Lecture 8 – Closure Properties of Regular Languages

COSE215: Theory of Computation

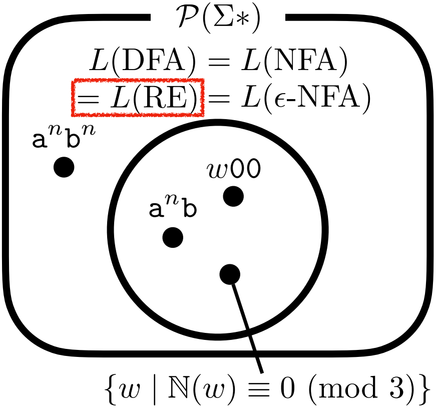
Jihyeok Park

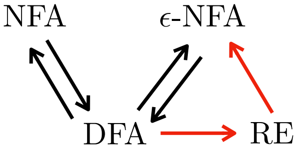
PLRG

2023 Spring

Recall

- Regular Languages





Contents

Closure Properties of Regular Languages

PLRG

Definition (Closure Properties)

The class of regular languages is **closed** under an n-ary operator op if and only if $\text{op}(L_1, \cdots, L_n)$ is regular for any regular languages $L_1, \cdots, L_n$. We say that such properties are **closure properties** of regular languages.

$$
\begin{array}{lcl}
\text{A language } L \text{ is regular} & \iff & \exists \text{ RE } R.\ L(R) = L \\
\text{A language } L \text{ is regular} & \iff & \exists\ \epsilon\text{-NFA } N_\epsilon.\ L(N_\epsilon) = L \\
\text{A language } L \text{ is regular} & \iff & \exists \text{ NFA } N.\ L(N) = L \\
\text{A language } L \text{ is regular} & \iff & \exists \text{ DFA } D.\ L(D) = L
\end{array}
$$

1. Construct a regular expression R such that $L(R) = \text{op}(L_1, \cdots, L_n)$ using the regular expressions $R_1, \cdots, R_n$ such that $L(R_i) = L_i$ for $i = 1, \cdots, n$.
2. Construct a finite automaton A such that $L(A) = \text{op}(L_1, \cdots, L_n)$ using the finite automata $A_1, \cdots, A_n$ such that $L(A_i) = L_i$ for $i = 1, \cdots, n$.

Closure under Union

Theorem (Closure under Union)

If L_1 and L_2 are regular languages, then so is $L_1 \cup L_2$.

Proof) Let R_1 and R_2 be the regular expressions such that $L(R_1) = L_1$ and $L(R_2) = L_2$, respectively. Consider the following regular expression:

$$R_1 \mid R_2$$

Then, by the definition of the union operator ($\mid$), $L(R_1 \mid R_2) = L_1 \cup L_2$. $\square$

Closure under Concatenation and Kleene Star

PLRG

Theorem (Closure under Concatenation)

If L_1 and L_2 are regular languages, then so is $L_1 \cdot L_2$.

Proof) Let R_1 and R_2 be the regular expressions such that $L(R_1) = L_1$ and $L(R_2) = L_2$, respectively. Consider the following regular expression:

$$R_1 \cdot R_2$$

Then, by the definition of the concatenation operator $(\cdot)$,
$L(R_1 \cdot R_2) = L_1 \cup L_2$. $\square$

Theorem (Closure under Kleene Star)

If L is a regular language, then so is L^*.

Proof) Let R be the regular expressions such that $L(R) = L$. Consider the following regular expression:

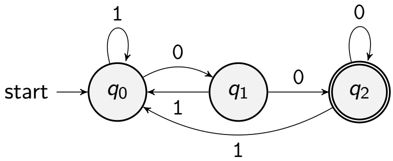
$$R^*$$

Then, by the definition of the Kleene star operator $(^*)$, $L(R^*) = L^*$. $\square$

Closure under Complement

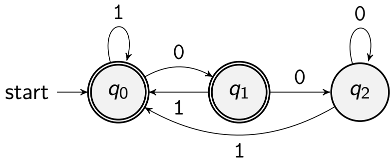
Consider the following DFA D such that $L(D) = \{w00 \mid w \in \{0,1\}^*\}$.

$D =$



How to construct a DFA $\overline{D}$ such that $L(\overline{D}) = \overline{L(D)}$?

$\overline{D} =$



Closure under Complement

Theorem (Closure under Complement)

If L is a regular language, then so is $\overline{L}$.

