Automata and Formal Languages

Winter 2018/19

Syllabus

Course schedule

Lectures

Javier Esparza (esparza@in.tum.de)

Monday:	10:00 – 11:30	Room:	IMETUM, E.126
Thursday:	14:00 – 15:30	Room:	02.13.010

Exercises

Salomon Sickert (sickert@in.tum.de)

Tuesday:	12:00 – 13:30	Room:	IMETUM, E.126

Course schedule

Lectures

Javier Esparza (esparza@in.tum.de)

Monday:	10:15 – 11:45?	Room:	IMETUM, E.126
Thursday:	14:15 – 15:45?	Room:	02.13.010

Exercises

Salomon Sickert (sickert@in.tum.de)

Tuesday:	12:15 – 13:45?	Room:	IMETUM, E.126

Course content

Automata on finite words

1. Automata classes and conversions
2. Minimization and reduction
3. Boolean operations and tests
4. Operations on relations
5. Operations on finite universes: decision diagrams
6. Automata and logic
7. Pattern-matching, verification, Presburger arithmetic

Course content

Automata on infinite words

8. Automata classes and conversions
9. Boolean operations
10. Emptiness check
11. Verification using temporal logic

Grading

Written exam

- end of the term
- /40 points

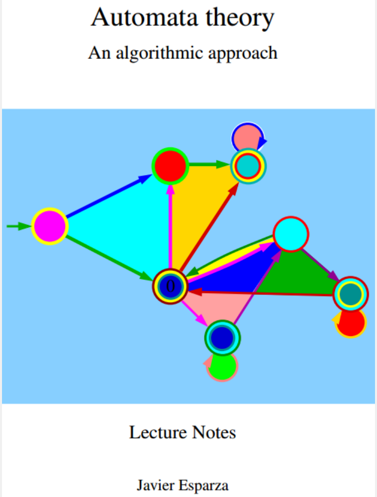
Exercises not graded!

Points	Grade
[36, 40]	1,0
[34, 36)	1,3
[32, 34)	1,7
[30, 32)	2,0
[28, 30)	2,3
[26, 28)	2,7
[24, 26)	3,0
[22, 24)	3,3
[19, 22)	3,7
[17, 19)	4,0
[11, 17)	4,3
[5, 11)	4,7
[0, 5)	5,0

Material

- Lecture notes available online
- Slides available online
- No book to buy

```
www7.in.tum.de > Teaching
               > Automata
               > more info
```



Automata theory: brief recap

Formal languages

An *alphabet* is a nonempty finite set of *letters*

e.g. $\{\mathtt{0}, \mathtt{1}\}$, $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$, $\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]\}$, $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$

A *(finite) word* is a finite sequence of letters

e.g. $\mathtt{1001}$, $\mathtt{hello}$, $\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]\left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$, $\clubsuit\clubsuit\diamondsuit$, ε

A *language* is a set of words

e.g. $\{\mathtt{1}, \mathtt{10}, \mathtt{100}, \mathtt{1000}, \ldots\}$, $\{\mathtt{aa}, \mathtt{aba}, \mathtt{abbba}, \ldots\}$

Formal languages

An *alphabet* is a nonempty finite set of *letters*

e.g. $\{\mathtt{0}, \mathtt{1}\}$, $\{\mathtt{a}, \mathtt{b}, \dots, \mathtt{z}\}$, $\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]\}$, $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$

A *(finite) word* is a finite sequence of letters

e.g. $\mathtt{1001}$, $\mathtt{hello}$, $\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]\left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$, $\clubsuit\clubsuit\diamondsuit$, ε

A *language* is a set of words

e.g. $\{\mathtt{1}, \mathtt{10}, \mathtt{100}, \mathtt{1000}, \dots\}$, $\{\mathtt{aa}, \mathtt{aba}, \mathtt{abbba}, \dots\}$

Formal languages

An *alphabet* is a nonempty finite set of *letters*

e.g. $\{\mathtt{0}, \mathtt{1}\}$, $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$, $\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]\}$, $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$

A *(finite) word* is a finite sequence of letters

e.g. $\mathtt{1001}$, $\mathtt{hello}$, $\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]\left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$, $\clubsuit\clubsuit\diamondsuit$, ε

A *language* is a set of words

e.g. $\{\mathtt{1}, \mathtt{10}, \mathtt{100}, \mathtt{1000}, \ldots\}$, $\{\mathtt{aa}, \mathtt{aba}, \mathtt{abbba}, \ldots\}$

Formal languages

Let $u = a_1 \cdots a_n$ and $v = b_1 \cdots b_m$ be words

Concatenation:

$$u \cdot v = uv = a_1 \cdots a_n b_1 \cdots b_m$$
$$\varepsilon \cdot u = u = u \cdot \varepsilon$$

Exponentiation: $u^0 = \varepsilon$, $u^{k+1} = u^k \cdot u$

e.g. $\mathtt{a}^0 = \varepsilon$, $\mathtt{a}^1 = \mathtt{a}$, $(\mathtt{hallo})^2 = \mathtt{hallohallo}$,
$\mathtt{1}^5 = \mathtt{11111}$, $\varepsilon^{1000} = \varepsilon$, $\mathtt{ab} \cdot \mathtt{cde} = \mathtt{abcde}$

Formal languages

Let $u = a_1 \cdots a_n$ and $v = b_1 \cdots b_m$ be words

Concatenation:

$$u \cdot v = uv = a_1 \cdots a_n b_1 \cdots b_m$$
$$\varepsilon \cdot u = u = u \cdot \varepsilon$$

Exponentiation: $u^0 = \varepsilon$, $u^{k+1} = u^k \cdot u$

e.g. $\mathtt{a}^0 = \varepsilon$, $\mathtt{a}^1 = \mathtt{a}$, $(\mathtt{hallo})^2 = \mathtt{hallohallo}$,
$\mathtt{1}^5 = \mathtt{11111}$, $\varepsilon^{1000} = \varepsilon$, $\mathtt{ab} \cdot \mathtt{cde} = \mathtt{abcde}$

