# Fundamentele Informatica 1 (I&E)

najaar 2014

`http://www.liacs.leidenuniv.nl/~vlietrvan1/fi1ie/`

**Rudy van Vliet**
rvvliet(at)liacs(dot)nl

college 4, 5 november 2014

3.1 Regular Languages and Regular Expressions

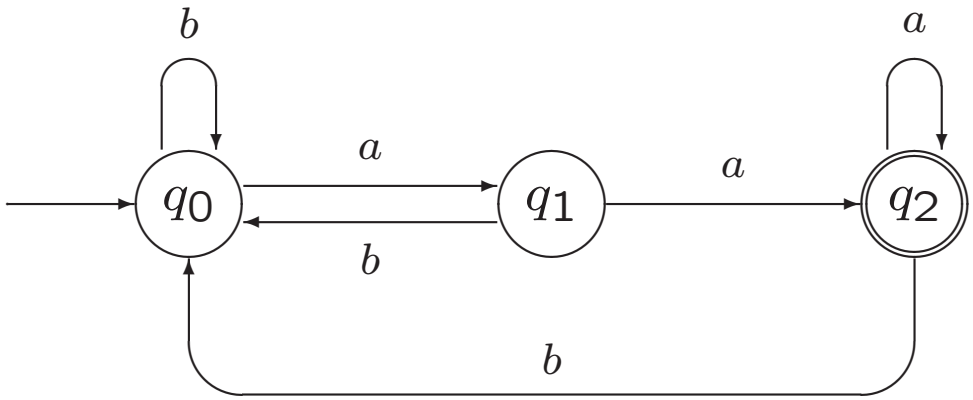
3.2 Nondeterministic Finite Automata

A slide from lecture 3:

**Example 2.1.**

A finite automaton for accepting

$L_1 = \{x \in \{a, b\}^* \mid \ x \text{ ends with } aa\}$



| | | | |
|---|---|---|---|
| reg. languages | FA | reg. grammar | reg. expression |
| cf. languages | PDA | cf. grammar | |
| re. languages | TM | unrestr. grammar | |

# 3.1. Regular Languages and Regular Expressions

(Part of) two slides from lecture 1:

Combination of union, concatenation, Kleene star:
$L_1 \cup L_2 L_3^* = \ldots$
$(L_1 \cup L_2) L_3^*$
$L_1 \cup (L_2 L_3)^*$
$(L_1 \cup L_2 L_3)^*$

Description of languages:
by formula: $L_1 = \{ab, bab\}^* \cup \{b\}\{ba\}^*\{ab\}^*$

From exercise class 2:

**Exercise 2.12.** For each of the following languages, draw an FA accepting it.

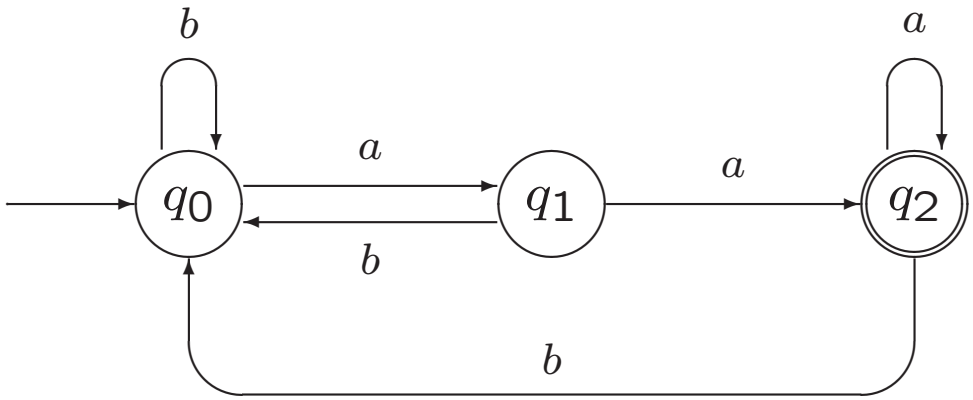
**a.** $\{a, b\}^*\{a\}$
**b.** $\{bb, ba\}^*$
**c.** $\{a, b\}^*\{b, aa\}\{a, b\}^*$
**d.** $\{bbb, baa\}^*\{a\}$
**e.** $\{a\} \cup \{b\}\{a\}^* \cup \{a\}\{b\}^*\{a\}$

