Universität Duisburg-Essen
Faculty of Engineering
Department of Computer Science and Applied Cognitive Science
Theoretical Computer Science Group
Prof. Dr. Barbara König

Exercise sheet 8

Automaten und Formale Sprachen

Sommersemester 2019, Teaching assistant: Dennis Nolte, Lara Stoltenow

Submission[1]: Wednesday, June 12, 2019[2], 10:00 Uhr

Exercise 24 *Pumping Lemma for advanced learners* (5 points)

Let $\Sigma = \{a, b\}$. Prove by means of the Pumping Lemma for regular languages, that the following languages are not regular:

(a) $L_1 = \{a^{m^2} \mid m \in \mathbb{N}_0\}$ (2.5 p)

(b) $L_2 = \{a^n b a^{2n} \mid n \in \mathbb{N}_0\}$ (2.5 p)

Note: It is *not sufficient* to just give what has to be filled into the holes of the cloze text. You have to write down the entire proof!

[1] *Options to submit your solutions:* Letterbox next to LF 259 (Campus Duisburg) or via Moodle `https://moodle.uni-due.de/course/view.php?id=15777`

[2] Only this time because of holidays

Exercise 25 *Equivalence of Words* (6 points)

In this exercise the equivalence of words according to the Myhill-Nerode equivalence shall be examined. Justify the correctness of your answers!

(a) Let the language $L_1 = \{(abc)^n \mid n \in \mathbb{N}_0\}$ over the alphabet $\Sigma = \{a, b, c\}$ be given. In the following we use the Myhill-Nerode equivalence $\equiv_{L_1}$.

   (i) Give one equivalent word for each of the following words: a, c and abc. Also give a word which is not equivalent to any of these three words. (2 p)

   (ii) Write down the equivalence class which contains the word a and the equivalence class which contains the word abc in set notation. (1 p)

(b) Let the language $L_2 = \{a^n b^m \mid n, m \in \mathbb{N}_0 \text{ and } n + m \text{ is even}\}$ over the alphabet $\Sigma = \{a, b\}$ be given. In the following we use the Myhill-Nerode equivalence $\equiv_{L_2}$.

   (i) Give one equivalent word for each of the following words: ab, b and a. Also give a word which is not equivalent to any of these three words. (2 p)

   (ii) Write down the equivalence class which contains the word ab and the equivalence class which contains the word a in set notation. (1 p)

Note: Do *not* confuse Myhill-Nerode equivalence classes with the language of a state as discussed in Exercise 23! (Given an automaton, equivalence classes are sets of words that lead *to* a set of acceptance-equivalent states, while the language of a state is a set of words that lead *from* some state to a final state.)

Exercise 26 *Models for regular languages (former exam task)* (6 points)

Let the following regular language over the alphabet $\Sigma = \{a, b, c\}$ be given:

$$L = \{wv \in \Sigma^* \mid w \text{ contains exactly one } a \text{ and } v \text{ contains exactly one } b\}$$

(a) Give a regular grammar that generates L. (2 p)

(b) Give a finite automaton which accepts L. (2 p)

(c) Give a regular expression which generates L. (1 p)

(d) Is the language L finite? (1 p)

Exercise 27 *Closure properties and language classification* (3 points)

Let $\Sigma = \{a, b\}$. We give the language

$$L = \{w \in \Sigma^* \mid \#_a(w) = \#_b(w)\}$$

Argue, using the closure properties of regular languages, why L cannot be regular. You may use the fact that $\{a^n b^n \mid n \in \mathbb{N}_0\}$ is not regular.

(In total, there are **20** points in this exercise sheet.)