Finite Automata and Formal Languages
TMV026/DIT321 – LP4 2010

Regular Expressions

Week 3 (and a bit of Week 4)

In these exercises, book sections, exercise numbers and pages refer to those in the third edition of the course book.

General hint: In some cases it may be easier to first compute a NFA, and then compute the regular expression from this NFA.

1. Take $\Sigma = \{a, b\}$. Give a regular expressions for the set of words containing an even number of a's and one for the set of words containing an odd number of a's.

   Compute a regular expression for strings with even length and one for strings whose length is a multiple of 3.

2. Simplify each of the following regular expressions:

   $\epsilon + ab + abab(ab)^*$

   $aa(b^* + a) + a(ab^* + aa)$

   $a(a + b)^* + aa(a + b)^* + aaa(a + b)^*$

3. Prove the following equalities:

   $b + ab^* + aa^*b + aa^*ab^* = a^*(b + ab^*)$

   $a^*(b + ab^*) = b + aa^*b^*$

4. Consider the regular sets denoted by the following pairs of regular expressions, with $\Sigma = \{a, b, c\}$. For each pair, say whether the two corresponding languages are equal. If not, give an example of a word in one that is not in the other.

   (a) $(a + b)^*$ and $a^* + b^*$

   (b) $a(bca)^*bc$ and $ab(cab)^*c$

   (c) $\emptyset^*$ and ϵ^*

   (d) $(a^*b^*)^*$ and $(a^*b)^*$

   (e) $(ab + a)^*a$ and $a(ba + a)^*$

5. Take $\Sigma = \{a, b\}$. Give a regular expression for the strings that do not contain the substring aa.

6. Do exercises 3.1.1, 3.1.2, 3.1.4 and 3.1.5.

7. Do exercises 3.4.1, 3.4.2 and 3.4.3.

8. Do exercises 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5 and 3.2.6.