Proof) Let $D = (Q, \Sigma, \delta, q_0, F)$ be the DFA such that $L(D) = L$. Consider the following DFA:

$$\overline{D} = (Q, \Sigma, \delta, q_0, Q \setminus F).$$

Then,

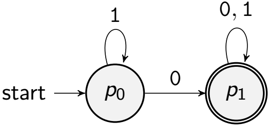
$$\begin{aligned}
\forall w \in \Sigma^*,\ w \in L(\overline{D}) \quad &\iff \quad \delta^*(q_0, w) \in Q \setminus F \\
&\iff \quad \delta^*(q_0, w) \notin F \\
&\iff \quad w \notin L(D) \\
&\iff \quad w \notin L \\
&\iff \quad w \in \overline{L}
\end{aligned}$$

$\square$

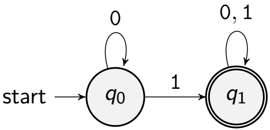
Closure under Intersection

Consider two DFA D_0 and D_1 such that $L(D_0) = \{w \in \{0,1\}^* \mid w \text{ has } 0\}$ and $L(D_1) = \{w \in \{0,1\}^* \mid w \text{ has } 1\}$, respectively.

$D_0 =$

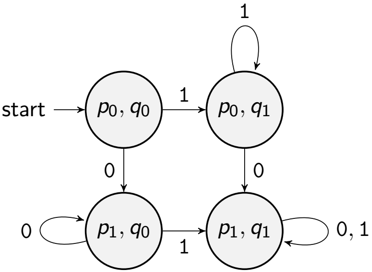


$D_1 =$



How to construct a DFA D such that $L(D) = L(D_0) \cap L(D_1)$?

$D =$



Closure under Intersection

Theorem (Closure under Intersection)

If L_0 and L_1 are regular languages, then so is $L_0 \cap L_1$.

Proof) Let $D_0 = (Q_0, \Sigma, \delta_0, q_0, F_0)$ and $D_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ be the DFA such that $L(D_0) = L_0$ and $L(D_1) = L_1$. Consider the following DFA:

$$D = (Q_0 \times Q_1, \Sigma, \delta, (q_0, q_1), F_0 \times F_1).$$

where $\forall q \in Q_0, q' \in Q_1, \mathrm{a} \in \Sigma.\ \delta((q, q'), \mathrm{a}) = (\delta_0(q, \mathrm{a}), \delta_1(q', \mathrm{a}))$. Then,

$$\begin{array}{rcl}
\forall w \in \Sigma^*,\ w \in L(D) & \iff & \delta^*((q_0, q_1), w) \in F_0 \times F_1 \\
& \iff & \delta^*(q_0, w) \in F_0 \text{ and } \delta^*(q_1, w) \in F_1 \\
& \iff & w \in L(D_0) \text{ and } w \in L(D_1) \\
& \iff & w \in L(D_0) \cap L(D_1) \\
& \iff & w \in L_0 \cap L_1
\end{array}$$

$\square$

Closure under Intersection

Theorem (Closure under Intersection)

If L_0 and L_1 are regular languages, then so is $L_0 \cap L_1$.

Proof) Another proof is to use De Morgan's law:

$$L_0 \cap L_1 = \overline{\overline{L_0} \cup \overline{L_1}}$$

Since we already know that the regular languages are closed under complement and union, we are done. $\square$

Closure under Difference

Theorem (Closure under Difference)

If L_0 and L_1 are regular languages, then so is $L_0 \setminus L_1$.

Proof) Similarly, we can use the following fact:

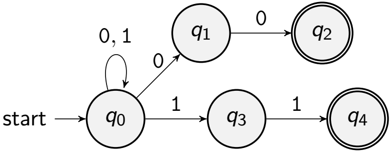
$$L_0 \setminus L_1 = L_0 \cap \overline{L_1}$$

Since we already know that the regular languages are closed under complement and intersection, we are done. $\square$

Closure under Reversal

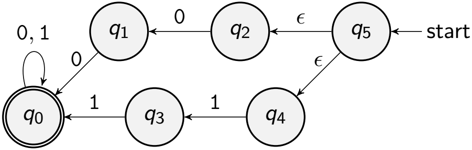
Consider the following ϵ-NFA N_ϵ such that $L(N_\epsilon) = \{w0 \text{ or } w1 \mid w \in \{0,1\}^*\}$:

$N_\epsilon =$



How to construct an ϵ-NFA ${N_\epsilon}^R$ such that $L({N_\epsilon}^R) = L(N_\epsilon)^R$?