Formal languages

Let $u = a_1 \cdots a_n$ and $v = b_1 \cdots b_m$ be words

$$\begin{aligned} \textit{Concatenation:} \quad u \cdot v &= uv = a_1 \cdots a_n b_1 \cdots b_m \\ \varepsilon \cdot u &= u \;\;= u \cdot \varepsilon \end{aligned}$$

Exponentiation: $u^0 = \varepsilon$, $u^{k+1} = u^k \cdot u$

e.g. $\texttt{a}^0 = \varepsilon$, $\texttt{a}^1 = \texttt{a}$, $(\texttt{hallo})^2 = \texttt{hallohallo}$,
$\texttt{1}^5 = \texttt{11111}$, $\varepsilon^{1000} = \varepsilon$, $\texttt{ab} \cdot \texttt{cde} = \texttt{abcde}$

Formal languages

Let $u = a_1 \cdots a_n$ and $v = b_1 \cdots b_m$ be words

Concatenation:
$$u \cdot v = uv = a_1 \cdots a_n b_1 \cdots b_m$$
$$\varepsilon \cdot u = u = u \cdot \varepsilon$$

Exponentiation: $u^0 = \varepsilon$, $u^{k+1} = u^k \cdot u$

e.g. $\mathtt{a}^0 = \varepsilon$, $\mathtt{a}^1 = \mathtt{a}$, $(\mathtt{hallo})^2 = \mathtt{hallohallo}$,
$\mathtt{1}^5 = \mathtt{11111}$, $\varepsilon^{1000} = \varepsilon$, $\mathtt{ab} \cdot \mathtt{cde} = \mathtt{abcde}$

Formal languages

Let L and L' be languages over alphabet Σ

Concatenation: $L \cdot L' = LL' = \{u \cdot v : u \in L, v \in L'\}$

Exponentiation: $L^0 = \{\varepsilon\}$, $L^{k+1} = L^k \cdot L$

Iteration: $L^* = L^0 \cup L^1 \cup L^2 \cup \cdots$

Complement: $\overline{L} = \Sigma^* \setminus L$

e.g.

$$\begin{aligned} \{\mathbf{aa}, \mathbf{bb}\} \cdot \{\mathbf{c}, \mathbf{d}\} &= \{\mathbf{aac}, \mathbf{aad}, \mathbf{bbc}, \mathbf{bbd}\}, \\ \{\mathbf{aa}, \mathbf{b}\}^* &= \{\varepsilon, \mathbf{aa}, \mathbf{b}, \mathbf{aaaa}, \mathbf{aab}, \ldots\} \end{aligned}$$

Formal languages

Let L and L' be languages over alphabet Σ

Concatenation: $L \cdot L' = LL' = \{u \cdot v : u \in L, v \in L'\}$

Exponentiation: $L^0 = \{\varepsilon\}$, $L^{k+1} = L^k \cdot L$

Iteration: $L^* = L^0 \cup L^1 \cup L^2 \cup \cdots$

Complement: $\overline{L} = \Sigma^* \setminus L$

e.g.
$$\begin{aligned}\{\mathbf{aa}, \mathbf{bb}\} \cdot \{\mathbf{c}, \mathbf{d}\} &= \{\mathbf{aac}, \mathbf{aad}, \mathbf{bbc}, \mathbf{bbd}\}, \\ \{\mathbf{aa}, \mathbf{b}\}^* &= \{\varepsilon, \mathbf{aa}, \mathbf{b}, \mathbf{aaaa}, \mathbf{aab}, \ldots\}\end{aligned}$$

Formal languages

Let L and L' be languages over alphabet Σ

Concatenation: $L \cdot L' = LL' = \{u \cdot v : u \in L, v \in L'\}$

Exponentiation: $L^0 = \{\varepsilon\}$, $L^{k+1} = L^k \cdot L$

Iteration: $L^* = L^0 \cup L^1 \cup L^2 \cup \cdots$

Complement: $\overline{L} = \Sigma^* \setminus L$

e.g.
$$\begin{aligned} \{\mathsf{aa}, \mathsf{bb}\} \cdot \{\mathsf{c}, \mathsf{d}\} &= \{\mathsf{aac}, \mathsf{aad}, \mathsf{bbc}, \mathsf{bbd}\}, \\ \{\mathsf{aa}, \mathsf{b}\}^* &= \{\varepsilon, \mathsf{aa}, \mathsf{b}, \mathsf{aaaa}, \mathsf{aab}, \ldots\} \end{aligned}$$

Formal languages

Let L and L' be languages over alphabet Σ

Concatenation: $L \cdot L' \;=\; LL' \;=\; \{u \cdot v : u \in L, v \in L'\}$

Exponentiation: $L^0 = \{\varepsilon\}$, $L^{k+1} = L^k \cdot L$

Iteration: $L^* \;=\; L^0 \;\cup\; L^1 \;\cup\; L^2 \;\cup\; \cdots$

Complement: $\overline{L} = \Sigma^* \setminus L$

e.g. $\{\mathbf{aa}, \mathbf{bb}\} \cdot \{\mathbf{c}, \mathbf{d}\} \;=\; \{\mathbf{aac}, \mathbf{aad}, \mathbf{bbc}, \mathbf{bbd}\}$,

$\{\mathbf{aa}, \mathbf{b}\}^* \;=\; \{\varepsilon, \mathbf{aa}, \mathbf{b}, \mathbf{aaaa}, \mathbf{aab}, \ldots\}$

Formal languages

Let L and L' be languages over alphabet Σ

Concatenation: $L \cdot L' = LL' = \{u \cdot v : u \in L, v \in L'\}$

Exponentiation: $L^0 = \{\varepsilon\}$, $L^{k+1} = L^k \cdot L$

Iteration: $L^* = L^0 \cup L^1 \cup L^2 \cup \cdots$

Complement: $\overline{L} = \Sigma^* \setminus L$

e.g.

