CS 228 : Logic in Computer Science

S. Krishna

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

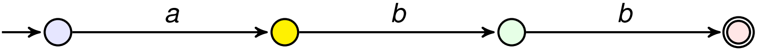
- Contains *abb*

(a+b)^* abb (a+b)^*

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Contains *abb*

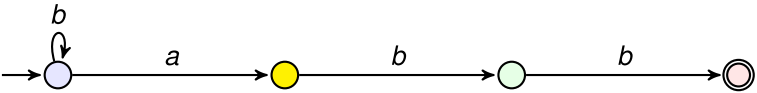


$$\exists x \exists y \exists z (Q_a(x) \wedge Q_b(y) \wedge Q_b(z) \wedge S(x, y) \wedge S(y, z))$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Contains *abb*

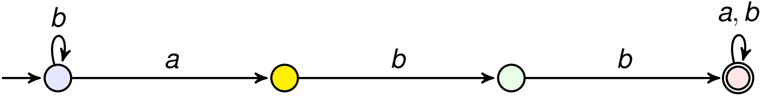


$$\exists x \exists y \exists z (Q_a(x) \land Q_b(y) \land Q_b(z) \land S(x, y) \land S(y, z))$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Contains *abb*

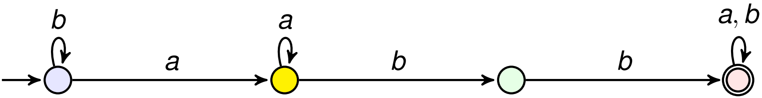


$$\exists x \exists y \exists z (Q_a(x) \wedge Q_b(y) \wedge Q_b(z) \wedge S(x, y) \wedge S(y, z))$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Contains *abb*

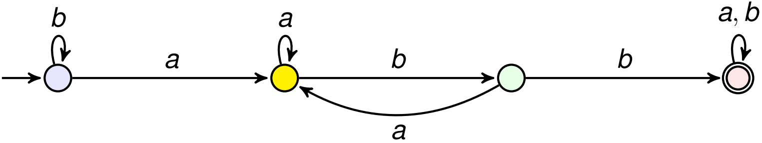


$$\exists x \exists y \exists z (Q_a(x) \wedge Q_b(y) \wedge Q_b(z) \wedge S(x, y) \wedge S(y, z))$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

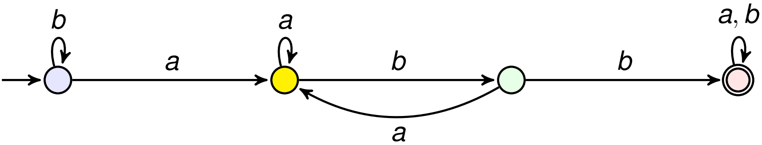
- Contains *abb*



Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Contains *abb*



$$\exists x \exists y \exists z (Q_a(x) \wedge Q_b(y) \wedge Q_b(z) \wedge S(x, y) \wedge S(y, z))$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

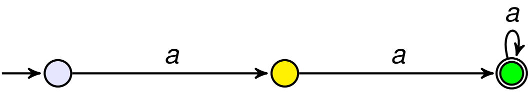
- Right before the last position is an a :

wa-

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Right before the last position is an a :
  Examples : ab, $babbaa$, $bbab$
  Non examples : ba, bb, aba

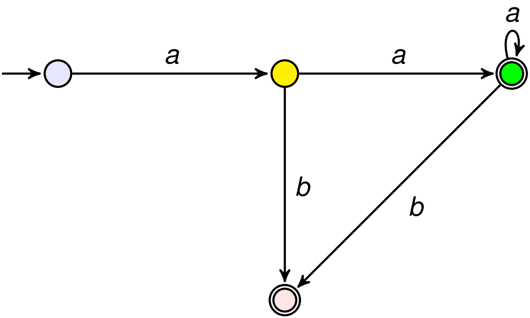


$$\exists x[Q_a(x) \wedge \exists y(S(x, y) \wedge \forall z(z \leqslant y))]$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Right before the last position is an a :
  Examples : ab, $babbaa$, $bbab$
  Non examples : ba, bb, aba

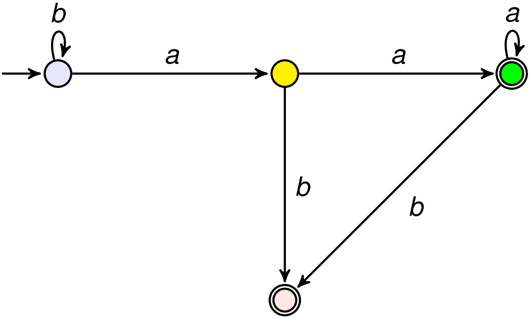


$$\exists x[Q_a(x) \wedge \exists y(S(x, y) \wedge \forall z(z \leqslant y))]$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Right before the last position is an a :
  Examples : ab, $babbaa$, $bbab$
  Non examples : ba, bb, aba

