of your automaton from a).

Exercise 2 (Language Classes): (5 Points)

In the lecture you have seen different categorisations of CFGs. Here we want to have a closer look at the language classes $LL(0)$, $LL(1)$ and $LR(0)$ capture:

a) Describe the characteristics of grammars captured by $LL(0)$.

b) Provide a grammar that is captured by $LL(0)$, but not by $LL(1)$.

c) Show that the grammars captured by $LR(0)$ and $LL(1)$ are incomparable, but that their intersection is non-empty.

Exercise 3 (LALR(1) sets and conflicts): (4 Points)

Consider the following subgrammar G of *WHILE*:

S	→	expr
expr	→	NUM | ID | subexpr | LBRAC subexpr RBRAC
subexpr	→	expr PLUS expr | expr MINUS expr | expr TIMES expr | expr DIV expr

Is G an $LALR(1)$ grammar? Prove your answer!

Exercise 4 (Parser implementation): (12 Points)

Build an $LALR(1)$ parser for *WHILE*. Your program should perform a lexical analysis (see sheet 1 exercise 4) and if it succeeds check whether the input is a syntactically correct *WHILE* program. Note that the input could be faulty and make sure your implementation will catch this and report a parsing error. Test your implementation with various valid and invalid inputs!
Remark: You can construct the $LALR(1)$ parser "naively", i.e. you can compute the $LR(1)$ sets first and then merge equivalent ones. You do *not* have to implement more sophisticated methods like kernel representation!
Hint: You can download one implementation for the lexer from the course website and build on top of that if you wish.