Instructions

- Classroom Problems C4.1–C4.3 will be discussed and solved onsite at the tutorial sessions in lecture week 4. No credit is given for these problems.
- Homework Problems H4.1–H4.3 you should solve on your own, and be available to present your solutions at one of the tutorial sessions in lecture week 5. In order to get course credit, you need to indicate your solved problems on the signup sheet circulated at the beginning of the session.
- Supplementary Problems S4.1–S4.3 provide further illustration and extension of the course material, but will usually not be covered at the tutorials. You are however invited to work on these problems too, and discuss them with the course staff. Sample solutions are provided on MyCourses.

Classroom Problems

C4.1 Give regular expressions describing the following languages:

(i) $\{w \in \{a, b\}^* \mid w \text{ contains } abb \text{ as a substring}\}$;

(ii) $\{w \in \{a, b\}^* \mid w \text{ contains either } abb \text{ or } bba \text{ (or both) as a substring}\}$;

(iii) $\{w \in \{0, 1\}^* \mid w \text{ contains exactly two 0's}\}$;

(iv) $\{w \in \{0, 1\}^* \mid w \text{ contains at least two 0's}\}$;

(v) $\{w \in \{0, 1\}^* \mid w \text{ contains an even number (possibly zero) of 0's}\}$;

(vi) $\{w \in \{0, 1\}^* \mid w \text{ begins and ends with different symbols}\}$;

C4.2 Following the guidelines presented in the lectures, design in a systematic way a deterministic finite automaton for the language described by the regular expression $(ba)^*b \cup (b^*a)$.

C4.3 Following the guidelines presented in the lectures, design in a systematic way, starting from a finite automaton, a regular expression for the language

$$\{w \in \{0, 1\}^* \mid \text{the number of 1's in } w \text{ is not a multiple of } 3\}.$$

Homework Problems

H4.1

(a) Give a regular expression that describes the language

$$\{w \in \{0,1\}^* \mid w \text{ contains 0110 or 1001 as a substring (possibly both)}\}.$$

(b) Following the guidelines presented in the lectures, design in a systematic way a deterministic finite automaton that recognises the language in part (a).

H4.2 Consider the following languages:

(a) $\{w \in \{a,b\}^* \mid w \text{ does not contain } aba \text{ as a substring}\}$;

(b) $\{w \in \{0,1\}^* \mid w \text{ contains an even number of both 0's and 1's}\}$.

In both cases, design a regular expression describing the language, by first constructing a finite automaton recognising it, and then converting the automaton in a systematic manner into the corresponding expression.

H4.3 Design (in outline) algorithms for determining whether the language described by a regular expression r over the alphabet $\{0,1\}$ is (a) empty, i.e. $L(r) = \emptyset$, (b) contains all possible binary strings, i.e. $L(r) = \{0,1\}^*$.

Supplementary Problems

S4.1 Simplify the following regular expressions, i.e. design simpler expressions describing the same languages:

1. $(\emptyset^* \cup a)(a^*)^*(b \cup a)b^*$
2. $(a \cup b)^* \cup \emptyset \cup (a \cup b)b^*a^*$
3. $a(b^* \cup a^*)(a^*b^*)^*$

S4.2 Determine whether the regular expressions $r_1 = b^*a(a^*b^*)^*$ and $r_2 = (a \cup b)^*a(a \cup b)^*$ are equivalent (i.e. describe the same language), by constructing the minimal deterministic finite automata corresponding to them.

S4.3 Prove that if L is a regular language, then so is the language $L' = \{xy \mid x \in L, y \notin L\}$.