T-79.148 **Autumn 2003**
Introduction to Theoretical Computer Science
Tutorial 8
Problems

Homework problems:

1. Convert the following grammar into Chomsky normal form:

$$\begin{aligned} S &\to AB \mid c \\ A &\to T \mid aA \\ B &\to TT \mid \varepsilon \\ T &\to bS \end{aligned}$$

2. Determine, using the CYK algorithm ("dynamic programming method", Sipser p. 241, Lewis & Papadimitriou p. 155), whether the strings *abba*, *bbaa* and *bbaab* are generated by the grammar

$$\begin{aligned} S &\to AB \mid BA \mid a \mid b \\ A &\to BA \mid a \\ B &\to AB \mid b \end{aligned}$$

In the positive cases, give also the respective parse trees.

3. Design pushdown automata recognising the following languages:
   (a) $\{ww^R \mid w \in \{a,b\}^*\}$;
   (b) $\{w \in \{a,b\}^* \mid w \text{ has as many } a\text{'s as } b\text{'s}\}$

Demonstration problems:

4. Design an algorithm for testing whether a given a context-free grammar $G = (V, \Sigma, P, S)$, generates a nonempty language, i.e. whether any terminal string $x \in \Sigma^*$ can be derived from the start symbol S.

5. Design a pushdown automaton corresponding to the grammar $G = (V, \Sigma, P, S)$, where

$$\begin{aligned} V &= \{S, (,), {}^*, \cup, \emptyset, a, b\} \\ \Sigma &= \{(,), {}^*, \cup, \emptyset, a, b\} \\ P &= \{S \to (SS), S \to S^*, S \to (S \cup S), \\ &\quad S \to \emptyset, S \to a, S \to b\} \end{aligned}$$

6. Design a grammar corresponding to the pushdown automaton $M = (Q, \Sigma, \Gamma, \Delta, s, F)$, where

$$\begin{aligned} Q =& \{s, q, f\},\ \Sigma = \{a, b\},\ \Gamma = \{a, b, c\},\ F = \{f\}, \\ \Delta =& \big\{ \big((s, e, e), (q, c)\big), \big((q, a, c), (q, ac)\big), \big((q, a, a), (q, aa)\big) \\ & \big((q, a, b), (q, e)\big), \big((q, b, c), (q, bc)\big), \big((q, b, b), (q, bb)\big) \\ & \big((q, b, a), (q, e)\big), \big((q, e, c), (f, e)\big) \big\} \end{aligned}$$