T-79.148 **Autumn 2002**

Introduction to Theoretical Computer Science

Tutorial 8, 6 – 8 November

Problems

Homework problems:

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

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

into Chomsky normal form.

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

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

In the positive case, give also the respective parse tree(s).

3. Design pushdown automata recognising the following languages:

   (a) $\{wcw^R \mid w \in \{a,b\}^*\}$;

   (b) $\{ww^R \mid w \in \{a,b\}^*\}$.

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

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

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

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