Automata & Formal Languages
Michaelmas Term 2023
Part II of the Mathematical Tripos
University of Cambridge
Prof. Dr. B. Löwe

Example Sheet #1

(1) Let $R = (\Omega, P)$ be a rewrite system and $\alpha, \beta, \gamma \in \Omega^*$. Show that if $\alpha \xrightarrow{R} \beta$, then $\alpha\gamma \xrightarrow{R} \beta\gamma$. Show that the converse does not hold in general. What properties of P could guarantee that the converse holds?

(2) Let Ω be a finite non-empty set of symbols. Give examples of rewrite systems $R = (\Omega, P)$ and $\alpha, \beta \in \Omega^*$ with the following properties:

(i) There is a unique derivation for $\beta \in \mathcal{D}(R, \alpha)$.
(ii) There are exactly two derivations for $\beta \in \mathcal{D}(R, \alpha)$.
(iii) There are infinitely many derivations for $\beta \in \mathcal{D}(R, \alpha)$.

(3) Let $\Sigma := \{a, b\}$ and let $G = (\Sigma, V, P, S)$ and $G' = (\Sigma, V', P', S')$ be grammars. For each of the questions, give an example or argue that it is impossible to do so.

(i) Is it possible that G and G' are equivalent, but $\mathcal{D}(G, S) \neq \mathcal{D}(G', S')$?
(ii) Is it possible that $\mathcal{D}(G, S) = \mathcal{D}(G', S')$, but G and G' are not equivalent?
(iii) Is it possible that G and G' are equivalent, but not isomorphic?
(iv) Is it possible that G and G' are isomorphic, but $\mathcal{D}(G, S) \neq \mathcal{D}(G', S')$?
(v) Is it possible that G and G' are not equivalent, but $\mathcal{L}(G) \cap \{a\}^* = \mathcal{L}(G') \cap \{a\}^*$?

(4) Let $\Sigma = \{a, b, c\}$. Show each of the following claims by producing an appropriate grammar that produces the given language. Explain why your grammar generates this language.

(i) The language consisting of words of the form $(abc)^n$ (for $n > 0$) is type 3.
(ii) The language consisting of words of the form $a^n bc^n$ (for $n \in \mathbb{N}$) is type 2.
(iii) The language consisting of words of the form $a^n b^n c^n$ (for $n > 0$) is type 1.

Are any of them of even higher type than listed?

(5) Let $G = (\Sigma, V, P, S)$ be any grammar. As in the lectures, a production rule $\alpha \to \beta$ is called *variable-based* if $\alpha \in V^*$. Suppose that $\alpha \to \beta$ is a noncontracting variable-based rule, say with $\alpha = A_1 ... A_n$ and $\beta = B_1 ... B_m$ for $A_i \in V$, $B_i \in \Omega$, and $n \leq m$. Let $X_1, ..., X_n$ be n new

variables that do not occur in V and consider the following list of $2n$ rules:

$$
\begin{array}{lcl}
A_1A_2A_3 \;\ldots\; A_{n-2}A_{n-1}A_n & \to & X_1A_2A_3 \;\ldots\; A_{n-2}A_{n-1}A_n \\
X_1A_2A_3 \;\ldots\; A_{n-2}A_{n-1}A_n & \to & X_1X_2A_3 \;\ldots\; A_{n-2}A_{n-1}A_n \\
X_1X_2A_3 \;\ldots\; A_{n-2}A_{n-1}A_n & \to & X_1X_2X_3 \;\ldots\; A_{n-2}A_{n-1}A_n \\
 & \vdots & \\
X_1X_2X_3 \;\ldots\; X_{n-2}A_{n-1}A_n & \to & X_1X_2X_3 \;\ldots\; X_{n-2}X_{n-1}A_n \\
X_1X_2X_3 \;\ldots\; X_{n-2}X_{n-1}A_n & \to & X_1X_2X_3 \;\ldots\; X_{n-2}X_{n-1}X_nB_{n+1} \;\ldots\; B_m \\
X_1X_2X_3 \;\ldots\; X_{n-2}X_{n-1}X_nB_{n+1} \;\ldots\; B_m & \to & B_1X_2X_3 \;\ldots\; X_{n-2}X_{n-1}X_nB_{n+1} \;\ldots\; B_m \\
B_1X_2X_3 \;\ldots\; X_{n-2}X_{n-1}X_nB_{n+1} \;\ldots\; B_m & \to & B_1B_2X_3 \;\ldots\; X_{n-2}X_{n-1}X_nB_{n+1} \;\ldots\; B_m \\
 & \vdots & \\
B_1B_2B_3 \;\ldots\; B_{n-2}X_{n-1}X_nB_{n+1} \;\ldots\; B_m & \to & B_1B_2B_3 \;\ldots\; B_{n-2}B_{n-1}X_nB_{n+1} \;\ldots\; B_m \\
B_1B_2B_3 \;\ldots\; B_{n-2}B_{n-1}X_nB_{n+1} \;\ldots\; B_m & \to & B_1B_2B_3 \;\ldots\; B_{n-2}B_{n-1}B_nB_{n+1} \;\ldots\; B_m
\end{array}
$$

