

# Compiler Construction 2018/19

## — Exercise Sheet 5 —

Hand in until November 19th before the exercise class.

### Exercise 1

(4 Points)

(a) Show that for each context free grammar  $G$  it holds:

$$G \in LL(1) \implies G \text{ unambiguous}$$

(b) Prove or disprove: There exists an  $LL(0)$  grammar whose language is infinite.

### Exercise 2

(4 Points)

Let  $G = (N, \Sigma, P, A_1) \in CFG_\Sigma$  be a grammar with productions of the following form:

$$\begin{aligned} A_1 &\rightarrow A_2\alpha_1 \\ A_2 &\rightarrow A_3\alpha_2 \\ &\vdots \\ A_{n-1} &\rightarrow A_n\alpha_{n-1} \\ A_n &\rightarrow A_1\alpha_n \mid \beta \end{aligned}$$

where  $n > 1$ ,  $\alpha_i \in X^+$  for  $i = 1, \dots, n$  and  $\beta \in \Sigma^*$ . We assume  $G$  is a grammar without cycles ( $A_1 \Rightarrow^+ A_1$ ) and without  $\varepsilon$ -productions ( $A_1 \rightarrow \varepsilon$ ).

Let  $G'$  be the transformed grammar of  $G$  after applying Algorithm 8.6 to eliminate indirect left recursion. The algorithm iterates over  $N = \{A_1, \dots, A_n\}$ .

Prove:  $G' \notin LL(1)$ .

### Exercise 3

(4 Points)

Let  $G = (N, \Sigma, P, S) \in CFG_\Sigma$  be the context-free grammar given by

$$\begin{aligned} S &\rightarrow [expr] \\ expr &\rightarrow factor\ expr' \\ expr' &\rightarrow \text{or } expr \mid \text{and } expr \mid \varepsilon \\ factor &\rightarrow \text{not } factor \mid (expr) \mid \text{true} \mid \text{false} \end{aligned}$$

where  $N = \{S, expr, expr', factor\}$  and  $\Sigma = \{[, ], (, ), \text{or}, \text{and}, \text{not}, \text{true}, \text{false}\}$ .

Complete the following skeleton of a recursive descent parser for this grammar.

*Hint:* As in the lecture, you may assume the existence of a procedure `next()` that provides the next token (or reports an error if no next token exists). Moreover, you may assume procedures `print(int)` and `print(error)` to print your analysis and report errors, respectively.

```

1 proc main():
2     token := next(); S()
3 proc S():
4     // implement parser here
  
```