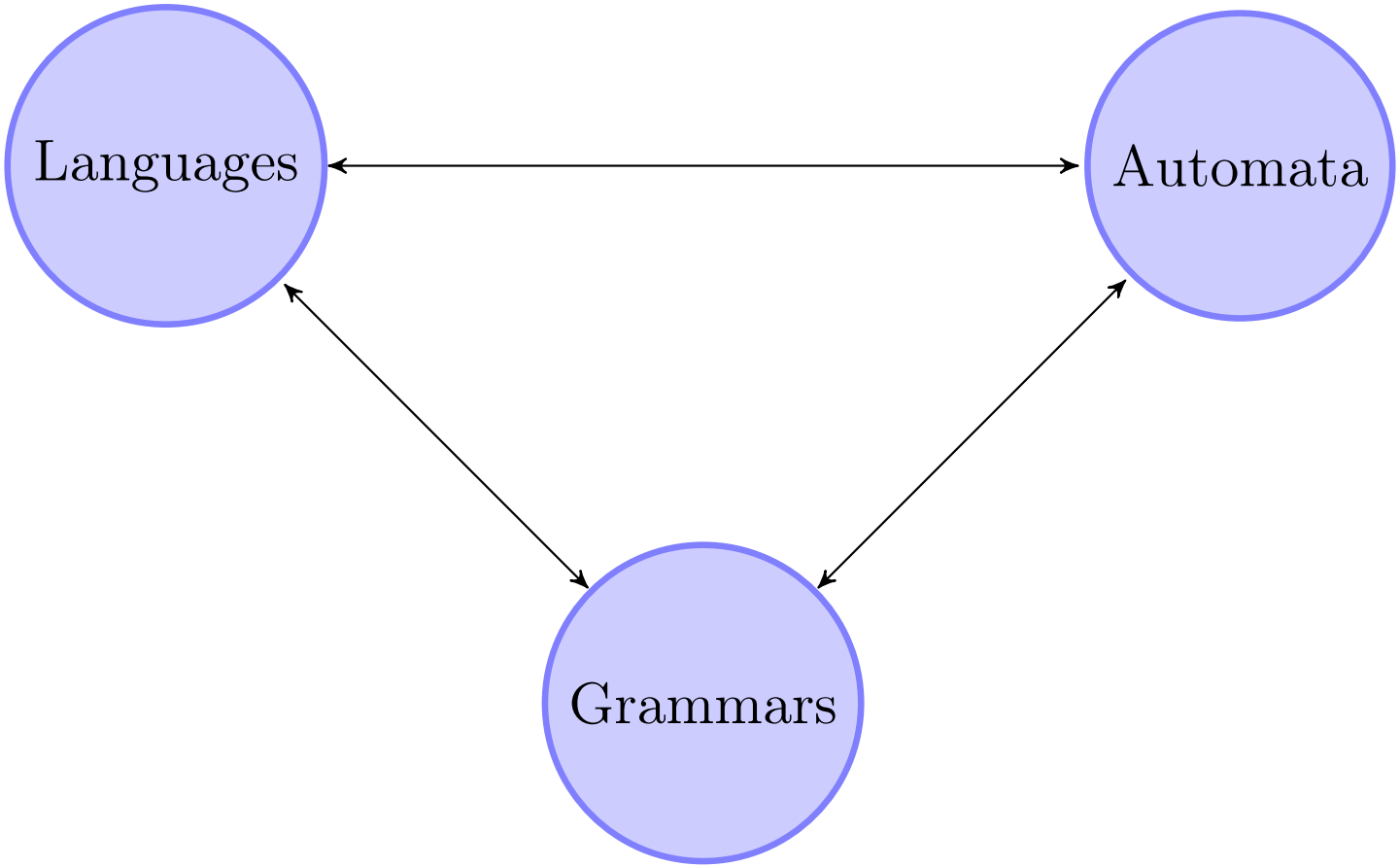
AUTOMATA, LANGUAGES AND MONOIDS I

Jiří Sýkora

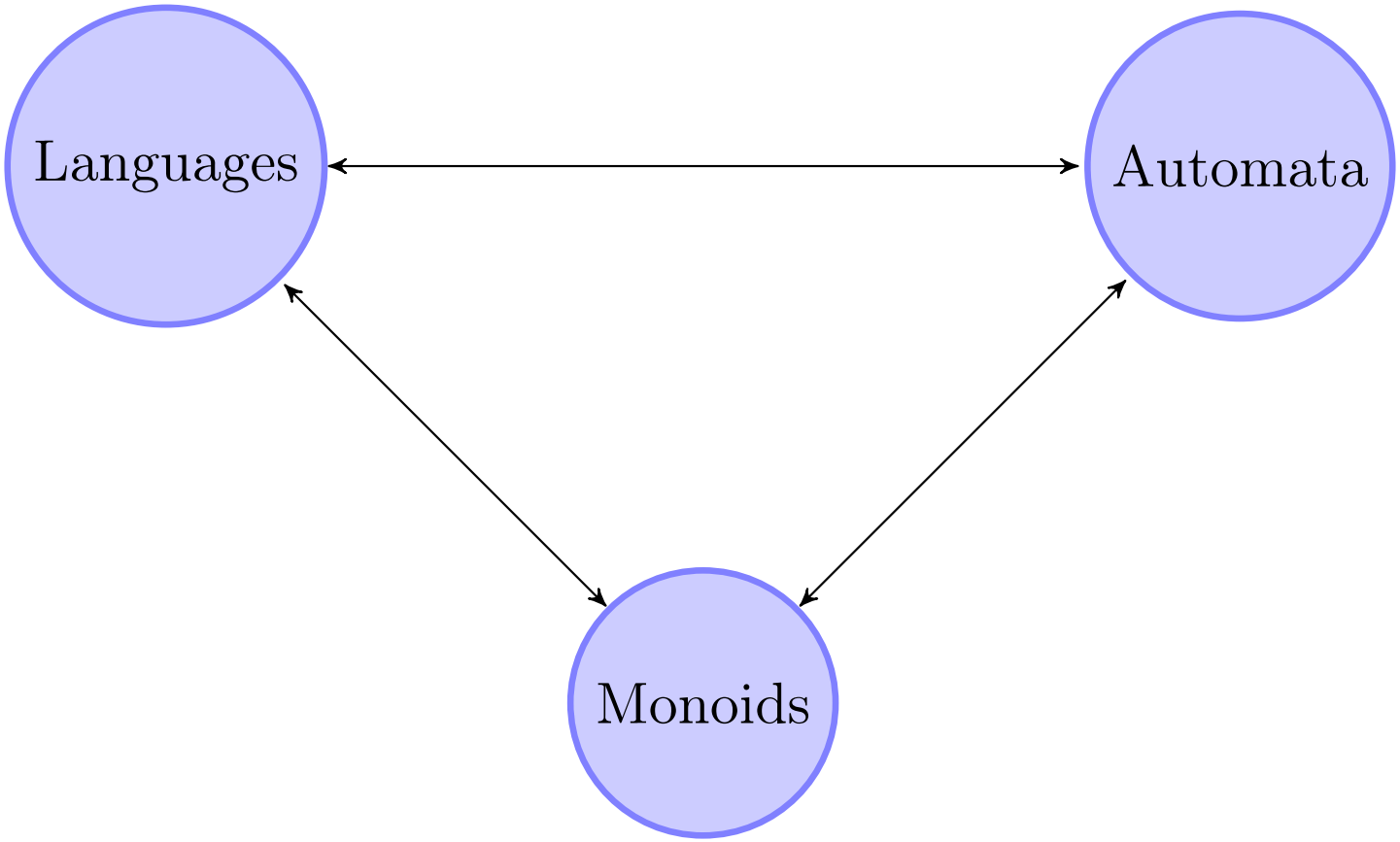
Department of Algebra
Faculty of Mathematics and Physics
Charles University in Prague

24th November 2014

Motivation



Motivation



Outline

- First part: introduction, basic definitions, properties etc.

Outline

- First part: introduction, basic definitions, properties etc.
- Second part (Adéla): piecewise testable languages

Outline

- First part: introduction, basic definitions, properties etc.
- Second part (Adéla): piecewise testable languages
- Third part (Miloš): Simon's theorem and its consequences

Outline

- First part: Useless
- Second part (Adéla): Useless
- Third part (Miloš): Cancelled

Outline

- First part: Not completely useless
- Second part (Adéla): Not completely useless
- Third part: Short summary

Semigroups and monoids

Definition

A *semigroup* is a set with an associative binary operation.

Semigroups and monoids

Definition

A *semigroup* is a set with an associative binary operation.

Definition

A *monoid* is a semigroup with a neutral element.

Semigroups and monoids

Definition

A *semigroup* is a set with an associative binary operation.

Definition

A *monoid* is a semigroup with a neutral element.

Definition

Let S and T be semigroups. We say that T is a *quotient* of S if there exists a surjective morphism $\varphi : S \to T$. The semigroup T *divides* the semigroup S if T is a quotient of a subsemigroup of S.

Semigroups and monoids

Definition

A *semigroup* is a set with an associative binary operation.

Definition

A *monoid* is a semigroup with a neutral element.

Definition

Let S and T be semigroups. We say that T is a *quotient* of S if there exists a surjective morphism $\varphi : S \to T$. The semigroup T *divides* the semigroup S if T is a quotient of a subsemigroup of S.

Definition

A *congruence* on a semigroup S is an equivalence relation $\sim$ on S satisfying for each $a, b, c \in S$: $a \sim b \Rightarrow (a \cdot c \sim b \cdot c \wedge c \cdot a \sim c \cdot b)$.

Words

Let A be (finite) a set. We will call it an *alphabet*. Elements of this set are called *letters* or *symbols*.

Words

Let A be (finite) a set. We will call it an *alphabet*. Elements of this set are called *letters* or *symbols*.

Definition

A *word* over the alphabet A is a finite sequence $a_1 a_2 \cdots a_n$ of letters from A.

