Languages & Automata

Overview

Main topics

Languages:	regular	context-free	[context-sensitive]	[enumerable]
			[natural languages?]	
Automata:	finite	push-down	[bounded Turing machine]	[Turing machine]

Languages

An *alphabet* Σ is a (finite) set of symbols

Examples

$\Sigma_1 = \{a\}$

$\Sigma_2 = \{0, 1\}$

$\Sigma_3 = \{A, C, G, T\}$

$\Sigma_4 = \{a, b, c, d, \ldots, x, y, z\}$

$\Sigma_5 = \{s \mid s \text{ is an ascii symbol}\}$

$\Sigma_6 =$ {あ、い、う、え、お、か、き、く、け、こ、...}

Japanese alphabet: 2×52 signs

$\Sigma_7 =$ {山 mountain, 川 river, 日 sun, 雨 rain, 水 water, 火 fire, 田 rice field, ...}

Chinese alphabet: ± 40.000 signs

$\Sigma_8 = \{0, 1, +, \times, x_0, x_1, x_2, \ldots\}$

mathematical alphabet, number of signs: countably infinite

$\Sigma_9 = \{0, 1, +, \times, x_0, x_1, x_2, \ldots\} \cup \{c_r \mid r \in \mathbb{R}\}$

mathematical alphabet, number of signs uncountably infinite

A *word* (string) over Σ is a finite list of elements from Σ

The set Σ^* consists of all words over Σ

Inductive generation of words

$$\begin{array}{rcll} & & \lambda\in\Sigma^*, & \lambda \text{ is the empty word} \\ w\in\Sigma^* & \Rightarrow & ws\in\Sigma^*, & \text{for all } s\in\Sigma \end{array}$$

Note the difference between $a\in\Sigma$ and $a\in\Sigma^*$

Think of a word as a chain of letters on a necklace:

$$\begin{array}{rcl} \lambda & = & \text{——} \\ Eva & = & \text{–}E\text{-}v\text{-}a\text{–} \end{array}$$

The difference between a and $-a-$ is clear

A *language* over Σ is a subset of Σ^*, notation $L \subseteq \Sigma^*$

Examples

$$L_1 = \{w \in \{a,b\}^* \mid abba \text{ does occur as substring of } w\}$$
$$L_2 = \{w \in \{a,b\}^* \mid abba \text{ does not occur as substring of } w\}$$

Operations on words

$$u \in \Sigma^*,\ s \in \Sigma \quad \Rightarrow \quad us \in \Sigma^* \qquad \text{part of def. of words}$$
$$u \in \Sigma^*,\ v \in \Sigma^* \quad \Rightarrow \quad u + v \in \Sigma^* \qquad \textit{concatenation}$$
$$u \in \Sigma^*,\ n \in \mathbb{N} \quad \Rightarrow \quad u^n \in \Sigma^* \qquad \text{repeating words}$$

Inductive definitions

$$\begin{array}{rcl} u + \lambda & = & u \\ u + (vs) & = & (u+v)s \end{array} \qquad \begin{array}{rcl} u^0 & = & \lambda \\ u^{k+1} & = & u^k + u \end{array}$$

usually we write concatenation $u + v$ as uv

Let $\Sigma = \{a, b, c\}$.

1. $L_1 = \{a^n \mid n \in \mathbb{N} \text{ is even}\}$
2. $L_2 = \{a^n b^n \mid n \in \mathbb{N}\}$
3. $L_3 = \{a^n b^n c^n \mid n \in \mathbb{N}\}$
4. $L_4 = \{a^n \mid n \in \mathbb{N} \text{ is prime}\}$

Given languages $L_1, L_2, L \subseteq \Sigma^*$ we can define

$$L_1 \cup L_2$$

$$L_1 L_2$$

$$L^*$$

again languages over Σ

$$\begin{array}{rcl} L_1 \cup L_2 & = & \{w \mid w{\in}L_1 \text{ or } w{\in}L_2\} \\ L_1 L_2 & = & \{w_1 w_2 \mid w_1{\in}L_1 \ \& \ w_2{\in}L_2\} \\ L^0 & = & \{\lambda\} \\ L^{n+1} & = & L^n L \\ L^* & = & \bigcup_{n\in\mathbb{N}} L^n = L^0 \cup L^1 \cup L^2 \cup \ldots \\ & \neq & \{w^n \mid w{\in}L,\ n{\in}\mathbb{N}\} \end{array}$$

Let $\Sigma = \{a, b\}$. Then $a(ba)^*bb$ is a *regular expression* denoting

$$\begin{aligned} L &= \{a(ba)^n bb \mid n \in \mathbb{N}\} \\ &= \{abb, ababb, abababb, ababababb, \ldots, a(ba)^n bb, \ldots\} \end{aligned}$$

For general Σ the regular expressions over Σ are generated by

$$\boxed{\mathtt{rexp}_\Sigma ::= \emptyset \mid \lambda \mid \mathtt{s} \mid \mathtt{rexp}_\Sigma\, \mathtt{rexp}_\Sigma \mid \mathtt{rexp}_\Sigma \cup \mathtt{rexp}_\Sigma \mid \mathtt{rexp}_\Sigma^*}$$

with $\mathtt{s} \in \Sigma$

This means $\emptyset \in \mathtt{rexp}_\Sigma$, $\lambda \in \mathtt{rexp}_\Sigma$, and $s \in \mathtt{rexp}_\Sigma$ for $s \in \Sigma$
and

$$\begin{aligned} e_1, e_2 \in \mathtt{rexp}_\Sigma &\Rightarrow (e_1 \cup e_2) \in \mathtt{rexp}_\Sigma \\ e_1, e_2 \in \mathtt{rexp}_\Sigma &\Rightarrow (e_1 e_2) \in \mathtt{rexp}_\Sigma \\ e \in \mathtt{rexp}_\Sigma &\Rightarrow (e^*) \in \mathtt{rexp}_\Sigma \end{aligned}$$

For example $(abb)^*(a \cup \lambda)$ is a regular expression

For a regular expression e over Σ we define the language $L(e) \subseteq \Sigma^*$:

$$\begin{aligned} L(\emptyset) &= \emptyset \\ L(\lambda) &= \{\lambda\} \\ L(s) &= \{s\} \\ L(e_1 e_2) &= L(e_1)L(e_2) \\ L(e_1 \cup e_2) &= L(e_1) \cup L(e_2) \\ L(e^*) &= L(e)^* \end{aligned}$$

A language L is called *regular* if $L = L(e)$ for some $e \in \mathtt{rexp}$

Examples

$L = \{w \mid bb \text{ occurs in } w\}$ over $\Sigma = \{a, b\}$ is regular:

$$L = L((a \cup b)^* bb (a \cup b)^*)$$

Also $L' = \Sigma^* - L = \{w \mid bb \text{ does not occur in } w\}$:

$$L' = L(a^*(baa^*)^* \cup a^*(baa^*)^* b)$$

Finite Automata	—	Regular Languages
Push-down Automata	—	Context-free Languages
Bounded Turing Machines	—	Context-sensitive Languages
Turing Machines	—	Enumerable Languages

Topics described on the last two lines are treated in other lectures