Computer Science 351

Regular Operations and Regular Expressions

Instructor: Wayne Eberly

Department of Computer Science
University of Calgary

Lecture #7

Goal for Today

Goals for Today:

- Introduce the ***regular operations***.
- Introduce ***closure properties*** and their uses.
- Introduce ***regular expressions***.

The "Union" of Two Languages

Definition: Let Σ be an alphabet and let $A, B \subseteq \Sigma^\star$. The ***union*** of the languages A and B is the language

$$A \cup B = \{\omega \in \Sigma^\star \mid \omega \in A \text{ or } \omega \in B \text{ (or both)}\}.$$

- ***Example:*** Suppose that $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $A = \{\mathtt{a}\}$, and $B = \{\mathtt{bb}, \mathtt{c}\}$. Then

$$A \cup B = B \cup A = \{\mathtt{a}, \mathtt{bb}, \mathtt{c}\}.$$

The "Concatenation" of Two Languages

Definition: Let Σ be an alphabet and let $A, B \subseteq \Sigma^\star$. The ***concatenation*** of the languages A and B is the language

$$A \circ B = \{\omega_1 \cdot \omega_2 \mid \omega_1 \in A \text{ and } \omega_2 \in B\}.$$

- ***Example:*** Suppose that $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $A = \{\mathtt{a}\}$, and $B = \{\mathtt{bb}, \mathtt{c}\}$. Then

$$A \circ B = \{\mathtt{abb}, \mathtt{ac}\} \quad \text{and} \quad B \circ A = \{\mathtt{bba}, \mathtt{ca}\}.$$

The Kleene Star of a Language

Definition: Let Σ be an alphabet and let $A \subseteq \Sigma^\star$. The ***Kleene star*** of the language A is the language

$$A^\star = \{\omega_1 \cdot \omega_2 \dots \omega_k \mid k \geq 0 \text{ and } \omega_i \in A \text{ for } 1 \leq i \leq k\}$$

This language is also, sometimes called the ***Kleene closure*** of A — or the ***star*** of A.

- ***Example:*** Suppose that $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $A = \{\mathtt{a}\}$, and $B = \{\mathtt{bb}, \mathtt{c}\}$. Then

$$\begin{aligned} A^\star &= \{\lambda, \mathtt{a}, \mathtt{aa}, \mathtt{aaa}, \mathtt{aaaa}, \dots\} \\ &= \{\omega \in \Sigma^\star \mid \omega \text{ only includes } \mathtt{a}\text{'s}\}. \end{aligned}$$

The Kleene Star of a Language

On the other hand, $B^\star$ includes the following strings (along with lots more):

- λ (obtained by setting $k = 0$)
- bb (obtained by setting $k = 1$ and $x_1 = \texttt{bb}$)
- c (obtained by setting $k = 1$ and $x_1 = \texttt{c}$)
- bbbb (obtained by setting $k = 2$ and $x_1 = x_2 = \texttt{bb}$)
- bbc (obtained by setting $k = 2$, $x_1 = \texttt{bb}$, and $x_2 = \texttt{c}$)
- cbb (obtained by setting $k = 2$, $x_1 = \texttt{c}$, and $x_2 = \texttt{bb}$)
- cc (obtained by setting $k = 2$ and $x_1 = x_2 = \texttt{c}$)

Regular Operations

- Each of ***union***, ***concatenation*** and ***Kleene star*** can be thought of as ***operations*** on the set of languages over an alphabet Σ.

  Definition: For any alphabet Σ, the set of operations

  - union,
  - concatenation, and
  - Kleene star

  are the ***regular operations over*** $\boldsymbol{\Sigma}$.

Closure Properties

Theorem 1. Let Σ be an alphabet, and let $A, B \subseteq \Sigma^{\star}$.

(a) If A and B are regular languages then $A \cup B$ is a regular language, as well.

(b) If A and B are regular languages, then $A \circ B$ is a regular language, as well.

(c) If A is a regular language then $A^{\star}$ is a regular language as well.

The proof of this makes use of a technical result that is given on the next slide.

Closure Properties

Lemma 2. Let Σ be an alphabet, and let $L \subseteq \Sigma^\star$. Then L is a regular language if and only if L is the language $L(M)$ of some nondeterministic finite automaton $M = (Q, \Sigma, \delta, q_0, F)$ which satisfies the following properties.