Words

Let A be (finite) a set. We will call it an *alphabet*. Elements of this set are called *letters* or *symbols*.

Definition

A *word* over the alphabet A is a finite sequence $a_1 a_2 \cdots a_n$ of letters from A.

Definition

Denote by A^* the set of all words over A (the empty sequence is also a word). Then A^* with the associative operation of concatenation forms a *free monoid* on the set A. The neutral element is the empty word λ.

Words

Let A be (finite) a set. We will call it an *alphabet*. Elements of this set are called *letters* or *symbols*.

Definition

A *word* over the alphabet A is a finite sequence $a_1 a_2 \cdots a_n$ of letters from A.

Definition

Denote by A^* the set of all words over A (the empty sequence is also a word). Then A^* with the associative operation of concatenation forms a *free monoid* on the set A. The neutral element is the empty word λ.

Definition

The set $A^+ = A^* \setminus \{\lambda\}$ with the same operation of concatenation is a *free semigroup* on the set A.

Subwords, factors, etc.

Definition

Let $w \in A^*$ be a word and let $a \in A$ be a letter. The number of occurences of a in w is denoted $|w|_a$.

Subwords, factors, etc.

Definition

Let $w \in A^*$ be a word and let $a \in A$ be a letter. The number of occurences of a in w is denoted $|w|_a$.

Definition

A word u is a *prefix* (or *left factor*) of a word w if there exists a word v such that $w = uv$. We define a *suffix* of w in a similar way.

Subwords, factors, etc.

Definition

Let $w \in A^*$ be a word and let $a \in A$ be a letter. The number of occurences of a in w is denoted $|w|_a$.

Definition

A word u is a *prefix* (or *left factor*) of a word w if there exists a word v such that $w = uv$. We define a *suffix* of w in a similar way.

Definition

A word u is a *factor* of a word w if there exist words v_1, v_2 such that $w = v_1 u v_2$.

Subwords, factors, etc.

Definition

Let $w \in A^*$ be a word and let $a \in A$ be a letter. The number of occurences of a in w is denoted $|w|_a$.

Definition

A word u is a *prefix* (or *left factor*) of a word w if there exists a word v such that $w = uv$. We define a *suffix* of w in a similar way.

Definition

A word u is a *factor* of a word w if there exist words v_1, v_2 such that $w = v_1 u v_2$.

Definition

A word $u = a_1 \dots a_n$, $a_i \in A$ is a *subword* of a word w if there exist words $v_0, v_1, \dots, v_n$ such that $w = v_0 a_1 v_1 \dots a_n v_n$.

Examples

Example

Take a word $u = abacbacb$.

$$abacbacb$$

Examples

Example

Take a word $u = abacbacb$. Then aba is a prefix of u,

$$\boldsymbol{aba}cbacb$$

Examples

Example

Take a word $u = abacbacb$. Then aba is a prefix of u, acb is a suffix of u,

$$abacb\boldsymbol{acb}$$

Examples

Example

Take a word $u = abacbacb$. Then aba is a prefix of u, acb is a suffix of u, $bacb$ is a factor of u

$$a\boldsymbol{bacb}acb$$

Examples

Example

Take a word $u = abacbacb$. Then aba is a prefix of u, acb is a suffix of u, $bacb$ is a factor of u and $bcbb$ is a subword of u.

$$a\boldsymbol{b}a\boldsymbol{cb}ac\boldsymbol{b}$$

Languages

Definition

Let A be a finite alphabet. A *language* over the alphabet A is a subset of A^*.

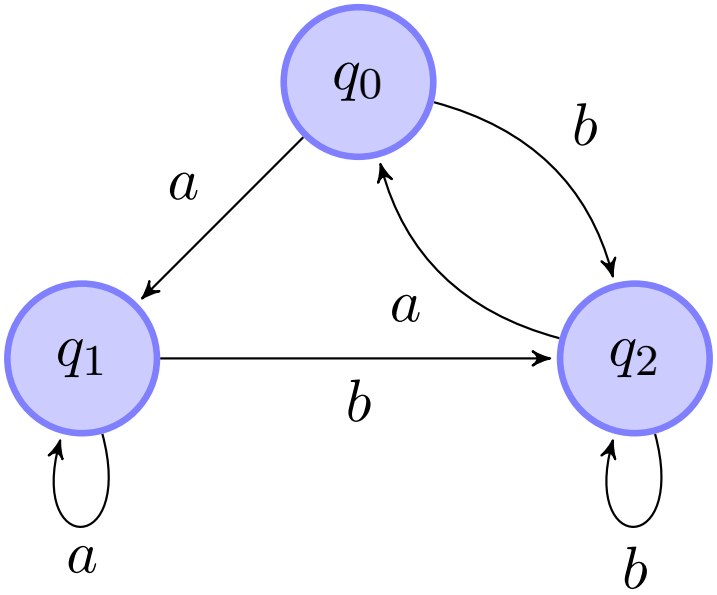
Languages

Definition

Let A be a finite alphabet. A *language* over the alphabet A is a subset of A^*.

