17 Finite words

↳ First connection between automata and logic
↳ Basic constructions

1. Regular languages and finite automata

↳ Recapitulation, notation, problems

1.1 Regular languages

↳ Basic notions
↳ Constructions from fgdp.

Notions:

- Finite alphabet = finite set $\Sigma = \{a, b, \ldots\}$
- Finite word = sequence $w = a_0 \ldots a_{n-1}$ with $a_i \in \Sigma$
- Length of w, $|w| = n$
- Empty word ε with length 0
- ith symbol $w(i) = a_i$
- Σ^* set of all finite words over Σ, $\Sigma^+ := \Sigma^* \setminus \{\varepsilon\}$ non-empty words
- Concatenation of $w, v \in \Sigma^*$ is $w.v \in \Sigma^*$
- Language $L \subseteq \Sigma^*$, typically infinite

↳ With this definition, all set-theoretic operations also apply to languages.

$L_1 \cup L_2$ (union), $L_1 \cap L_2$ (intersection), $L_1 \setminus L_2$ (difference), $\overline{L_1} := \Sigma^* \setminus L_1$ (complement)

↳ Concatenation:

$$L_1.L_2 := \{w.v \mid w \in L_1 \text{ and } v \in L_2\}$$

↳ Kleene star:

$$L^* := \bigcup_{i \in \mathbb{N}} L^i \text{ with } L^0 := \{\varepsilon\},\ L^{i+1} := L.L^i$$

(finitely many concatenations with words in L)

$= \{w_1 \dots w_n \mid n \in \underset{\{0,1,2,\dots\}}{\overset{\shortparallel}{\mathbb{N}}} \text{ and } w_i \in L \text{ f.a. } i\}.$

Definition.

The class of regular languages over alphabet Σ is denoted by REG_Σ. It is the smallest class that satisfies

(1) $\emptyset \in REG_\Sigma$ and $\{a\} \in REG_\Sigma$ f.a. $a \in \Sigma$

(2) $L_1, L_2 \in REG_\Sigma$ implies $L_1 \cup L_2 \in REG_\Sigma$,
$L_1 . L_2 \in REG_\Sigma$,
$L_1^* \in REG_\Sigma$.

Every regular language is obtained by application of finitely many operations in (2) from (1).

Notation:

↳ Brackets: $*$ stronger than $.$ stronger than $\cup$

↳ $\{a\}$ as a (singleton sets)

Example:

$\varepsilon \cup (a \cup b)^* . b$

Observation:

↳ Every finite set of words forms a regular language

↳ Regular languages are not closed under infinite unions (this gives all (finite word) languages)

↳ By definition, REG closed under $\cup$, $.$, $*$

Show:

REG is also closed under remaining set operations:

$\cap$, $\overline{}$, $\setminus$, Δ

Not clear from definition. Note: $L_1 \setminus L_2 := L_1 \cap \overline{L_2}$.

- For this, need an <u>alternative</u> <u>characterisation</u> of regular languages
  ↳ Also needed for representation and operations on regular languages
  ⇒ Languages are infinite sets
  ⇒ Finite representations not always easy to find (one of the goals of tcs).

1.2 Finite automata

Fix the alphabet Σ.

<u>Definition (NFA):</u>

A <u>non-deterministic</u> <u>finite automaton</u> (over Σ) is a tuple $A = (Q, q_0, \rightarrow, Q_F)$ with

- <u>states</u> Q, <u>initial state</u> q_0, <u>final states</u> Q_F, and
- <u>transition</u> <u>relation</u> $\rightarrow \subseteq Q \times \Sigma \times Q$
  (typically write $q \xrightarrow{a} q'$ instead of $(q, a, q') \in \rightarrow$)

<u>Run</u> of A is a sequence

$$q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \rightarrow \ldots q_{n-1} \xrightarrow{a_{n-1}} q_n .$$

If $w = a_0 \ldots a_{n-1}$, say this a <u>run of A on</u> w

Run is <u>accepting</u>, if $q_n \in Q_F$.

Write $q_0 \xrightarrow{w} q_n$ for the fact that there are corresponding intermediary states.

<u>Language of A</u>

$$L(A) := \{ w \in \Sigma^* \mid q_0 \xrightarrow{w} q \text{ with } q \in Q_F \}.$$

(there is an accepting run of A on w)

<u>Size of $\mathcal{A}$</u>

$$|\mathcal{A}| := |Q| + |Q_F| + |\rightarrow|$$
$$\leq |Q| + |Q| + |Q|^2|\Sigma|$$
$$\in \mathcal{O}(|Q|^2) \quad \text{for } \Sigma \text{ fixed.}$$

Number of states is important.