$$\{\mathbf{aa}, \mathbf{bb}\} \cdot \{\mathbf{c}, \mathbf{d}\} = \{\mathbf{aac}, \mathbf{aad}, \mathbf{bbc}, \mathbf{bbd}\},$$
$$\{\mathbf{aa}, \mathbf{b}\}^* = \{\varepsilon, \mathbf{aa}, \mathbf{b}, \mathbf{aaaa}, \mathbf{aab}, \ldots\}$$

Formal languages

Let L and L' be languages over alphabet Σ

Concatenation: $L \cdot L' = LL' = \{u \cdot v : u \in L, v \in L'\}$

Exponentiation: $L^0 = \{\varepsilon\}$, $L^{k+1} = L^k \cdot L$

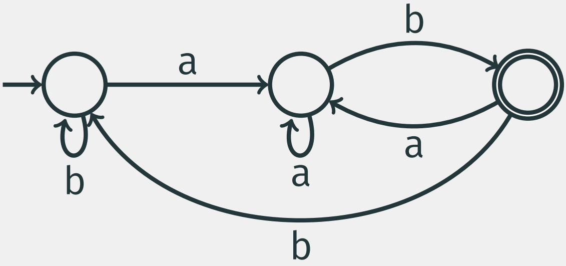
Iteration: $L^* = L^0 \cup L^1 \cup L^2 \cup \cdots$

Complement: $\overline{L} = \Sigma^* \setminus L$

e.g. $$\begin{aligned} \{\mathsf{aa}, \mathsf{bb}\} \cdot \{\mathsf{c}, \mathsf{d}\} &= \{\mathsf{aac}, \mathsf{aad}, \mathsf{bbc}, \mathsf{bbd}\}, \\ \{\mathsf{aa}, \mathsf{b}\}^* &= \{\varepsilon, \mathsf{aa}, \mathsf{b}, \mathsf{aaaa}, \mathsf{aab}, \ldots\} \end{aligned}$$

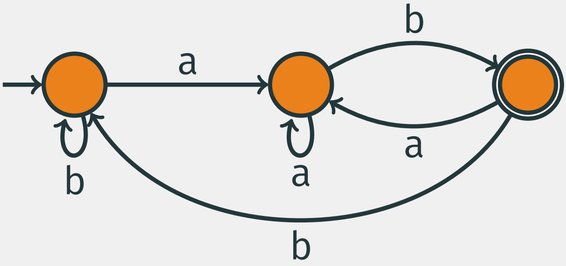
Deterministic finite automata (DFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to Q$
- *Initial state*: $q_0 \in Q$
- *Final states*: $F \subseteq Q$



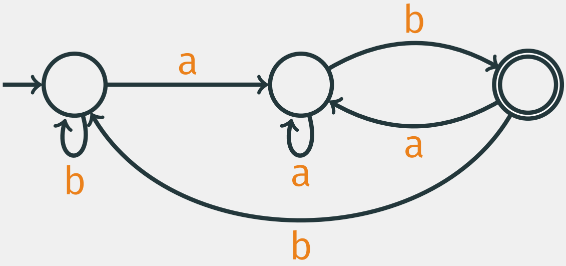
Deterministic finite automata (DFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to Q$
- *Initial state*: $q_0 \in Q$
- *Final states*: $F \subseteq Q$



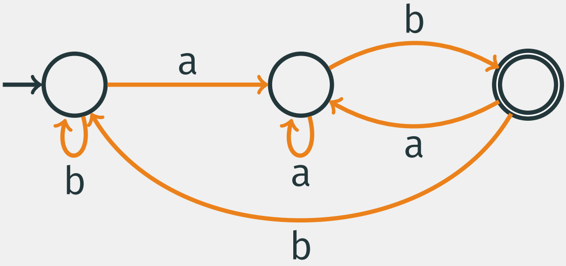
Deterministic finite automata (DFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to Q$
- *Initial state*: $q_0 \in Q$
- *Final states*: $F \subseteq Q$



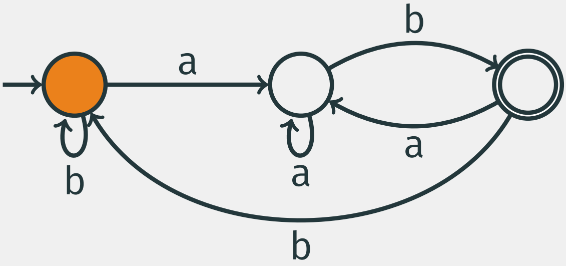
Deterministic finite automata (DFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to Q$
- *Initial state*: $q_0 \in Q$
- *Final states*: $F \subseteq Q$



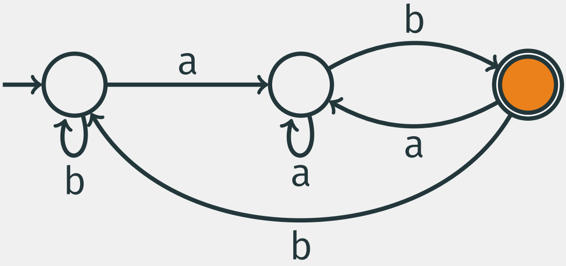
Deterministic finite automata (DFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to Q$
- *Initial state*: $q_0 \in Q$
- *Final states*: $F \subseteq Q$



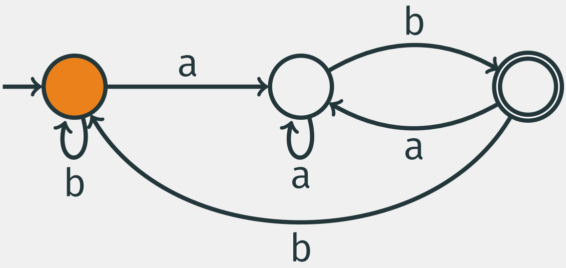
Deterministic finite automata (DFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to Q$
- *Initial state*: $q_0 \in Q$
- *Final states*: $F \subseteq Q$