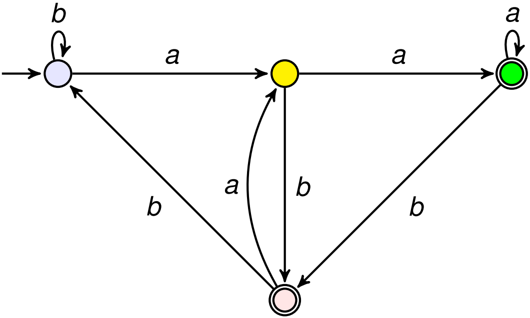


$$\exists x[Q_a(x) \wedge \exists y(S(x,y) \wedge \forall z(z \leqslant y))]$$

Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

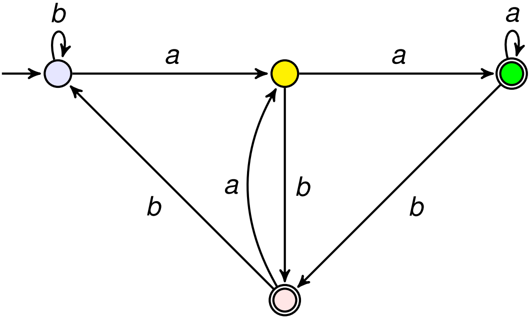
- Right before the last position is an a :
  Examples : ab, $babbaa$, $bbab$
  Non examples : ba, bb, aba



Is it Regular? Is it FO-definable?

$\Sigma = \{a, b\}$. Consider the following languages $L \subseteq \Sigma^*$:

- Right before the last position is an a :
  Examples : $ab, babbaa, bbab$
  Non examples : ba, bb, aba



$$\exists x[Q_a(x) \wedge \exists y(S(x, y) \wedge \forall z(z \leqslant y))]$$

Deterministic Finite Automata

- Every state on every symbol goes to a unique state

Deterministic Finite Automata

- Every state on every symbol goes to a unique state
  - $\delta : Q \times \Sigma \to Q$ is a transition function
- Given a string $w \in \Sigma^*$ and a state $q \in Q$, iteratively apply δ

Deterministic Finite Automata

- Every state on every symbol goes to a unique state
  - $\delta : Q \times \Sigma \to Q$ is a transition function
- Given a string $w \in \Sigma^*$ and a state $q \in Q$, iteratively apply δ
  - $w = aab$
  - $\delta(q, a) = q_1$, $\delta(\delta(q, a), a) = \delta(q_1, a) = q_2$,

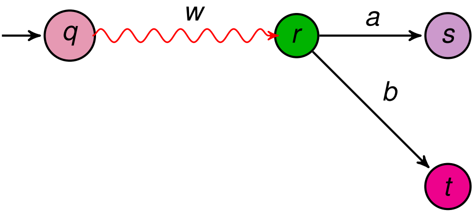
Deterministic Finite Automata

- Every state on every symbol goes to a unique state
  - $\delta : Q \times \Sigma \to Q$ is a transition function
- Given a string $w \in \Sigma^*$ and a state $q \in Q$, iteratively apply δ
  - $w = aab$
  - $\delta(q, a) = q_1$, $\delta(\delta(q, a), a) = \delta(q_1, a) = q_2$,
    $\delta(\delta(\delta(q, a), a), b) = \delta(\delta(q_1, a), b) = \delta(q_2, b) = q_3$

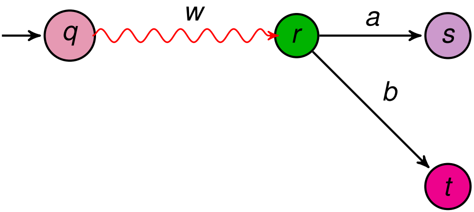
Deterministic Finite Automata

- Every state on every symbol goes to a unique state
  - $\delta : Q \times \Sigma \to Q$ is a transition function
- Given a string $w \in \Sigma^*$ and a state $q \in Q$, iteratively apply δ
  - $w = aab$
  - $\delta(q, a) = q_1$, $\delta(\delta(q, a), a) = \delta(q_1, a) = q_2$, $\delta(\delta(\delta(q, a), a), b) = \delta(\delta(q_1, a), b) = \delta(q_2, b) = q_3$
  - $\hat{\delta} : Q \times \Sigma^* \to Q$ extension of δ to strings
    - $\hat{\delta}(q, \epsilon) = q$
    - $\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a)$

DFA : Transition Function on Words



DFA : Transition Function on Words



- $\hat{\delta}(q, wa) = s = \delta(\hat{\delta}(q, w), a) = \delta(r, a)$
- $\hat{\delta}(q, wb) = t = \delta(\hat{\delta}(q, w), b) = \delta(r, b)$