Helsinki University of Technology
Laboratory for Theoretical Computer Science
Harri Haanpää (tel. 5243), Tommi Syrjänen (tel. 5082)

T-79.148 Introduction to Theoretical Computer Science (2 cr)
Exam Mon 16 Feb 2004, 4 p.m. – 7 p.m.

Write down on each answer sheet:
- Your name, department, and study book number
- The text: "T-79.148 Introduction to Theoretical Computer Science 16.02.2004"
- The total number of answer sheets you are submitting for grading

1. Let the alphabet of the finite state automaton M be $\Sigma = \{a, b\}$. The transition function of M is described in Figure 1; the initial state is marked with $\rightarrow$ and accepting final states are marked with $\leftarrow$. The automaton M recognizes the language L.

   (a) Determine the minimal deterministic finite state automaton that recognizes the language L. *9 p.*

   (b) Present L as a regular expression. *6 p.*

	a	b
$\rightarrow$ A	B	E
B	C	F
$\leftarrow$ C	D	H
D	E	H
E	F	I
$\leftarrow$ F	G	B
G	H	B
H	I	C
$\leftarrow$ I	A	E

Figure 1: The finite state automaton M in tabular form

2. Let us define a *string of properly nested parentheses* inductively: ε is a string of properly nested parenthesis, and if x and y are strings of properly nested parenthesis, then so are `(`x`)`, `[`y`]`, and xy. For example, `([])[]` ja `[([])]` are strings of properly nested parenthesis, but `([]`, `[)` and `]()[` are not. Let L be the language of strings of properly nested parenthesis.

   (a) Prove in detail that L is not regular. *8 p.*

   (b) Design a context-free grammar that produces L. *8 p.*

   (c) Design a pushdown automaton that recognizes L. *9 p.*

3. (a) Define the concepts *recursive language* and *recursively enumerable language*. What is their most important difference? *5 p.*

   (b) Prove that if the language L is recursive, then so is the language

$$L^* = \bigcup_{k \geq 0} L^k = \{w_1 \dots w_k \mid k \geq 0, w_i \in L \text{ for all } 1 \leq i \leq k\}.$$

15 p.

Total 60 p.