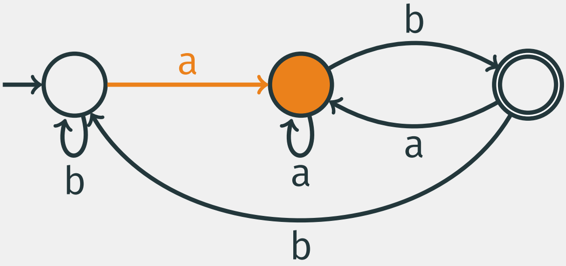
Deterministic finite automata (DFA)

w = aabab



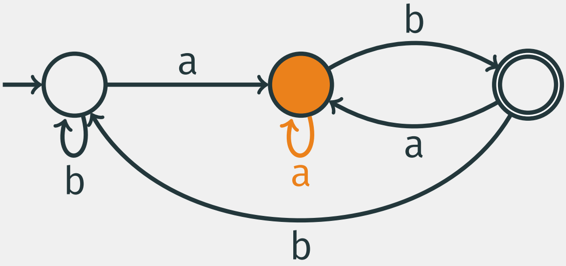
Deterministic finite automata (DFA)

w = aabab



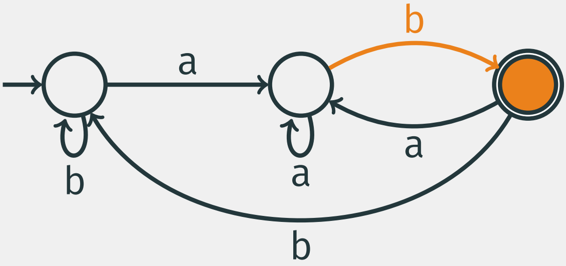
Deterministic finite automata (DFA)

w = aabab

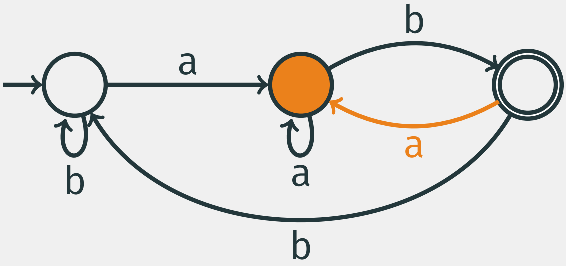


Deterministic finite automata (DFA)

w = aababb

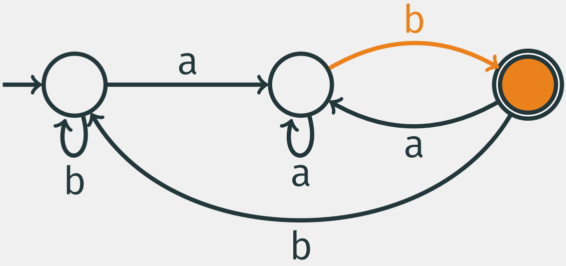


w = aabab



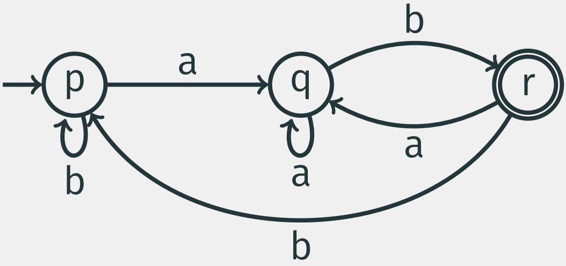
Deterministic finite automata (DFA)

w = aabab

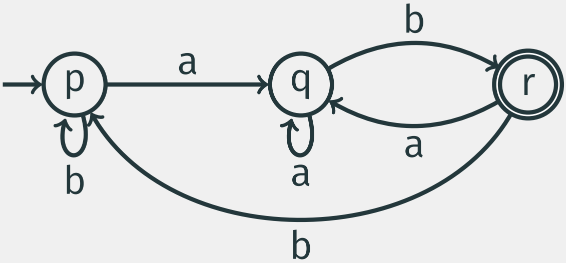


Deterministic finite automata (DFA)

$$p \xrightarrow{a} q \xrightarrow{a} q \xrightarrow{b} r \xrightarrow{a} q \xrightarrow{b} r$$

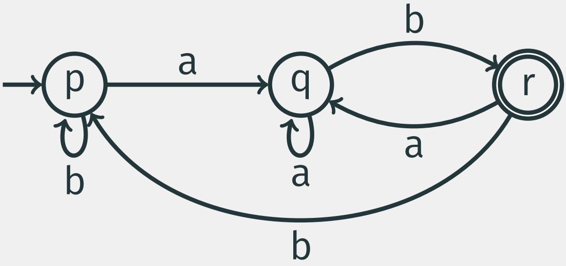


Deterministic finite automata (DFA)



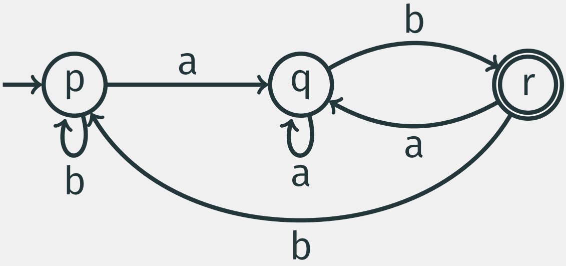
Deterministic finite automata (DFA)

$$L(A) = \{w \in \Sigma^* : \exists q \in F \text{ s.t. } q_0 \xrightarrow{w} q\}$$

