Alphabet and Strings

Finite automaton

A finite automaton $\mathbf{G}$ is a five tuple
$\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_a)$, where

Q : state set; a finite set of states

Σ : alphabet; a finite set of symbols

$\delta : Q \times \Sigma \to Q$: state transition function

$q_0 \in Q$: initial state

$Q_a \subseteq Q$: subset of accept states

Alphabet

Alphabet Σ is a finite set of symbols:

$$\Sigma = \{\alpha, \beta, \gamma, \sigma, ...\}$$

Strings

A string on Σ is a finite sequence of symbols from Σ:

$$s = \sigma_1 \sigma_2 \cdots \sigma_k, \quad \sigma_i \in \Sigma, \; i = 1, ..., k$$

Denote by Σ^+ the set of all strings on Σ

Strings

string without symbol

Let ϵ be the empty string ($\epsilon \notin \Sigma^+$)

union

Write $\Sigma^* = \{\epsilon\} \cup \Sigma^+$

Examples

1) $\Sigma = \{a\}$

$\Sigma^+ = ?$ $\{a, aa, aaa, \cdots\}$

$\Sigma^* = ?$ $\{\varepsilon, a, aa, aaa, \cdots\}$

$\{a\}^* = a^*$

Examples

Σ : finite set

Σ^+, Σ^* : infinite set

2) $\Sigma = \{\alpha, \beta\}$

$\Sigma^+ = ?$ $\{\alpha, \alpha\alpha, \alpha\alpha\alpha, \ldots$
$\beta, \beta\beta, \beta\beta\beta, \ldots$
$\alpha\beta, \beta\alpha, \alpha\alpha\beta, \alpha\beta\beta, \beta\alpha\alpha, \ldots\}$

$\Sigma^* = ?$

$\Sigma^+ \cup \{\varepsilon\}$

Examples

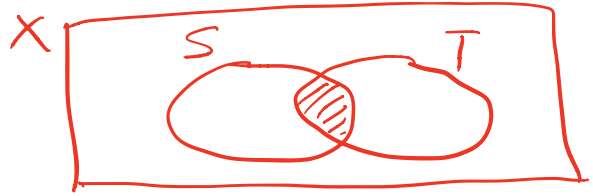
3) $\underline{\Sigma = \emptyset}$

$\Sigma^+ = ?\ \emptyset$

$\Sigma^* = ?\ \Sigma^+ \cup \{\varepsilon\} = \{\varepsilon\}$

Aside: Set operations

Let X be a set and
S, T be two subsets of X



The *union* of S and T is $S \cup T = \{x \in X \mid x \in S \text{ or } x \in T\}$

The *intersection* of S and T is $S \cap T = \{x \in X \mid x \in S \text{ and } x \in T\}$

The *complement* of S is $S^c = \{x \in X \mid x \notin S\}$

(note: $(S^c)^c = S$)

De Morgan's Law:

① $S^c \cap T^c = (S \cup T)^c$

② $S^c \cup T^c = (S \cap T)^c$

Catenation of strings

- Catenation function $cat : \Sigma^* \times \Sigma^* \to \Sigma^*$

$$cat(s_1, s_2) = s_1 s_2, \;\; s_1, s_2 \in \Sigma^*$$

s_1 s_2

$cat(s_1, s_2) = s_1 s_2$

ϵ is the unit element of catenation:

$$cat(s, \epsilon) = s = cat(\epsilon, s), \;\; s \in \Sigma^*$$

$s_1 = \alpha\beta$ $s_2 = \beta$

$s_1 s_2 = \alpha\beta\beta$

$s_2 s_1 = \beta\alpha\beta$

cat is *not* commutative:

$$cat(s_1, s_2) \neq cat(s_2, s_1), \;\; s_1, s_2 \in \Sigma^*$$

String length

Let $s \in \Sigma^*$ be a string.

Length $|s|$:

$$\text{if } s = \sigma_1 \cdots \sigma_k \in \Sigma^+, \text{ then } |s| = k$$

$$\text{if } s = \epsilon, \text{ then } |s| = 0$$

st

$$|cat(s,t)| = |s| + |t|, \quad s, t \in \Sigma^* \text{ (why?)}$$

Finite automaton

A finite automaton $\mathbf{G}$ is a five tuple
$\mathbf{G} = (Q, \Sigma, \delta, q_0, Q_a)$, where

Q : state set; a finite set of states

Σ : alphabet; a finite set of symbols

$\delta : Q \times \Sigma \to Q$: state transition function

$q_0 \in Q$: initial state

$Q_a \subseteq Q$: subset of accept states

State transition function

$$q \xrightarrow{\sigma} q'$$

$$\delta(q, \sigma) = q'$$

It's convenient to extend $\delta : Q \times \Sigma \to Q$ to $\hat{\delta} : Q \times \Sigma^* \to Q$

e.g. $q_1 \xrightarrow{\sigma_1} q_2 \xrightarrow{\sigma_2} q_3 \xrightarrow{\sigma_3} q_4$

$$\hat{\delta}(q, \epsilon) = q, \quad q \in Q$$

$$\hat{\delta}(q_1, \epsilon) = q_1$$

$$\hat{\delta}(q, \sigma) = \delta(q, \sigma), \quad q \in Q, \sigma \in \Sigma$$

$$\hat{\delta}(q_1, \sigma_1) = \delta(q_1, \sigma_1) = q_2$$

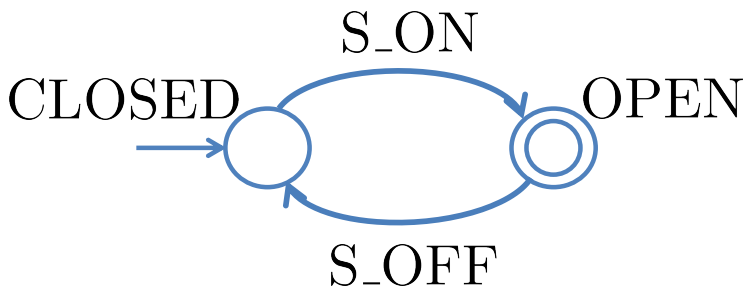
$$\hat{\delta}(q_1, \sigma_1\sigma_2) = \delta(\hat{\delta}(q_1, \sigma_1), \sigma_2) = \delta(q_2, \sigma_2) = q_3$$

$$\hat{\delta}(q, s\sigma) = \delta(\hat{\delta}(q, s), \sigma), \quad q \in Q, s \in \Sigma^*, \sigma \in \Sigma$$

$$\hat{\delta}(q_1, \sigma_1\sigma_2\sigma_3) = \delta(\hat{\delta}(q_1, \sigma_1\sigma_2), \sigma_3) = \delta(q_3, \sigma_3) = q_4$$

Henceforth drop ˆ and just write δ.

Example: automatic door



$\delta(\text{OPEN}, \varepsilon) = \text{OPEN}$

$\delta(\text{OPEN}, \text{S_OFF}.\text{S_ON}) = \text{OPEN}$

$\delta(\text{CLOSED}, \text{S_ON}.\text{S_OFF}.\text{S_ON}) = \text{OPEN}$