Example

$L = \{a^k \mid k \geq 1\} \subset \{a, b\}^*$ is a language.



Definition

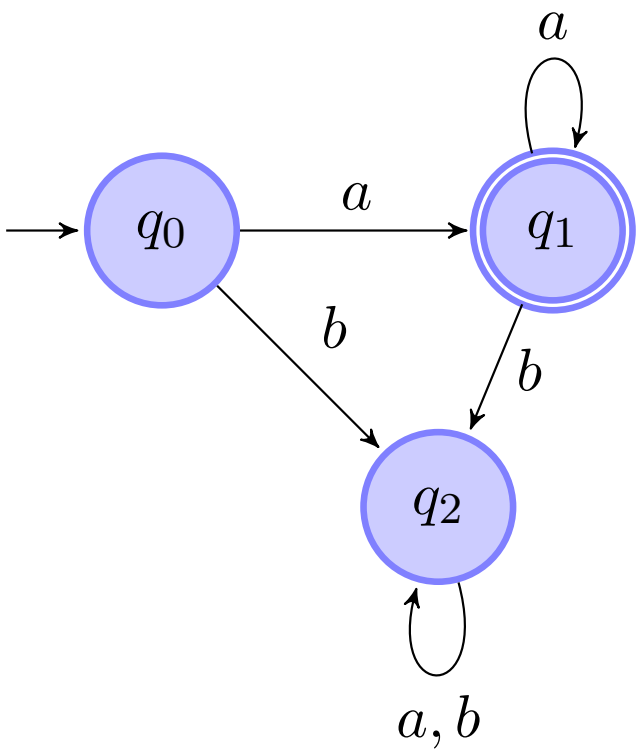
An *automaton* is a triplet $\mathscr{A} = (Q, A, \cdot)$, where Q is a (finite) set of states, A is a finite alphabet and $\cdot$ is a function from $Q \times A$ to Q.

Definition

We say that a language L is *recognized* by an automaton $\mathscr{A} = (Q, A, \cdot)$ if there exists a state $q_0 \in Q$ and a set of states F such that $u \in L$ iff $q_0 \cdot u \in F$.

Definition

We say that a language L is *recognized* by an automaton $\mathscr{A} = (Q, A, \cdot)$ if there exists a state $q_0 \in Q$ and a set of states F such that $u \in L$ iff $q_0 \cdot u \in F$.



Regular languages

Definition

Let A be a finite alphabet. The set of *regular languages* (also *rational languages*) over A is the smallest set of languages of A^* such that

1. the empty language $\varnothing$ is regular,
2. for every word $u \in A^*$, the language $\{u\}$ is regular,
3. if L_1 and L_2 are regular languages, then $L_1 \cup L_2$, $L_1 \cdot L_2$ and L_1^* are also regular.

Regular languages

Definition

Let A be a finite alphabet. The set of *regular languages* (also *rational languages*) over A is the smallest set of languages of A^* such that

1. the empty language $\varnothing$ is regular,
2. for every word $u \in A^*$, the language $\{u\}$ is regular,
3. if L_1 and L_2 are regular languages, then $L_1 \cup L_2$, $L_1 \cdot L_2$ and L_1^* are also regular.

Example

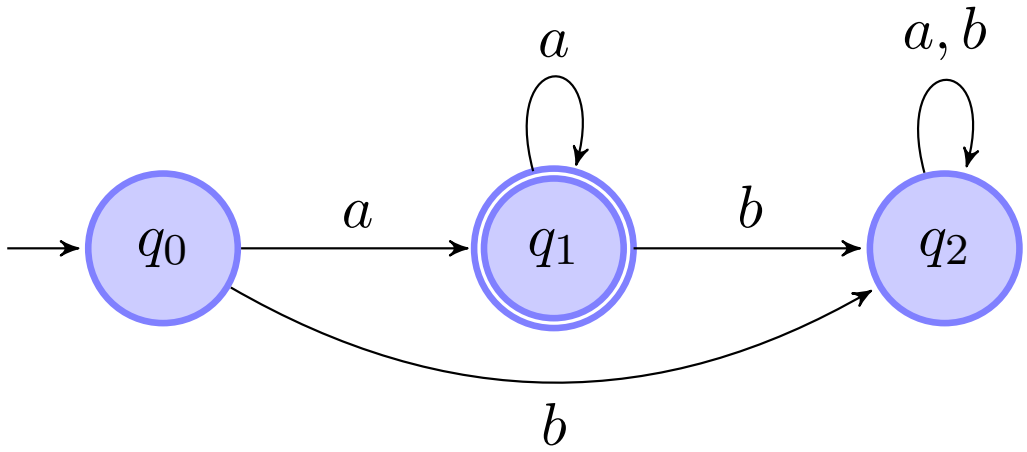
$L = \{a^k \mid k \geq 1\} = \{a\} \cdot \{a\}^*$ is a regular language.