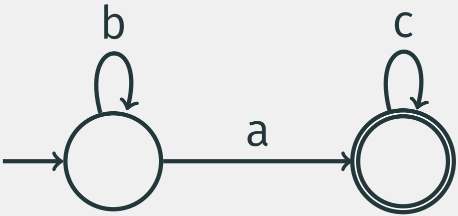


Deterministic finite automata (DFA)

$$L(A) = \{w \in \Sigma^* : \quad w \text{ ends with } \mathbf{ab} \quad \}$$

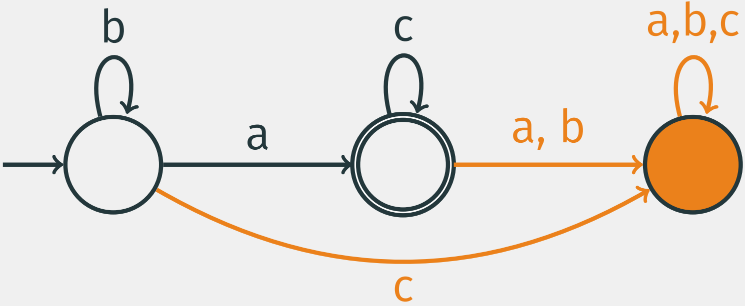


Transition function δ defined *on every* input



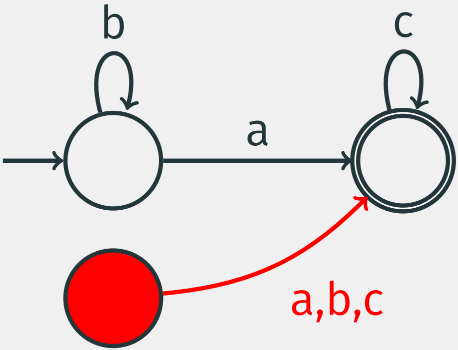
DFA: trap states and unreachable states

Transition function δ defined *on every* input



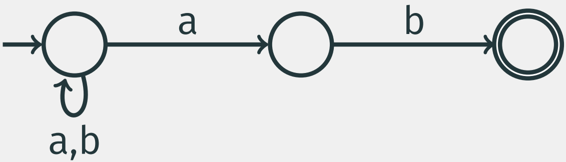
DFA: trap states and unreachable states

Every state *reachable* from initial state



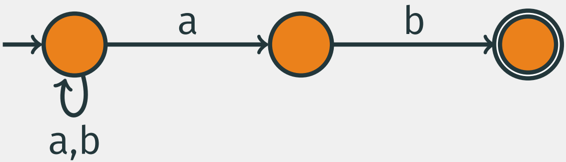
Nondeterministic finite automata (NFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$
- *Initial states*: $\emptyset \subset Q_0 \subseteq Q$
- *Final states*: $F \subseteq Q$



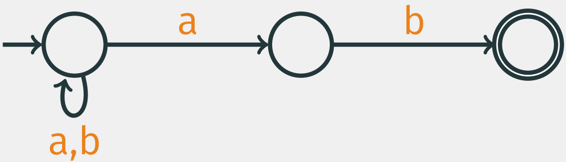
Nondeterministic finite automata (NFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$
- *Initial states*: $\emptyset \subset Q_0 \subseteq Q$
- *Final states*: $F \subseteq Q$



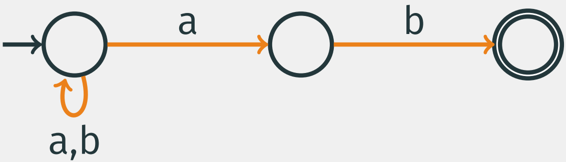
Nondeterministic finite automata (NFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$
- *Initial states*: $\emptyset \subset Q_0 \subseteq Q$
- *Final states*: $F \subseteq Q$



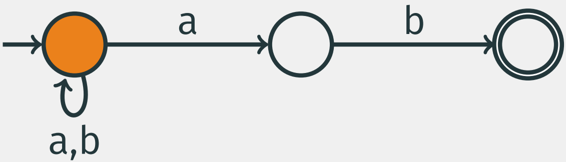
Nondeterministic finite automata (NFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$
- *Initial states*: $\emptyset \subset Q_0 \subseteq Q$
- *Final states*: $F \subseteq Q$



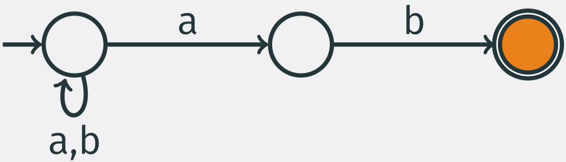
Nondeterministic finite automata (NFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$
- *Initial states*: $\emptyset \subset Q_0 \subseteq Q$
- *Final states*: $F \subseteq Q$



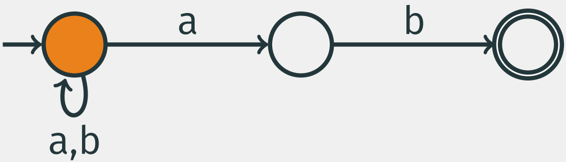
Nondeterministic finite automata (NFA)

- *States*: nonempty finite set Q
- *Alphabet*: nonempty finite set Σ
- *Transitions*: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$
- *Initial states*: $\emptyset \subset Q_0 \subseteq Q$
- *Final states*: $F \subseteq Q$