(a) There are no transitions into q_0, at all. That is, $q_0 \notin \delta(q, \sigma)$ for any state $q \in Q$ or any symbol $\sigma \in \Sigma_\lambda$, so that the only string $\omega \in \Sigma^\star$ such that $q_0 \in \delta^\star(q_0, \omega)$ is the empty string, $\omega = \lambda$.

(b) M has exactly one accepting state, q_F, and there are no transitions out of this state. That is, $F = \{q_F\}$ and $\delta(q_F, \sigma) = \emptyset$ for every symbol $\sigma \in \Sigma_\lambda$.

Closure Properties

To prove part (a) of Theorem 1, let $A, B \subseteq \Sigma^\star$ such that A and B are both regular languages.

- Then there exist nondeterministic finite automata

$$M_1 = (Q_1, \Sigma, \delta_1, q_{1,0}, F_1) \text{ and } M_2 = (Q_2, \Sigma, \delta_2, q_{2,0}, F_2)$$

  that each satisfy the additional properties in Lemma 1 (so that $F_1 = \{q_{1,F}\}$ for a state $q_{1,F} \in Q_1$ and $F_2 = \{q_{2,F}\}$ for a state $q_{2,F} \in Q_2$) such that $L(M_1) = A$ and $L(M_2) = B$.
- Renaming states, as needed, we may assume that $q_0 \notin Q_1 \cup Q_2$ and that $Q_1 \cap Q_2 = \emptyset$.

Closure Properties

Consider an NFA

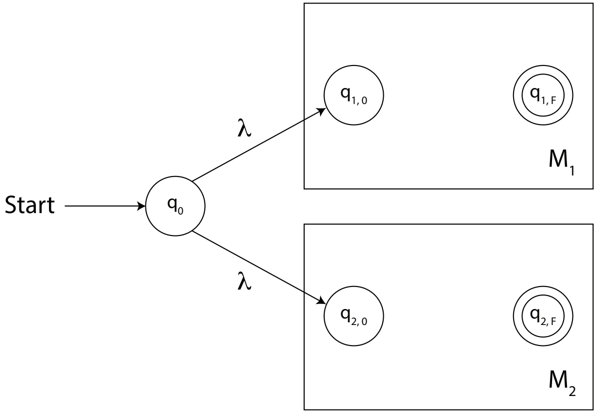
$$M = (Q, \Sigma, \delta, q_0, F)$$

where $Q = \{q_0\} \cup Q_1 \cup Q_2$, $F = F_1 \cup F_2$, and where $\delta : Q \times \Sigma \to Q$ such that, for $q \in Q$ and $\sigma = \Sigma_\lambda$,

$$\delta(q, \sigma) = \begin{cases} \{q_{1,0}, q_{2,0}\} & \text{if } q = q_0 \text{ and } \sigma = \lambda, \\ \emptyset & \text{if } q = q_0 \text{ and } \sigma \in \Sigma, \\ \delta_1(q, \sigma) & \text{if } q \in Q_1, \\ \delta_2(q, \sigma) & \text{if } q \in Q_2. \end{cases}$$

Then M is as shown on the next slide.

Closure Properties



Closure Properties

- It is easily shown, by induction on the length of ω, that if $\omega \in \Sigma^\star$ then

$$\delta^\star(q_0, \omega) = \begin{cases} \delta_1^\star(q_{1,0}, \omega) \cup \delta_2^\star(q_{2,0}, \omega) \cup \{q_0\} & \text{if } \omega = \lambda, \\ \delta_1^\star(q_{1,0}, \omega) \cup \delta_2^\star(q_{2,0}, \omega) & \text{if } \omega \neq \lambda. \end{cases}$$

- Since $F = F_1 \cup F_2$, this can be used to show that $L(M) = L(M_1) \cup L(M_2) = A \cup B$, as required to show that $A \cup B$ is also a regular language.

Closure Properties

- Similar constructions can be used to show that if $A, B \subseteq \Sigma^\star$ are regular languages then $A \circ B$ is a regular language, and that if $A \subseteq \Sigma^\star$ is a regular language then so is $A^\star$.
- A supplemental document includes the rest of the proofs of Lemma 2 and Theorem 1.

Closure Properties

Definition: A ***closure property*** for a set S of languages over an alphabet Σ, is a property stating — for an *operation* on languages over Σ — that if the operation is applied to languages that all belong to the set S, then the result is a language that belongs to the set S, as well.

