Software-Modellierung und Verifikation
Informatik 2
Prof. J.-P. Katoen
RWTH Aachen

Priv.-Doz. T. Noll noll@cs.rwth-aachen.de
D. Klink klink@cs.rwth-aachen.de

1. Exercise sheet *Compiler Construction 2008*

Due to Wed., 23 April 2008, *before* the exercise course begins.

Hand in your solutions in groups of three!

Exercise 1.1:

In this exercise we focus on regular expressions (regex) over the alphabet of all words in $\{a, \ldots, z\}^*$ that end with a or b. E.g. $bla \in \mathcal{L}_{\text{word}} =: \Omega$ but $bah \notin \mathcal{L}_{\text{word}}$. Furthermore, $bla \cdot (blub^* + bla^* \cdot blub) \in \mathcal{L}_{\text{regex}}$, the set of regular expressions (as described above), but $bla\ blub \notin \mathcal{L}_{\text{regex}}$ as we require all operators to be explicitly stated and regular expressions shall not contain any white spaces.

a) Give regular expressions and languages for all symbol classes that will be needed for a lexical analysis of elements of $\mathcal{L}_{\text{regex}}$. Denote the languages of each symbol class by the associated token, e.g. for tokens lambda, concat, word, ... write $\mathcal{L}_{\text{lambda}}, \mathcal{L}_{\text{concat}}, \mathcal{L}_{\text{word}}, \ldots$.

b) Give the seqence of lexemes for $bla \cdot (blub^* + bla^* \cdot blub)$ and translate each lexeme into a symbol.

c) Devise an automaton that accepts $w \in \mathcal{L}_{\text{tok}}$ in state q_{tok} for all tok $\in \{\text{lambda}, \text{concat}, \text{word}, \ldots\})$. Give an accepting run for bla. (Note that $bla + blub$ is a regular expression but it is in none of the symbol class languages.)

d) Give a grammar describing all *token sequences* that represent regular expressions. Find a derivation of the *token sequence* from b). (A token sequence is the projection of a sequence of symbols to the tokens.)

Exercise 1.2:

Extend regular expressions by a complement operator, i.e. if α is a regular expression so is $\neg\alpha$. The semantics is given by $[\![\neg\alpha]\!] := \Omega^* \setminus [\![\alpha]\!]$. Let $\Omega = \{a, \ldots, z\}$ and $\alpha = moo^*$.

a) Give an NFA for regular expression α.

b) Construct the complement automaton $\mathfrak{A}(\neg\alpha)$, i.e. $\mathcal{L}(\mathfrak{A}(\neg\alpha)) = [\![\neg\alpha]\!]$, in a way such that it can be used for the Thompson construction.

c) Construct an NFA $\mathfrak{A}(\neg\alpha \cdot h)$ via Thompson construction.

d) Check whether the NFA from c) accepts *moodoh* and *mooh*.