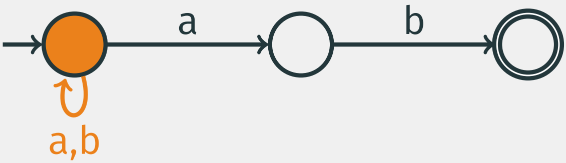
Nondeterministic finite automata (NFA)

w = aab



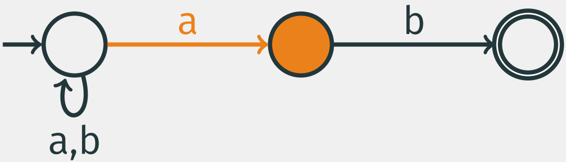
Nondeterministic finite automata (NFA)

w = aab



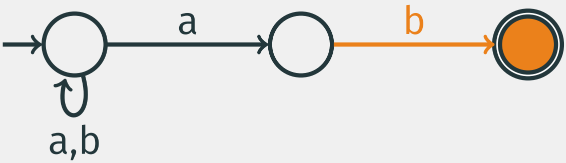
Nondeterministic finite automata (NFA)

$$w = aab$$

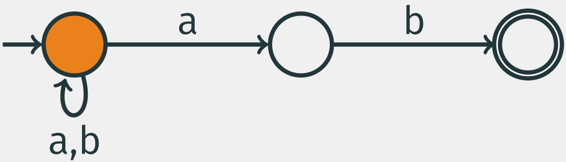


Nondeterministic finite automata (NFA)

$$w = aab$$

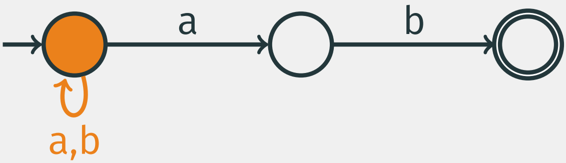


w = aab



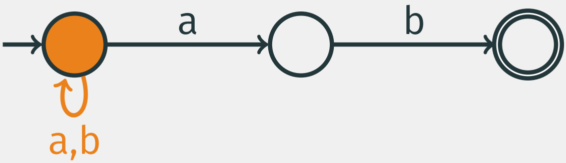
Nondeterministic finite automata (NFA)

$w = aab$



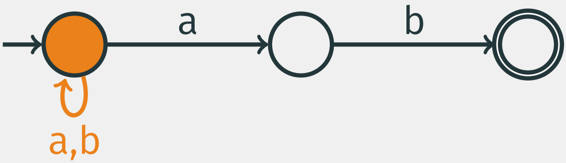
Nondeterministic finite automata (NFA)

$w = aab$



Nondeterministic finite automata (NFA)

w = aab

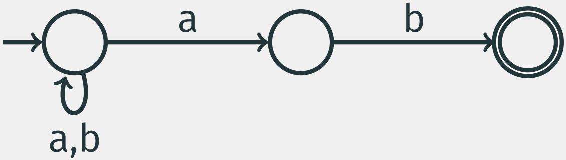


Nondeterministic finite automata (NFA)

$$p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} p_n$$

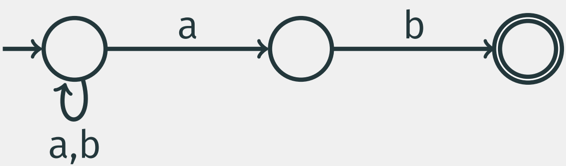
$$\Updownarrow$$

$$p_i \in \delta(p_{i-1}, a_i) \text{ for every } 0 < i \leq n$$



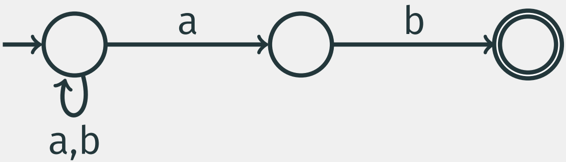
Nondeterministic finite automata (NFA)

$$L(A) = \{w \in \Sigma^* : \exists q_0 \in Q_0, q \in F \text{ s.t. } q_0 \xrightarrow{w} q\}$$



Nondeterministic finite automata (NFA)

$$L(A) = \{w \in \Sigma^* : \quad w \text{ ends with } \mathbf{ab} \quad \}$$



Regular expressions

$$r ::= \emptyset \mid \varepsilon \mid a \mid r_1 r_2 \mid r_1 + r_2 \mid r^*$$

Regular expressions

$$r \ ::= \ \emptyset \ \mid \ \varepsilon \ \mid \ a \ \mid \ r_1\, r_2 \ \mid \ r_1 + r_2 \ \mid \ r^*$$

$$\begin{array}{lclclcl} L(\emptyset) & = & \emptyset & \quad & L(r_1\, r_2) & = & L(r_1) \cdot L(r_2) \\ L(\varepsilon) & = & \{\varepsilon\} & & L(r_1 + r_2) & = & L(r_1) \cup L(r_2) \\ L(a) & = & \{a\} & & L(r^*) & = & L(r)^* \end{array}$$

Regular expressions

$$L(\,(a+b)^*ab\,) = \{w \in \{a,b\}^* : w \text{ ends with } \texttt{ab}\}$$

$$\begin{aligned}
L(\emptyset) &= \emptyset & \qquad L(r_1\,r_2) &= L(r_1)\cdot L(r_2) \\
L(\varepsilon) &= \{\varepsilon\} & \qquad L(r_1+r_2) &= L(r_1)\cup L(r_2) \\
L(a) &= \{a\} & \qquad L(r^*) &= L(r)^*
\end{aligned}$$

More examples

$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* : w \text{ contains } \mathtt{aaa}\}$$

More examples

$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* : w \text{ contains } \mathtt{aaa}\}$$

Regular expression?

More examples

$$L = \{w \in \{\mathbf{a}, \mathbf{b}\}^* : w \text{ contains } \mathbf{aaa}\}$$

Regular expression?

$$(\mathbf{a} + \mathbf{b})^* \mathbf{aaa} (\mathbf{a} + \mathbf{b})^*$$

More examples

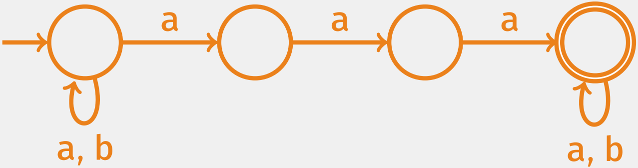
$$L = \{w \in \{\mathtt{a}, \mathtt{b}\}^* : w \text{ contains } \mathtt{aaa}\}$$

NFA?

