T-79.148 **Autumn 2004**
Introduction to Theoretical Computer Science
Tutorial 8, 2–3 November
Problems

Homework problems:

1. Convert the following grammar for certain type of list structures,

$$\begin{array}{rcl} S & \rightarrow & (L) \mid a \\ L & \rightarrow & N \mid \varepsilon \\ N & \rightarrow & S, N \mid S \end{array}$$

into Chomsky normal form.

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

$$\begin{array}{rcl} S & \rightarrow & AB \mid BA \\ A & \rightarrow & BA \mid a \\ B & \rightarrow & AB \mid b \end{array}$$

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) the language described by the grammar in Problem 2 of Tutorial 7.

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

$$\begin{array}{l} V = \{S, (,), {}^*, \cup, \emptyset, a, b\} \\ \Sigma = \{(,), {}^*, \cup, \emptyset, a, b\} \\ P = \{S \rightarrow (SS), S \rightarrow S^*, S \rightarrow (S \cup S), \\ \qquad S \rightarrow \emptyset, S \rightarrow a, S \rightarrow b\} \end{array}$$

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

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