T–79.1001/2 **Autumn 2007**

Introduction to Theoretical Computer Science T/Y
Tutorial 2, 20 to 25 September
Problems

Remember to enroll for the course using the TOPI registration system by 26 September. For bookkeeping reasons, registration is **compulsory***, even if you were not intending to attend the lectures or the tutorial sessions.*

Homework problems:

1. Design finite automata that recognise the following languages:

   (a) $\{w \in \{0, 1\}^* \mid w$ contains 00 as a substring$\}$;

   (b) $\{w \in \{0, 1\}^* \mid 00$ occurs exactly twice in w as a substring$\}$.

2. Design a finite automaton that models the behaviour of a lift moving between two storeys. The lift can be either up or down. Both storeys have a simple 'call here' button for the lift, and inside the lift there are buttons for going 'up' and 'down'. In addition, the lift has a door that can be opened and closed; the lift only moves when the door is closed. The time required for the lift to travel between the two storeys does not need to be taken into account, and any possible service requests occurring during this interval can be ignored. The automaton does not need to have any distinct "final states".

3. Design finite automata that recognise the following languages:

   (a) $\{w \in \{a, b\}^* \mid w$ starts with the substring $abba\}$;

   (b) $\{w \in \{a, b\}^* \mid w$ ends with the substring $abba\}$;

   (c) $\{w \in \{a, b\}^* \mid w$ contains $abba$ as a substring$\}$.

Demonstration problems:

4. Formulate the model of a simple coffee machine presented in class (lecture notes p. 17) precisely according to the mathematical definition of a finite automaton (Definition 2.1). What is the formal language recognised by this automaton?

5. Design finite automata that recognise the following languages:

   (a) $\{a^m b^n \mid m = n \mod 3\}$;

   (b) $\{w \in \{a, b\}^* \mid w$ contains equally many a's and b's, modulo 3$\}$.

   (The notation "$m = n \mod 3$" means that the numbers m and n yield the same remainder when divided by three.)

6. Design a finite automaton that recognises sequences of integers separated by plus and minus signs (e.g. 11+20-9, -5+8). Implement your automaton as a computer program that also calculates the numerical value of the input expression.