More examples

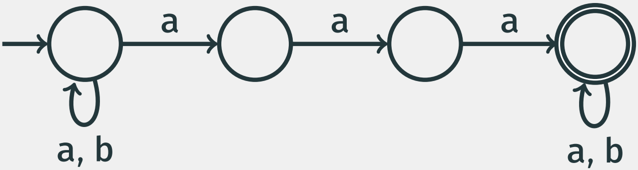
$$L = \{w \in \{\mathbf{a}, \mathbf{b}\}^* : w \text{ contains } \mathbf{aaa}\}$$

NFA?



$$L = \{w \in \{\mathbf{a}, \mathbf{b}\}^* : w \text{ contains } \mathbf{aaa}\}$$

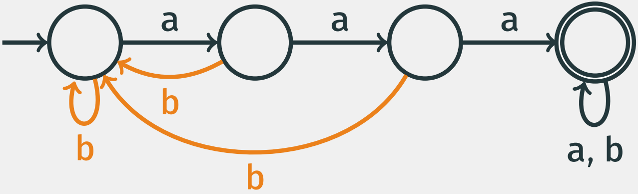
DFA?



More examples

$$L = \{w \in \{\mathbf{a}, \mathbf{b}\}^* : w \text{ contains } \mathbf{aaa}\}$$

DFA?



More examples

$$L = \{w \in \{\mathtt{0}, \mathtt{1}\}^* : w \text{ contains an even number of } \mathtt{0} \text{ or an odd number of } \mathtt{1}\}$$

More examples

$$L = \{w \in \{\mathtt{0}, \mathtt{1}\}^* : w \text{ contains an even number of } \mathtt{0} \text{ or an odd number of } \mathtt{1}\}$$

Regular expression?

More examples

$$L = \{w \in \{\mathtt{0}, \mathtt{1}\}^* : w \text{ contains an even number of } \mathtt{0} \text{ or an odd number of } \mathtt{1}\}$$

Regular expression?

$$(1^*01^*0)^*1^* + (0^*10^*1^*)^*0^*1^*0^*$$

More examples

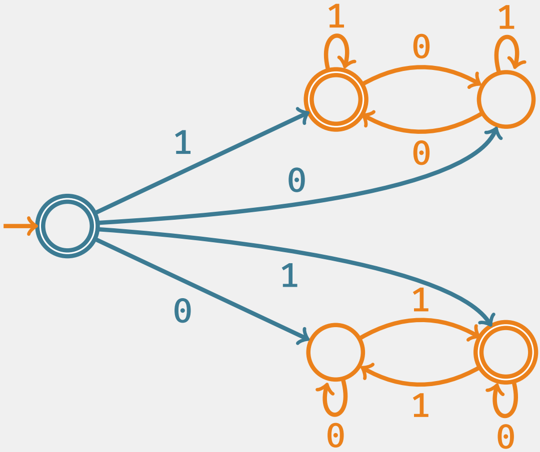
$$L = \{w \in \{\mathtt{0}, \mathtt{1}\}^* : w \text{ contains an even number of } \mathtt{0} \text{ or an odd number of } \mathtt{1}\}$$

NFA?

More examples

$$L = \{w \in \{\mathtt{0}, \mathtt{1}\}^* : w \text{ contains an even number of } \mathtt{0} \text{ or an odd number of } \mathtt{1}\}$$

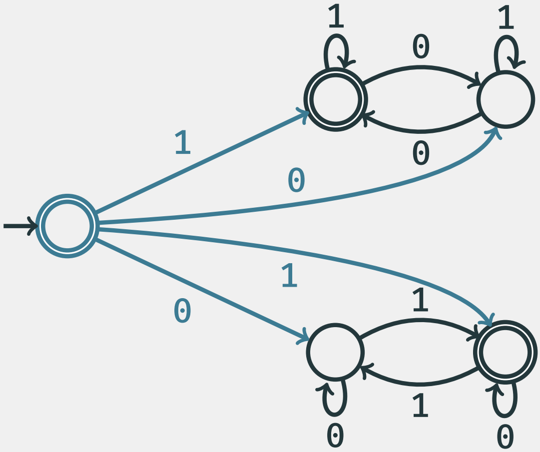
NFA?



More examples

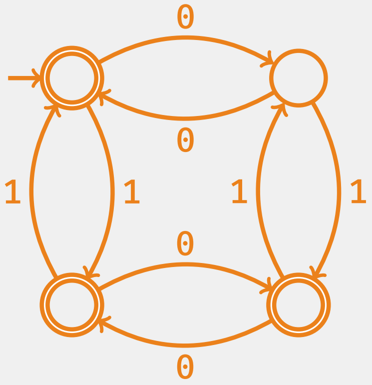
$L = \{w \in \{\mathtt{0}, \mathtt{1}\}^* : w$ contains an even number of $\mathtt{0}$ or an odd number of $\mathtt{1}$ $\}$

DFA?

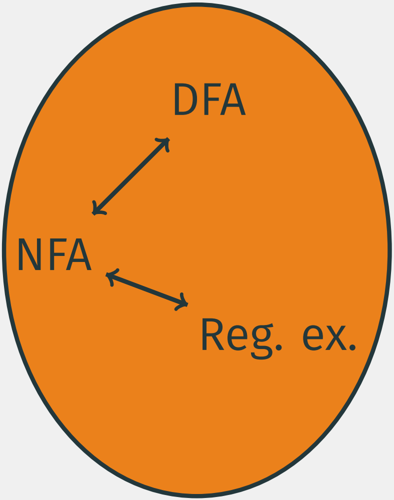


$$L = \{ w \in \{\mathtt{0}, \mathtt{1}\}^* : w \text{ contains an even number of } \mathtt{0} \text{ or an odd number of } \mathtt{1} \}$$

