Introduction to Language Theory and Compilation
Exercises
Session 1: Regular languages

Reminders

Languages and operations

Let Σ be a (finite) alphabet. A *language* is a set of *words* defined on a given alphabet. Let L, L_1 and L_2 be languages, we can then define some operations:

Definition 1 (Union)**.** $L_1 \cup L_2 = \{w \mid w \in L_1 \textit{ or } w \in L_2\}$

Definition 2 (Concatenation)**.** $L_1 \cdot L_2 = \{w_1 w_2 \mid w_1 \in L_1 \textit{ and } w_2 \in L_2\}$

Definition 3 (Kleene closure)**.** $L^* = \{\varepsilon\} \cup \{w \mid w \in L\} \cup \{w_1 w_2 \mid w_1, w_2 \in L\} \cup \cdots$

Regular languages

Regular languages are defined inductively:

Definition 4 (Regular language)**.**

- $\emptyset$ *is a regular language*
- $\{\varepsilon\}$ *is a regular language*
- *For all* $a \in \Sigma$, $\{a\}$ *is a regular language*

If L, L_1, L_2 *are regular languages, then:*

- $L_1 \cup L_2$ *is a regular language*
- $L_1 \cdot L_2$ *is a regular language*
- L^* *is a regular language*

Finite automata (FA)

$M = \langle Q, \Sigma, \delta, q_0, F \rangle$ where:

- Q is a *finite* set of *states*
- Σ is the *input alphabet*
- δ is the *transition function*
- $q_0 \in Q$ is the *start state*
- $F \subseteq Q$ is the set of *accepting states*

M is a *deterministic* finite automaton (DFA) if the transition function $\delta : Q \times \Sigma \to Q$ is total. In other words, on each input, there is *one and only one* state to which the automaton can transition from its state.

Determinisation

The transition function can be extended to sets of states as follows: for $S \subseteq Q, \delta(S,a) = \bigcup_{s \in S} \delta(s,a)$. The ε-closure is defined as $\varepsilon\text{closure}(q) = \left\{ q' \in Q \mid \exists n \in \mathbb{N}, \exists q_1 \ldots q_n \in Q, q \xrightarrow{\varepsilon} q_1 \xrightarrow{\varepsilon} \ldots \xrightarrow{\varepsilon} q_n \xrightarrow{\varepsilon} q' \right\}$.

For $M = \langle Q, \Sigma, \delta, q_0, F \rangle$, the DFA $D = \langle Q^D, \Sigma, \delta^D, q_0^D, F^D \rangle$, where:

- $Q^D = 2^Q$
- $q_0^D = \varepsilon\text{closure}(q_0)$
- $F^D = \{ S \in Q^D \mid S \cap F \neq \emptyset \}$
- For all $S \in Q^D$, for all $a \in \Sigma$, $\delta^D(S,a) = \varepsilon\text{closure}(\delta(S,a))$

is such that $L(D) = L(M)$.

Exercises

Ex. 1. Consider the alphabet $\Sigma = \{0,1\}$. Using the inductive definition of regular languages, prove that the following languages are regular:

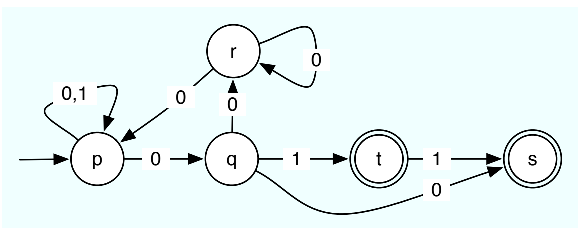
1. The set of words made of an arbitrary number of ones, followed by 01, followed by an arbitrary number of zeroes.
2. The set of odd binary numbers.

Ex. 2. Prove that any finite language is regular. Is the language $L = \{0^n 1^n \mid n \in \mathbb{N}\}$ regular? Explain.

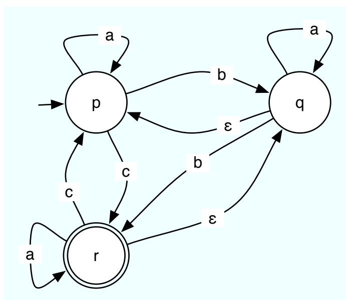
Ex. 3. For each of the following languages (defined on the alphabet $\Sigma = \{0,1\}$), design a nondeterministic finite automaton (NFA) that accepts it.

1. The set of strings ending with 00.
2. The set of strings whose 10^{th} symbol, counted from the end of the string, is a 1.
3. The set of strings where each pair of zeroes is followed by a pair of ones.
4. The set of strings not containing 101.
5. The set of binary numbers divisible by 4.

Ex. 4. Transform the following (ε-)NFAs into DFAs:



(a)



(b)