1. Let \( L \) be the language defined by the regular expression \((a \cup \epsilon)(ab \cup b)^*\).

   (a) Give a non-deterministic finite automaton that recognises the language \( L \).
   
   7 p.

   (b) Give a minimal deterministic finite automaton that recognises the language \( L \).
   
   8 p.

2. Consider the following grammar that produces parenthesis expressions.

   \[ S \rightarrow (S) \mid SS \mid \epsilon \]

   (a) Give a parse tree of the grammar for the string \(((())())\).
   
   4 p.

   (b) Show that the grammar is ambiguous.
   
   4 p.

   (c) Show (precisely!) that the language defined by the grammar is not regular.
   
   7 p.

3. Design a deterministic and single-tape Turing machine that recognises the language \( L = \{ w \in \{a, b\}^* \mid w = w^R \} \). Present your Turing machine as a state diagram. Show the accepting computation sequence of your machine on input \( aba \) and the rejecting computation sequence on input \( aab \).

   15 p.

4. Are the following claims true or false. Motivate your answer.

   (a) All recursive languages are finite (contain a finite number of strings).
   
   3 p.

   (b) The regular languages are closed with respect to complementation (i.e. if \( L \)
   is a regular language, then \( \overline{L} \) is also a regular language).
   
   4 p.

   (c) All languages recognised by a deterministic Turing machine are recursive.
   
   4 p.

   (d) The language \( L = \{a^k c^i b^k \mid i, k \geq 0 \} \) is context-free.
   
   4 p.

Total 60 p.

If you complete the feedback form of the course at http://www.cs.hut.fi/Opinnot/Palaute/kurssipalaute-en.html by May 20, 2005, you will be awarded one bonus exam point.