A slide from lecture 3:

**Example 2.1.**

A finite automaton for accepting

$L_1 = \{x \in \{a, b\}^* \mid \ x \text{ ends with } aa\}$



**Definition 3.1.** Regular Languages over an Alphabet $\Sigma$.

If $\Sigma$ is an alphabet,
the set $\mathcal{R}$ of regular languages over $\Sigma$ is defined as follows.

1. The language $\emptyset$ is an element of $\mathcal{R}$,
   and for every $\sigma \in \Sigma$, the language $\{\sigma\}$ is in $\mathcal{R}$.

2. For any two languages $L_1$ and $L_2$ in $\mathcal{R}$,
   the three languages
   $$L_1 \cup L_2, \qquad L_1 L_2, \qquad \text{and } L_1^*$$
   are elements of $\mathcal{R}$.

(and nothing more)

$\{a, b\}^*\{aa\} \in \mathcal{R}$,
because $\{a, b\}^*\{aa\} = (\{a\} \cup \{b\})^*(\{a\}\{a\})$.

$\{\Lambda\} \in \mathcal{R}$, because ...

| **Regular Language** | **Regular Expression** |
|---|---|
| $\emptyset$ | $\emptyset$ |
| $\{\wedge\}$ | $\wedge$ |
| $\{aab\}$ | $aab$ |
| $\{a,b\}^*$ | $(a+b)^*$ |
| $\{aab\}^*\{a,ab\}$ | $(aab)^*(a+ab)$ |
| $(\{aa,bb\} \cup \{ab,ba\}\{aa,bb\}^*\{ab,ba\})^*$ | $\ldots$ |

$$(a^*b^*)^* = (a+b)^*$$

$$(a+b)^*ab(a+b)^* + b^*a^* = (a+b)^*$$

**Example 3.2.** The Language of Strings in $\{a, b\}^*$ with an Odd Number of $a$'s

**Example 3.3.** The Language of Strings in $\{a, b\}^*$ Ending with $b$ and Not Containing $aa$

**Exercise.**

Find a regular expression corresponding to the language of all strings over $\{a, b\}$ of even length.

**Example 3.4.** Strings in $\{a, b\}^*$ in Which Both the Number of $a$'s and the Number of $b$'s are Even

**Example 3.5.** Regular Expressions and Programming Languages

identifiers
numeric 'literals' (constants)

## Regular Expressions in Unix

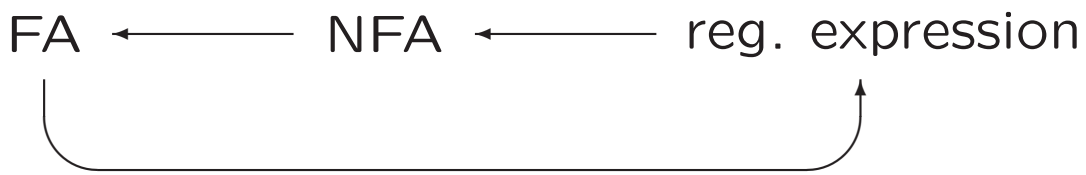
Lex

Vi

grep

# 3.2 Nondeterministic Finite Automata

| reg. languages | FA | reg. grammar | reg. expression |
|---|---|---|---|
| cf. languages | PDA | cf. grammar | |
| re. languages | TM | unrestr. grammar | |



**Example 3.6.** Accepting the Language $\{aa, aab\}^*\{b\}$

Computation tree. . .

**Example 3.9.** Accepting the Language $\{aab\}^*\{a, aba\}^*$

Computation tree. . .