Transition monoid

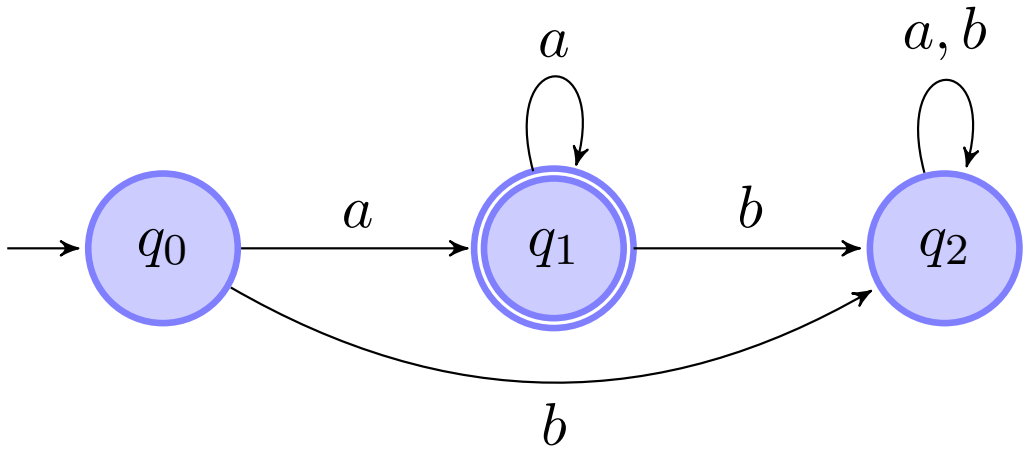
Definition

Let $\mathscr{A} = (Q, A, \cdot)$ be an automaton. We can extend the operation $\cdot$ to a function from $Q \times A^*$ to Q by the following rules: $q \cdot \lambda = q$ and $q \cdot (wa) = (q \cdot w) \cdot a$ where $q \in Q, w \in A^*$ and $a \in A$. Each word from A^* thus defines a function from Q to Q. The monoid generated by all these functions (w varying over A^*) is called the *transition monoid* of the automaton $\mathscr{A}$. It is denoted $M(\mathscr{A})$.

Examples



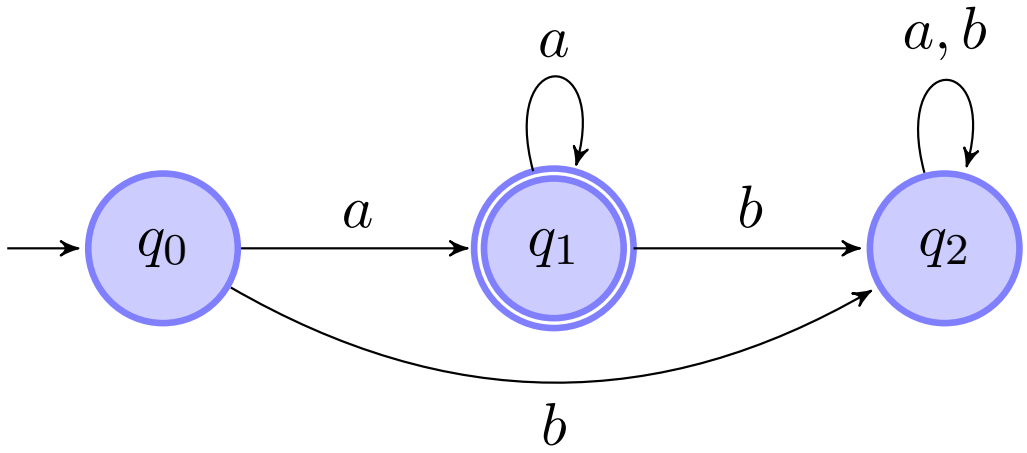
Examples



Example

$$q_0 \cdot a = q_1 \quad q_1 \cdot a = q_1 \quad q_2 \cdot a = q_2$$
$$q_0 \cdot b = q_2 \quad q_1 \cdot b = q_2 \quad q_2 \cdot b = q_2$$

Examples

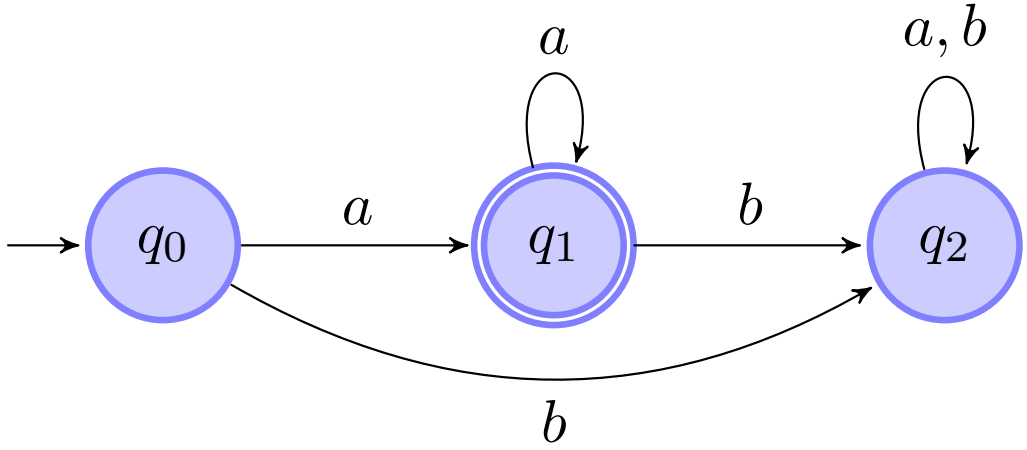


Example

$$q_0 \cdot a = q_1 \quad q_1 \cdot a = q_1 \quad q_2 \cdot a = q_2$$
$$q_0 \cdot b = q_2 \quad q_1 \cdot b = q_2 \quad q_2 \cdot b = q_2$$

