Prof. Javier Esparza
Philipp Czerner

Technical University of Munich
Chair for Foundations of Software Reliability

Automata and Formal Languages

Winter Term 2023/24 – Exercise Sheet 2

Exercise 2.1.

Consider the regular expression $r = (a + ab)^*$.

(a) Convert r into an equivalent NFA-ε A.

(b) Convert A into an equivalent NFA B. (It is not necessary to use algorithm *NFAεtoNFA*)

(c) Convert B into an equivalent DFA C.

(d) By inspecting B, give an equivalent minimal DFA D. (No algorithm needed).

(e) Convert D into an equivalent regular expression r'.

(f) Prove formally that $L(r) = L(r')$.

Exercise 2.2.

Prove or disprove the following.

(a) If L_1 and $L_1 \cup L_2$ are regular, then L_2 is regular.

(b) If L_1 and $L_1 \cap L_2$ are regular, then L_2 is regular.

(c) If L_1 and $L_1 L_2$ are regular, then L_2 is regular.

(d) If L^* is regular, then L is regular.

Exercise 2.3.

Recall that a nondeterministic automaton A accepts a word w if at least one of the runs of A on w is accepting. This is sometimes called the *existential* accepting condition. Consider the variant in which A accepts w if *all* runs of A on w are accepting (in particular, if A has no run on w then it accepts w). This is called the *universal* accepting condition and such automata will be referred to as a co-NFA.

Intuitively, we can visualize a co-NFA as executing all runs in parallel. After reading a word w, the automaton is simultaneously in all states reached by all runs labelled by w, and accepts if all those states are accepting.

(a) Suppose A_1 and A_2 are two co-NFA which accept languages L_1 and L_2 respectively. Let n_1 and n_2 be the number of states of A_1 and A_2 respectively. Show that there is a co-NFA B over $n_1 + n_2$ states which accepts $L_1 \cap L_2$.

(b) Give an algorithm that transforms a co-NFA into a DFA recognizing the same language. This shows that automata with universal accepting condition recognize the regular languages.

Let $\Sigma = \{a, b\}$. Given a word $w = a_1 a_2 ... a_n$ where each $a_i \in \Sigma$, let $w^R = a_n a_{n-1} ... a_1$ denote the *reverse* of w. For any $n \in \mathbb{N}$, consider the language $L_n := \{ww^R \in \Sigma^{2n} \mid w \in \Sigma^n\}$.

(c) Give a co-NFA with $O(n^2)$ states that recognizes L_n.

(d) Prove that every NFA (and hence also every DFA) recognizing L_n has at least 2^n states.

Puzzle exercise 2.4.

Let Σ denote an alphabet and let $L \subseteq (\Sigma^3)^*$ denote a regular language where the length of each word is a multiple of three. Prove or disprove that $L' := \{w_1 ... w_n\, w_{2n+1} ... w_{3n} : w_1 ... w_{3n} \in L\}$ is regular.

Notes: The puzzle exercises cover advanced material and are not directly relevant to the exam. No model solutions will be provided. You can get feedback on your solution by sending it to me (Zulip or mail), or coming to my office in person (MI 03.11.037).