Show that each of these rules is context-sensitive and that replacing $\alpha \to \beta$ in P by this collection of $2n$ rules does not change the language produced by G. Use this to prove that a language is noncontracting if and only if it is context-sensitive.

(6) Give an example of a class of languages that is closed under unions and intersections, but not under complementation.

(7) In the lectures, we proved that the *concatenation grammar* of two grammars G and G' produces the concatenation of the two languages produced by G and G' *under the assumption that they are variable-based and do not share any variables.* Show that these assumptions are necessary by giving grammars G and G' such that the language produced by the concatenation grammar is not $\mathcal{L}(G)\mathcal{L}(G')$. Is the same true for the union grammar construction?

(8) Construct deterministic automata by drawing transition diagrams which accept the following languages. Explain your answers.

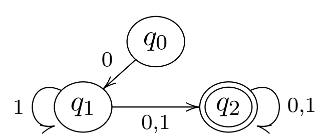
(a) $\{w \in \{0,1\}^* \,;\, |w| > 2\}$;

(b) $\{w \in \{0,1\}^* \,;\, w$ is a nonempty alternating sequence of 0s and 1s$\}$;

(c) $\{w \in \{0,1\}^* \,;\, w$ is a multiple of 3 when interpreted in binary$\}$;

(d) $\{w \in \{0,1\}^* \,;\, w$ contains 01010 as a substring$\}$.

(9) Consider the following nondeterministic automaton over the alphabet $\Sigma = \{0,1\}$:



Convert it to a deterministic automaton with $2^3 = 8$ states using the power set construction. Can you simplify the deterministic automaton without changing the accepted language?

(10) For each of the following languages $L \subseteq \{0,1\}^*$, determine whether or not they are regular. Justify your answers.

(a) $\{0^n 1^{2n}\,;\, n \geq 1\}$;

(b) $\{ww\,;\, \varepsilon \neq w \in \{0,1\}^*\}$;

(c) $\{w1w\,;\, w \in \{0\}^*\}$;

(d) $\{v1w\,;\, v, w \in \{0\}^*\}$;

(e) $\{0^n 1^m\,;\, n > m\}$;

(f) $\{0^n 1^m\,;\, n \neq m\}$;

(g) $\{0^n 1^m\,;\, n \geq m \text{ and } m \leq 1000\}$;

(h) $\{0^n 1^m\,;\, n \geq m \text{ and } m \geq 1000\}$;

(i) $\{1^p\,;\, p \text{ is a prime}\}$.

(11) Let $L \subseteq \{0^n 1^n\,;\, n \geq 1\}$. Show that L is regular if and only if L is finite.

(12) Suppose that $G = (\Sigma, V, P, S)$ is a regular grammar with $|V| = n$. Prove that if G produces a word of length at least 2^{n+1}, then it produces infinitely many words.

(13) In the lectures, we have seen two different constructions that prove that the class of regular languages is closed under unions: the union grammar and the product automaton construction. Take two grammars G and G' and form deterministic automata D and D', using the two mentioned constructions such that $\mathcal{L}(D) = \mathcal{L}(G) \cup \mathcal{L}(G') = \mathcal{L}(D')$. Compare the number of states that the automata D and D' have.