So we have: $f_a(q_0) = q_1$, $f_a(q_1) = q_1$, $f_a(q_2) = q_2$ and $f_b(q_0) = q_2$, $f_b(q_1) = q_2$, $f_b(q_2) = q_2$.

Examples

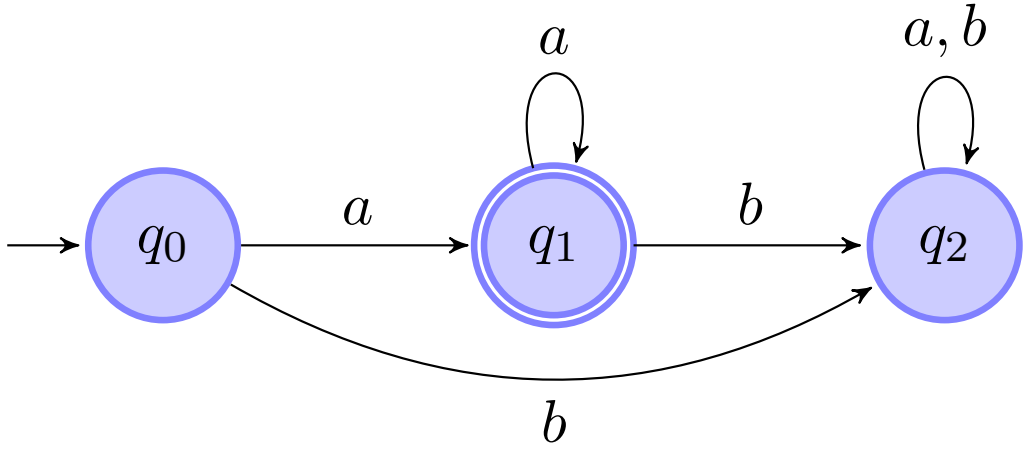


Example

$$q_0 \cdot a = q_1 \quad q_1 \cdot a = q_1 \quad q_2 \cdot a = q_2$$
$$q_0 \cdot b = q_2 \quad q_1 \cdot b = q_2 \quad q_2 \cdot b = q_2$$

So we have: $f_a(q_0) = q_1$, $f_a(q_1) = q_1$, $f_a(q_2) = q_2$ and $f_b(q_0) = q_2$, $f_b(q_1) = q_2$, $f_b(q_2) = q_2$. Moreover $f_a \circ f_a = f_a$, $f_a \circ f_b = f_b \circ f_a = f_b \circ f_b = f_b$.

Examples



Example

$$q_0 \cdot a = q_1 \quad q_1 \cdot a = q_1 \quad q_2 \cdot a = q_2$$
$$q_0 \cdot b = q_2 \quad q_1 \cdot b = q_2 \quad q_2 \cdot b = q_2$$

So we have: $f_a(q_0) = q_1$, $f_a(q_1) = q_1$, $f_a(q_2) = q_2$ and $f_b(q_0) = q_2$, $f_b(q_1) = q_2$, $f_b(q_2) = q_2$. Moreover $f_a \circ f_a = f_a$, $f_a \circ f_b = f_b \circ f_a = f_b \circ f_b = f_b$. Therefore $M(\mathscr{A}) = \{\mathrm{id}, f_a, f_b\}$.

Recognizable languages

Definition

A language L is called *recognizable* if it is recognized by a finite monoid, i.e. there exists a finite monoid M and a morphism $\alpha : A^* \to M$ such that $L = \alpha^{-1}(P)$ for some $P \subseteq M$.

Recognizable languages

Definition

A language L is called *recognizable* if it is recognized by a finite monoid, i.e. there exists a finite monoid M and a morphism $\alpha : A^* \to M$ such that $L = \alpha^{-1}(P)$ for some $P \subseteq M$.

Proposition

If $L \subseteq A^*$ is recognized by an automaton, it is recognized by the transition monoid of this automaton. Moreover, L is recognized by a finite automaton if and only if L is recognizable.

Recognizable languages

Definition

A language L is called *recognizable* if it is recognized by a finite monoid, i.e. there exists a finite monoid M and a morphism $\alpha : A^* \to M$ such that $L = \alpha^{-1}(P)$ for some $P \subseteq M$.

Proposition

If $L \subseteq A^*$ is recognized by an automaton, it is recognized by the transition monoid of this automaton. Moreover, L is recognized by a finite automaton if and only if L is recognizable.

