Homework problems:

1. Convert the following grammar into Chomsky normal form:

   \[
   S \rightarrow AB \mid c \\
   A \rightarrow T \mid aA \\
   B \rightarrow TT \mid \varepsilon \\
   T \rightarrow bS
   \]

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

   \[
   S \rightarrow AB \mid BA \mid a \mid b \\
   A \rightarrow BA \mid a \\
   B \rightarrow AB \mid b
   \]

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

3. Design pushdown automata recognising the following languages:

   (a) \( \{ w \in \{a, b\}^* \mid w = w^R\}\); 
   (b) The language generated by grammar

   \[
   S \rightarrow (S) \mid S, S \mid a
   \]

   (Cf. Tutorial 5, Problem 3.)

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

   \[
   V = \{S, (, ), *, \cup, \emptyset, a, b\} \\
   \Sigma = \{(, ), *, \cup, \emptyset, a, b\} \\
   P = \{S \rightarrow (SS), S \rightarrow S^*, S \rightarrow (S \cup S), S \rightarrow \emptyset, S \rightarrow a, S \rightarrow b\}
   \]

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

   \[
   Q = \{s, q, f\}, \Sigma = \{a, b\}, \Gamma = \{a, b, c\}, F = \{f\}, \\
   \Delta = \{(s, e, e), (q, c), ((q, a, e), (q, c, (q, ac)), ((q, a, a), (q, aa)) \\
          ((q, a, b), (q, e)), ((q, b, c), (q, bc)), ((q, b, b), (q, bb)) \\
          ((q, b, a), (q, c)), ((q, c, e), (f, e))\}
   \]