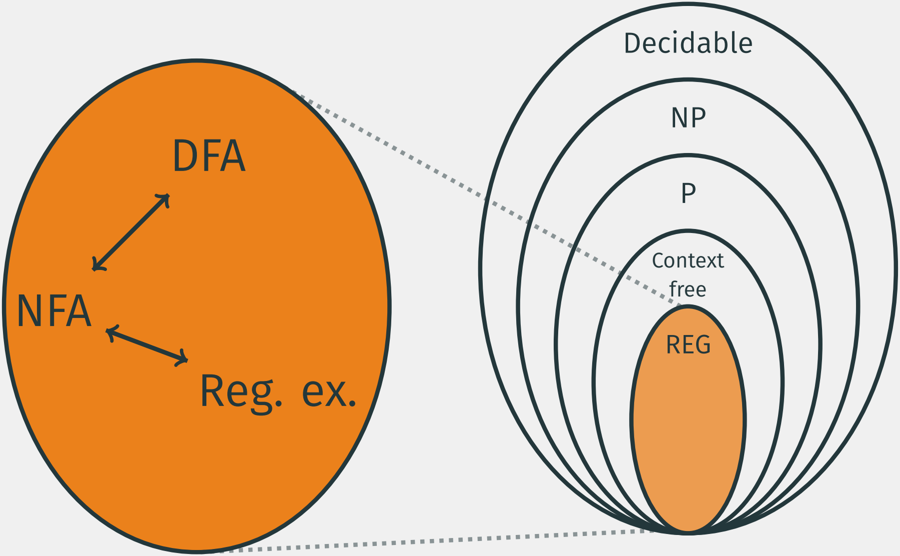
DFA?



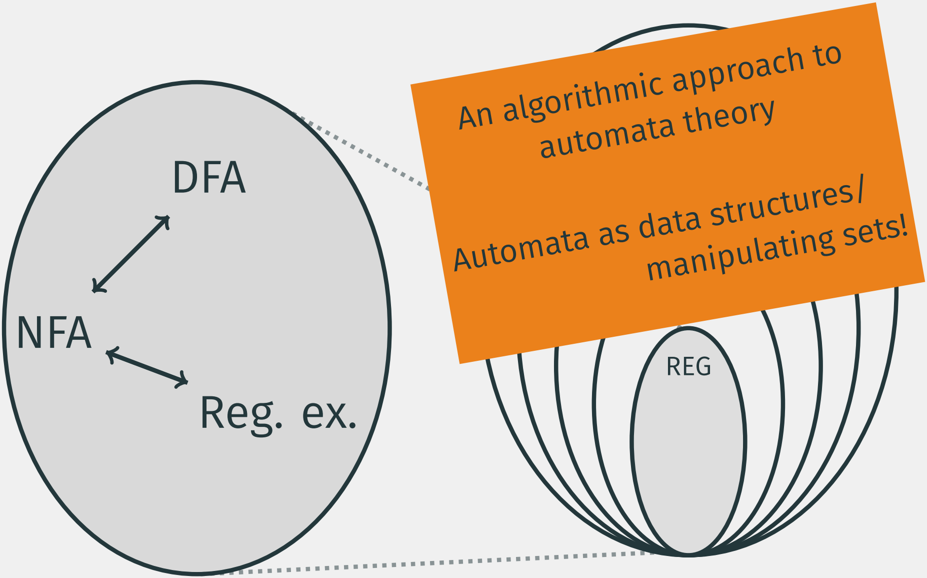
Regular languages



Regular languages



Regular languages

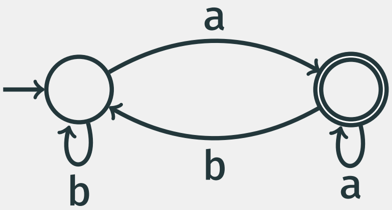


Beyond finite words

Büchi automata

An *infinite word* is an infinite sequence $a_0 a_1 a_2 \cdots$ over some Σ

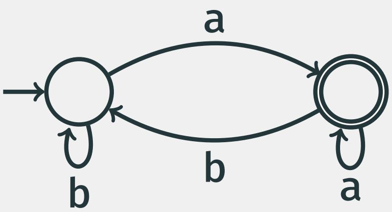
A *Büchi automaton* is ”as an NFA”, but accepts infinite words



Büchi automata

An *infinite word* is an infinite sequence $a_0 a_1 a_2 \cdots$ over some Σ

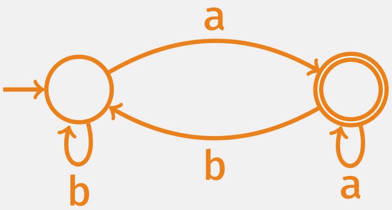
A *Büchi automaton* is ”as an NFA”, but accepts infinite words



Büchi automata

An *infinite word* is an infinite sequence $a_0 a_1 a_2 \cdots$ over some Σ

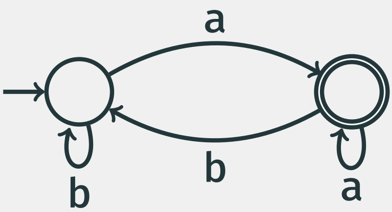
A *Büchi automaton* is ”as an NFA”, but accepts infinite words



Büchi automata

An *infinite word* is an infinite sequence $a_0 a_1 a_2 \cdots$ over some Σ

A *Büchi automaton* is ”as an NFA”, but accepts infinite words



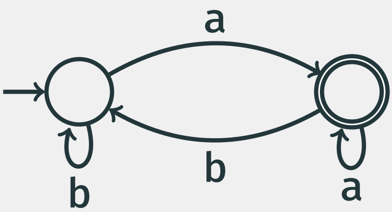
$L_\omega(A) = \{w \in \{\mathbf{a}, \mathbf{b}\}^\omega : w \text{ contains infinitely many } \mathbf{a}\}$

Büchi automata

An *in*… over some Σ

Coming later this semester!

A *Bü*… ite words



$$L_\omega(A) = \{w \in \{\mathbf{a}, \mathbf{b}\}^\omega : w \text{ contains infinitely many } \mathbf{a}\}$$