Theorem

A language $L \subseteq A^*$ is regular iff it is recognizable.

Examples

Remark

Regular $\Leftrightarrow$ *recognizable* $\Leftrightarrow$ *recognized by a finite automaton.*

Examples

Remark

Regular $\Leftrightarrow$ *recognizable* $\Leftrightarrow$ *recognized by a finite automaton.*

Example

The language $\{a^i b^i \mid i \geq 0\}$ is not regular.

Examples

Remark

Regular $\Leftrightarrow$ *recognizable* $\Leftrightarrow$ *recognized by a finite automaton.*

Example

The language $\{a^i b^i \mid i \geq 0\}$ is not regular.

Proof.

Suppose there exists a finite automaton which recognizes the language. This automaton has n states. Take a word $a^n b^n$. Then, when reading a's, there must be some loop. We can repeat this path once again, i.e. the automaton would accept the word $a^{n+\ell} b^n$ for some $\ell \geq 1$. However, this word does not belong to the language. $\square$

Properties of regular languages

Proposition

Let L_1 and L_2 be regular languages. Then also

1. $L_1 \cup L_2$,

are regular languages.

Properties of regular languages

Proposition

Let L_1 and L_2 be regular languages. Then also

1. $L_1 \cup L_2$,
2. $L_1 \cap L_2$,

are regular languages.

Properties of regular languages

Proposition

Let L_1 and L_2 be regular languages. Then also

1. $L_1 \cup L_2$,
2. $L_1 \cap L_2$,
3. $L_1 \setminus L_2$,

are regular languages.

Properties of regular languages

Proposition

Let L_1 and L_2 be regular languages. Then also

1. $L_1 \cup L_2$,
2. $L_1 \cap L_2$,
3. $L_1 \setminus L_2$,
4. $\overline{L_1}$

are regular languages.

Syntactic monoid

Definition

Let $L \subseteq A^*$ be a language. We define the *syntactic* congruence of L (denoted $\sim_L$) on A^* by $u \sim_L v$ iff $xuy \in L \Leftrightarrow xvy \in L$ for every $x, y \in A^*$. The *syntactic monoid* of L is then defined as $M(L) = A^*/\sim_L$.

Syntactic monoid

Definition

Let $L \subseteq A^*$ be a language. We define the *syntactic* congruence of L (denoted $\sim_L$) on A^* by $u \sim_L v$ iff $xuy \in L \Leftrightarrow xvy \in L$ for every $x, y \in A^*$. The *syntactic monoid* of L is then defined as $M(L) = A^*/\sim_L$.

Example

Let $L = \{a^k \mid k \geq 1\}$. Then there are three equivalence classes of $\sim_L$, namely $[a]_{\sim_L} = \{u \in A^* \mid u \neq \lambda, |u|_b = 0\}$, $[b]_{\sim_L} = \{u \in A^* \mid |u|_b \geq 1\}$ and $[\lambda]_{\sim_L} = \{\lambda\}$.

Syntactic monoid

Definition

Let $L \subseteq A^*$ be a language. We define the *syntactic* congruence of L (denoted $\sim_L$) on A^* by $u \sim_L v$ iff $xuy \in L \Leftrightarrow xvy \in L$ for every $x, y \in A^*$. The *syntactic monoid* of L is then defined as $M(L) = A^*/\sim_L$.

Example

Let $L = \{a^k \mid k \geq 1\}$. Then there are three equivalence classes of $\sim_L$, namely $[a]_{\sim_L} = \{u \in A^* \mid u \neq \lambda, |u|_b = 0\}$, $[b]_{\sim_L} = \{u \in A^* \mid |u|_b \geq 1\}$ and $[\lambda]_{\sim_L} = \{\lambda\}$. The operation is described by the following table:

$\cdot$	$[\lambda]_{\sim_L}$	$[a]_{\sim_L}$	$[b]_{\sim_L}$
$[\lambda]_{\sim_L}$	$[\lambda]_{\sim_L}$	$[a]_{\sim_L}$	$[b]_{\sim_L}$
$[a]_{\sim_L}$	$[a]_{\sim_L}$	$[b]_{\sim_L}$	$[b]_{\sim_L}$
$[b]_{\sim_L}$	$[b]_{\sim_L}$	$[b]_{\sim_L}$	$[b]_{\sim_L}$

Syntactic monoid II

Proposition

A monoid M recognizes a language L iff $M(L)$ divides M.

Syntactic monoid II

Proposition

A monoid M recognizes a language L iff $M(L)$ divides M.

Proposition

Let L be a regular language. Then there exists a uniquely determined (up to renaming of states) finite automaton recognizing L that has a minimum number of states among the automata recognizing L. It is called the minimal automaton of L.

