Solved exercises must be marked and solutions (as a single PDF file) uploaded in OLAT. The (strict) deadline is 7 am on January 30.

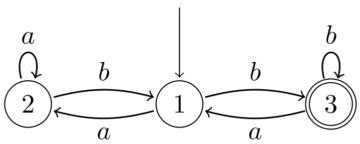
Exercises

⟨2⟩ 1. Determine whether the following sets over $\Sigma = \{a, b\}$ are regular or not. Prove your answers.

(a) $\{xy \mid x, y \in \Sigma^* \text{ with } x \neq y \text{ and } |y| > 0\}$

(b) $\{xabx \mid x \in \Sigma^*\}$

⟨2⟩ 2. Construct a WSMO formula φ such that $L(\varphi) = L(M)$ for the following DFA M:



⟨3⟩ 3. Consider the Presburger arithmetic formula $\varphi = \neg \exists\, y.\, x - 3y = 1$.

(a) Which of the following strings belong to $L(\varphi)$?

i. 00011 ii. 10101 iii. 11001

(b) Construct a finite automaton that accepts $L(x - 3y = 1)$

(c) Construct a finite automaton that accepts $L(\varphi)$.

⟨3⟩ 4. Consider the set $A = \{x \in \{a, b\}^\omega \mid \text{every } a \text{ in } x \text{ is immediately followed by a } b\}$.

(a) Give an LTL formula φ that defines A.

(b) Use the procedure from the lecture to construct an alternating Büchi automaton M that accepts $L(\varphi)$.

(c) Use the procedure given by the proof on slide 27 of lecture 13 to construct a nondeterministic Büchi automaton from M.

Bonus Exercise

⟨5⟩ 5. This goal of the final (bonus) exercise is to implement a simplified LTL-based grammar checker that processes text input and flags violations of predefined grammatical rules. The grammar checker should verify the following three rules:

φ_1 The e symbol can never occur without a preceding number.

φ_2 If an auxiliary verb (AUX) occurs, it must eventually be followed by a past participle (PART) before the next end-of-sentence (EOS).

φ_3 A finite verb (VERB) cannot immediately follow an auxiliary verb (AUX) without a participle or an end-of-sentence occurring first.

For the sake of simplicity, we focus on a limited set of tokens and rules. We model a document as a sequence of tokens. Each token is represented by exactly one symbol from the alphabet Σ defined as follows:

$$\Sigma = \{\mathsf{NUMBER}, \mathsf{EUR}, \mathsf{AUX}, \mathsf{PART}, \mathsf{VERB}, \mathsf{EOS}\}$$

The task is to implement this grammar checker using LTL formulas and automata in a language of your choice. You may hardcode the mapping of words to tokens in your implementation (e.g., $1 \to \mathsf{NUMBER}$, €$\to \mathsf{EUR}$, *have* $\to \mathsf{AUX}$, *eaten* $\to \mathsf{PART}$, *apple* $\to \mathsf{VERB}$, . $\to \mathsf{EOS}$).

(a) Formulate the rules φ_1, φ_2 and φ_3 as LTL formulas over the alphabet Σ.

(b) Combine the LTL formulas into a Büchi automaton that accepts exactly those sequences of tokens that satisfy all three grammatical rules (by using and implementing the procedures from the lecture or using available libraries).

(c) Write a procedure that tests whether an input token sequence satisfies all grammar rules. If a violation occurs, the program should report which rule is violated.