CSC 240

REGULAR EXPRESSIONS

- A language is called a regular language if some finite automaton recognizes it.
- Every nondeterministic finite automaton has an equivalent deterministic finite automaton.
- A language is regular if and only if some nondeterministic finite automaton recognizes it.

OPERATIONS ON REGULAR LANGUAGES (THE REGULAR OPERATIONS)

Let A and B be regular languages:

Union: $A \cup B = \{ x \mid x \in A \vee x \in B \}$

Concatenation: $A \circ B = \{ xy \mid x \in A \wedge y \in B \}$

Star: $A^* = \{ x_1 x_2 \ldots x_k \mid k \geq 0 \wedge \text{each } x_i \in A \}$

Usually written
AB

Examples:

$\Sigma = \{a \ldots z\}$

$A = \{ \text{happy}, \text{sad} \}$

$B = \{ \text{cat}, \text{dog} \}$

$A \cup B = \{ \text{happy}, \text{sad}, \text{cat}, \text{dog} \}$

$A \circ B = \{ \text{happycat}, \text{sadcat}, \text{happydog}, \text{saddog} \}$

$A^* \{ \varepsilon$, happy, sad, happyhappy, sadsad, happysad,
sadhappy, sadsadhappy, sadhappysad, … $\}$

CONCATENATION

Σ = {a…z, A…Z}

Article = { A, The }

Noun = { Cat, Dog, Moose, Cow }

Verb = { Eats, Chases, Kicks, Loves }

ArticleNounVerbArticleNoun = {
ACatEatsTheDog, TheMooseChasesACow,
ACowLovesACat, … }

If Article, Noun, and Verb are regular languages, is ArticleNountVerbArticleNoun a regular language?

CONCATENATION SYNTAX

$\Sigma = \{a\ldots z\}$

$A = \{ a, b \}$

$AA = \{ aa, ab, ba, bb \}$

$AAA = \{ aaa, aab, aba, baa, bba, bab, abb, bbb \}$

$A^0 = \{\varepsilon\}$

$A^1 = A$

$A^2 = AA$

$A^* = \{ x_1x_2\ldots x_k \mid k \geq 0 \wedge \text{each } x_i \in A\}$

$A^+ = \{ x_1x_2\ldots x_k \mid k \geq 1 \wedge \text{each } x_i \in A\}$

Sometimes called the "Kleene Star"

Sometimes called the "Kleene Plus"

CLOSURE

$\mathbb{N} = \{1, 2, 3, \ldots\}$

$\mathbb{N}$ is *closed* under multiplication:

$$
\begin{aligned}
&\forall x.\ (x \in \mathbb{N} \rightarrow \\
&\qquad \forall y.\ (y \in \mathbb{N} \rightarrow x \cdot y \in \mathbb{N}) \\
&\quad)
\end{aligned}
$$

For every x and y in the set of natural numbers, the product of x and y is a natural number.

Regular Languages are closed under the Union operation.

Regular Languages are closed under the Concatenation operation.

Regular Languages are closed under the Star operation.

- If we can create a DFA that recognizes L.
- If we can create an NFA that recognizes L.
- If L can be formed from other regular languages using regular operations.

Regular Expressions are expressions built from regular operations and are used to describe a language.

R is a regular expression if R is:

a for some a in Σ

ε

$\emptyset$

$(R_1 \cup R_2)$ Where R_1 and R_2 are regular expressions

$(R_1 \circ R_2)$ Where R_1 and R_2 are regular expressions

$(R_1{}^*)$ Where R_1 is a regular expression

$\Sigma = \{0, 1\}$

$0^*10^* = \{w \mid w \text{ contains a single } 1\}$

$\Sigma^*1\Sigma^* = \{w \mid w \text{ contains at least a single } 1\}$

Means "star operator on the alphabet"

$\Sigma^*001\Sigma^* = \{w \mid w \text{ contains the substring } 001\}$

$1^*(01^+)^* = \{w \mid \text{every } 0 \text{ in } w \text{ is followed by at least a single } 1\}$

$(\Sigma\Sigma)^* = \{w \mid w \text{ is a string of even length}\}$

$01 \cup 10 = \{01, 10\}$

$0\Sigma^*0 \cup 1\Sigma^*1 \cup 0 \cup 1 = \{w \mid w \text{ starts and ends with the same symbol}\}$

$(0 \cup \varepsilon)1^* = 01^* \cup 1^*$

$R \cup \varnothing = R$

Adding the empty language to another language doesn't change it.

$R \circ \varepsilon = R$

Concatenating the empty string to another string doesn't change it.

However:

$R \cup \varepsilon \neq R$

If $R = 0$, then $L(R) = \{0\}$, but $L(R \cup \varepsilon) = \{0, \varepsilon\}$

$R \circ \varnothing \neq R$

If $R = 0$, then $L(R) = \{0\}$, but $L(R \circ \varnothing) = \varnothing$

REGULAR EXPRESSION IN PROGRAMMING

$D = \{0 \ldots 9\}$

$\Sigma = \{D, +, -, .\}$

$(+ \cup - \cup \varepsilon)\ (D^{+} \cup D^{+}.D^{*} \cup D^{*}.D^{*})$

Positive or negative numbers with or without decimals.

$W = \{a \ldots z, A \ldots Z\}$

$\Sigma = \{W, ., @\}$

$W^{+}(.W^{+})^{*}@W^{+}(.W^{+})^{+}$

Valid email addresses