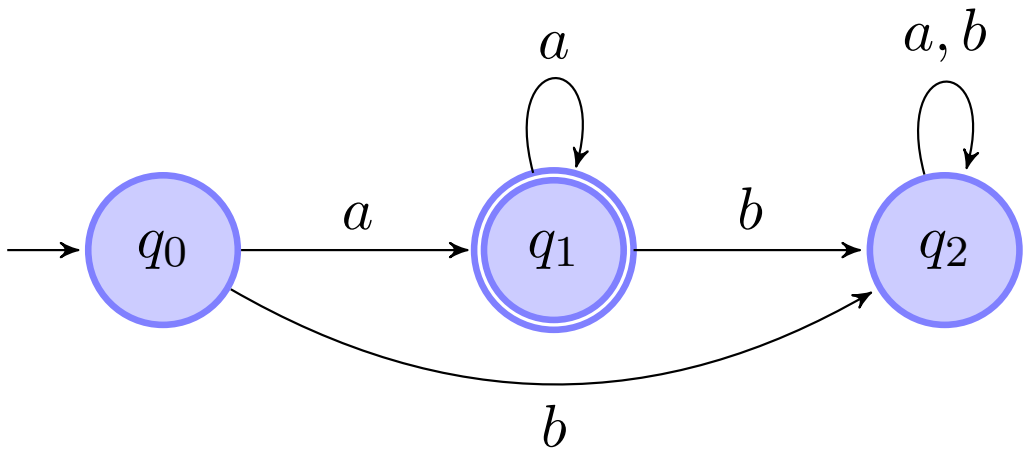
Syntactic monoid II

Proposition

A monoid M recognizes a language L iff $M(L)$ divides M.

Proposition

Let L be a regular language. Then there exists a uniquely determined (up to renaming of states) finite automaton recognizing L that has a minimum number of states among the automata recognizing L. It is called the minimal automaton of L.



Syntactic monoid III

Proposition

Let L be a regular language. The transition monoid of the minimal automaton of L is equal (isomorphic) to the syntactic monoid of L.

Syntactic monoid III

Proposition

Let L be a regular language. The transition monoid of the minimal automaton of L is equal (isomorphic) to the syntactic monoid of L.

Example

Take $L = \{a^k \mid k \geq 1\}$ and its minimal automaton $\mathscr{A}$. We know that $M(\mathscr{A}) = \{\mathrm{id}, f_a, f_b\}$ and $M(L) = \{[\lambda]_{\sim_L}, [a]_{\sim_L}, [b]_{\sim_L}\}$. It is obvious that φ defined by
$\varphi(\mathrm{id}) = [\lambda]_{\sim_L}$, $\varphi(f_a) = [a]_{\sim_L}$, $\varphi(f_b) = [b]_{\sim_L}$ is a monoid isomorphism.

Varieties

Definition

A *variety* of finite semigroups (or monoids) is a class of finite semigroups (or monoids) closed under division and finite products.

Varieties

Definition

A *variety* of finite semigroups (or monoids) is a class of finite semigroups (or monoids) closed under division and finite products.

Definition

We say that a semigroup S satisfies the equation $u = v$, $u, v \in A^+$ if $\varphi(u) = \varphi(v)$ for every morphism $\varphi : A^+ \to S$.

Definition

We say that a variety $\mathbf{V}$ is defined (ultimately defined) by equations $u_n = v_n$, $n > 0$ if S lies in $\mathbf{V}$ iff S satisfies the equations $u_n = v_n$ for every $n > 0$ (for every n large enough).

Varieties II

Definition

We say that a variety $\mathbf{V}$ is defined (ultimately defined) by equations $u_n = v_n$, $n > 0$ if S lies in $\mathbf{V}$ iff S satisfies the equations $u_n = v_n$ for every $n > 0$ (for every n large enough).

Example

The variety of finite commutative semigroups is defined by the equation $xy = yx$.

Definition

Let M be a monoid. We define on M an equivalence relation $\mathscr{J}$ in the following way: $a \mathscr{J} b \Leftrightarrow MaM = MbM$.

Definition

Let M be a monoid. We define on M an equivalence relation $\mathscr{J}$ in the following way: $a \mathscr{J} b \Leftrightarrow MaM = MbM$.

Remark

$a \mathscr{J} b$ iff there exist $u, v, x, y \in M$ such that $uav = b$ and $xby = a$.

Definition

Let M be a monoid. We define on M an equivalence relation $\mathscr{J}$ in the following way: $a \mathscr{J} b \Leftrightarrow MaM = MbM$.

Remark

$a \mathscr{J} b$ iff there exist $u, v, x, y \in M$ such that $uav = b$ and $xby = a$.

Definition

We say that M is $\mathscr{J}$-trivial if $a \mathscr{J} b \Rightarrow a = b$.

Definition

Let M be a monoid. We define on M an equivalence relation $\mathscr{J}$ in the following way: $a \mathscr{J} b \Leftrightarrow MaM = MbM$.

Remark

$a \mathscr{J} b$ iff there exist $u, v, x, y \in M$ such that $uav = b$ and $xby = a$.

Definition

We say that M is $\mathscr{J}$-trivial if $a \mathscr{J} b \Rightarrow a = b$.

Theorem

The variety $\boldsymbol{J}$ of finite $\mathscr{J}$-trivial monoids is ultimately defined by the equations $(xy)^n x = (xy)^n = y(xy)^n$.

Thank you for your attention!