Homework I, Theory of Computation 2026

Submission: **The deadline for Homework 1 is 23:59 on March 15th.** Please submit your solutions on Moodle. Typing your solutions using a typesetting system such as LaTeX is strongly encouraged! If you must handwrite your solutions, write cleanly and with a pen. Messy and unreadable homeworks will not be graded. No late homeworks will be accepted.

Writing: Please be precise, concise and (reasonably) formal. Keep in mind that many of the problems ask you to provide a proof of a statement (as opposed to, say, just to provide an example). Therefore, make sure that your reasoning is correct and there are no holes in it. A solution that is hard/impossible to decipher/follow might not get full credit (even if it is in principle correct). You do not need to reprove anything that was shown in the class—just state clearly what was proved and where.

Collaboration: These problem sets are meant to be worked on in groups of 2–4 students. Please submit only one writeup per team—it should contain the names of all the students. You are strongly encouraged to solve these problems by yourself. If you must, you may use books or online resources to help solve homework problems, but you must credit all such sources in your writeup and you must never copy material verbatim. Even though only one writeup is submitted, it is expected that each one of the team members is able to fully explain the solutions if requested to do so.

Grading: Each of the two problems will be graded on a scale from 0 to 5.

Warning: Your attention is drawn to the EPFL policy on academic dishonesty. In particular, you should be aware that copying solutions, in whole or in part, from other students in the class or any other source (e.g., ChatGPT) without acknowledgment constitutes cheating. Any student found to be cheating risks automatically failing the class and being referred to the appropriate office.

Homework 1

1 For a word $w \in \Sigma^*$ and an integer k with $0 \le k \le |w|$, write $w = xy$ where $|x| = k$, and define the rotation of w by k positions as

$$\mathsf{rot}_k(w) = yx.$$

For a language $L \subseteq \Sigma^*$, define $\mathsf{Rot}(L) = \{ \mathsf{rot}_k(w) \mid w \in L,\ 0 \le k \le |w| \}$. Show that if L is regular, then $\mathsf{Rot}(L)$ is also regular.

2 Let Σ be a finite alphabet. For a language $L \subseteq \Sigma^*$, define

$$E_1(L) = \{ w \in \Sigma^* \mid \exists x \in L : \ d_{\text{edit}}(w, x) \le 1 \},$$

where $d_{\text{edit}}(w, x)$ is the minimum number of unit-cost operations needed to transform w into x: substitution of one symbol, insertion of one symbol, and deletion of one symbol.

2a Suppose L is recognized by a DFA $M = (Q, \Sigma, \delta, q_0, F)$. Construct an NFA N recognizing $E_1(L)$ with *exactly* $2|Q|$ *states.* (Your construction may use ε-transitions.)

2b Assume $\Sigma = \{0, 1\}$ and

$$L = \{ w \in \{0,1\}^* \mid w \text{ ends with } 11 \}.$$

1. Give a DFA M for L with exactly 3 states.
2. Build the 6-state NFA N for $E_1(L)$ using your construction from Part 2a.
3. Convert N to an equivalent DFA: first remove the ε-transitions, then write down the subset construction and remove any states that are not reachable.

2c Consider the following:

> **Regular expression conventions (over $\Sigma = \{0, 1\}$).** A regular expression R is built from the atoms $0, 1, \varepsilon$ (empty string), and $\emptyset$ (empty language), using: union ($R_1 \cup R_2$), concatenation ($R_1 R_2$), and the star operation $(R_1)^\star$ (repetition). (The superscript * denotes repetition, not numerical exponentiation.) Parentheses may be used for grouping. Precedence is: * highest, then concatenation, then $\cup$. For a regular expression R, $L(R)$ denotes the language described by R.

For the language L from Part 2b, give a regular expression R for $E_1(L)$, and prove that $L(R) = E_1(L)$.