- Parts (a), (b), and (c) of Theorem #1 are examples of closure properties for the set of regular languages over an alphabet Σ.
- This gives us ***another way to prove that a language is regular:*** Show that it is the union, concatenation or star of other regular language(s).

Regular Expressions and Their Languages

- Let Σ be an alphabet that *does not* include any of the symbols

  $$\lambda, \emptyset, \Sigma, (,), \cup, \circ, ^\star$$

  and let

  $$\Sigma_{\text{regexp}} = \Sigma \cup \{\lambda, \emptyset, \text{“}\Sigma\text{”}, (,), \cup, \circ, ^\star\}$$

  so that Σ_{regexp} incudes a copy of the *symbol*, “Σ”, that we are also using as the name of the language we are starting with.
- A ***regular expression over the alphabet Σ*** is a kind of string of symbols, in $\Sigma_{\text{regexp}}^\star$, as defined by the following list of seven rules.
- The ***language of a regular expression, over the alphabet Σ***, is a subset of $\Sigma^\star$ that is defined as follows.

Regular Expressions and Their Languages

1. For every ***symbol*** $\sigma \in \Sigma$, the ***string*** σ, with length one in $\Sigma^{\star}_{\text{regexp}}$ is a regular expression over Σ.

   The ***language***, $L(\sigma)$, of the regular expression σ, is the ***set*** $\{\sigma\}$.

2. The ***string*** λ, with length one in $\Sigma^{\star}_{\text{regexp}}$, is a regular expression over Σ.

   The ***language***, $L(\lambda)$, of the regular expression λ, is the ***set*** $\{\lambda\}$.

Regular Expressions and Their Languages

3. The ***string*** $\emptyset$, with length one in $\Sigma^\star_{\text{regexp}}$ is a regular expression over Σ.

   The ***language***, $L(\emptyset)$, of the regular expression $\emptyset$, is the ***set*** $\emptyset$.

4. The ***string*** Σ, with length one in $\Sigma^\star_{\text{regexp}}$, is a regular expression over (the alphabet) Σ.

   The ***language***, $L(\Sigma)$, of the regular expression Σ, is the ***finite set*** Σ.

Regular Expressions and Their Languages

5. If $R_1 \in \Sigma^{\star}_{\text{regexp}}$ and $R_2 \in \Sigma^{\star}_{\text{regexp}}$ are ***regular expressions over*** $\boldsymbol{\Sigma}$ then the ***string***

$$(R_1 \cup R_2) \tag{1}$$

(of symbols in Σ_{regexp}) is a regular expression over Σ.

The ***language*** $L(R)$, of the regular expression R at line (1), is the ***set***

$$L(R_1) \cup L(R_2).$$

Regular Expressions and Their Languages

6. If $R_1 \in \Sigma^{\star}_{\text{regexp}}$ and $R_2 \in \Sigma^{\star}_{\text{regexp}}$ are ***regular expressions over Σ*** then the ***string***

$$(R_1 \circ R_2) \tag{2}$$

(of symbols in Σ_{regexp}) is a regular expression over Σ.

The ***language*** $L(R)$, of the regular expression R at line (2), is the ***set***

$$L(R_1) \circ L(R_2).$$

Regular Expressions and Their Languages

7. If $R_1 \in \Sigma^{\star}_{\text{regexp}}$ is a ***regular expression over* Σ** then the ***string***

$$(\ R_1\ )^{\star} \tag{3}$$

(of symbols in Σ_{regexp}) is a regular expression over Σ.

The ***language*** $L(R)$, of the regular expression R at line (3), is the ***set***

$$L(R_1)^{\star}$$

Regular Expressions and Their Languages

Theorem 2: Let Σ be an alphabet that does not include any of the symbols "λ" "$\emptyset$", "Σ", "(", ")", "$\cup$", "$\circ$", or "$\star$" and let $L \subseteq \Sigma^\star$. Then L is a ***regular language*** if and only if there exists a regular expression R, over the alphabet Σ, such that $L = L(R)$.

- There is a ***constructive*** proof of this claim. That is, the proof describes processes to convert between deterministic finite automata with alphabet Σ and regular expressions over Σ (in both directions, preserving the language being described).
- A supplemental document for this lecture gives more information about this.

Regular Expressions and Their Languages

Why Study, and Work With, Regular Expression?

- More effective use of text editors, word processors, database systems, web scrapers...
- Introduction to issues and methods in system software development
- Practice using “inductive definitions” to solve problems

A series of supplemental documents — which are not required reading for this course — gives more information about all this.