${N_\epsilon}^R =$



Closure under Reversal

Theorem (Closure under Reversal)

If L is a regular language, then so is L^R.

Proof) Let $N_\epsilon = (Q, \Sigma, \delta, q_0, F)$ be the ϵ-NFA such that $L(N_\epsilon) = L$. Consider the following

$$N_\epsilon{}^R = (Q \uplus \{q_s\}, \Sigma, \delta^R, q_s, \{q_0\})$$

where

$$\begin{aligned}
\forall q \in Q.\ \forall a \in \Sigma.\ \delta^R(q, \mathrm{a}) &= \{q' \in Q \mid q \in \delta(q', \mathrm{a})\} \\
\forall q \in Q.\ \delta^R(q, \epsilon) &= \{q' \in Q \mid q \in \delta(q', \epsilon)\} \\
\forall a \in \Sigma.\ \delta^R(q_s, \mathrm{a}) &= \varnothing \\
\delta^R(q_s, \epsilon) &= F
\end{aligned}$$

$\square$

Theorem (Closure under Reversal)

If L is a regular language, then so is L^R.

Proof) Another proof is to use the structural induction on the regular expressions. Let R be a regular expression. Then, we define its reverse R^R as follows:

- If $R = \varnothing$, then $R^R = \varnothing$.
- If $R = \epsilon$, then $R^R = \epsilon$.
- If $R = \mathrm{a}$, then $R^R = \mathrm{a}$.
- If $R = R_0 \,|\, R_1$, then $R^R = R_0^R \,|\, R_1^R$.
- If $R = R_0 \cdot R_1$, then $R^R = R_1^R \cdot R_0^R$.
- If $R = R_0^*$, then $R^R = (R_0^R)^*$.
- If $R = (R_0)$, then $R^R = (R_0^R)$.

$$R = \mathtt{ab(cd)}^*\mathtt{|ef}$$

$$R^R = \mathtt{(dc)}^*\mathtt{ba|fe}$$

□

Closure under Homomorphism

Definition (Homomorphism)

Suppose Σ and Γ are two finite sets of symbols. Then, a function

$$h : \Sigma \to \Gamma^*$$

is called a **homomorphism**. For a given word $w = \mathtt{a}_1\mathtt{a}_2 \cdots \mathtt{a}_n$,

$$h(w) = h(\mathtt{a}_1)h(\mathtt{a}_2) \cdots h(\mathtt{a}_n)$$

For a language L,

$$h(L) = \{h(w) \mid w \in L\}$$

Example (Homomorphism)

Let $\Sigma = \{0, 1\}$, $\Gamma = \{\mathtt{a}, \mathtt{b}\}$, and $h(0) = \mathtt{ab}$, $h(1) = \mathtt{a}$. Then,

$$h(10) = \mathtt{aab} \qquad h(010) = \mathtt{abaab} \qquad h(1100) = \mathtt{aaabab}$$

Theorem (Closure under Homomorphism)

If h is a homomorphism and L is a regular language, then so is $h(L)$.

Proof) Let R be the regular expression such that $L(R) = L$. Then, we define its homomorphic regular expression $h(R)$ as follows:

- If $R = \varnothing$, then $h(R) = \varnothing$.
- If $R = \epsilon$, then $h(R) = \epsilon$.
- If $R = \mathtt{a}$, then $h(R) = h(\mathtt{a})$.
- If $R = R_0 \,|\, R_1$, then $h(R) = h(R_0) \,|\, h(R_1)$.
- If $R = R_0 \cdot R_1$, then $h(R) = h(R_1) \cdot h(R_0)$.
- If $R = R_0^*$, then $h(R) = (h(R_0))^*$.
- If $R = (R_0)$, then $h(R) = (h(R_0))$.

$$h(\mathtt{0}) = \mathtt{ab}$$
$$h(\mathtt{1}) = \mathtt{a}$$

$$R = \mathtt{0}(\mathtt{0}|\mathtt{1})^*\mathtt{0}^*$$

$$h(R) = (\mathtt{ab}(\mathtt{ab}|\mathtt{a})^*\mathtt{ab})^*$$

□

Summary

1. Closure Properties of Regular Languages
    - Union
    - Concatenation and Kleene Star
    - Complement
    - Intersection
    - Difference
    - Reversal
    - Homomorphism

Next Lecture

- The Pumping Lemma for Regular Languages

Jihyeok Park
jihyeok_park@korea.ac.kr
https://plrg.korea.ac.kr