Regular expression

Lin Chen

Email: Lin.Chen@ttu.edu

Grader: zulfi.khan@ttu.edu

TEXAS TECH UNIVERSITY

Strings

- An alphabet Σ is a finite set of symbols
  - $\Sigma_1 = \{0,1\}$
  - $\Sigma_2 = \{a, b, c, \cdots, z\}$
- A string is a finite sequence of symbols from an alphabet
  - apple, banana are both strings over $\Sigma_2 = \{a, b, c, \cdots, z\}$
  - 100110 is a string over $\Sigma_1 = \{0,1\}$

Formal language

- The set of all strings over alphabet Σ is denoted as Σ^*
  - $e \in \Sigma^*$
- Any subset of Σ^* is called a language
  - English is a subset of $\{a, b, \cdots, z\}^*$

Language operations

- Concatenation
  - $L_1, L_2 \subseteq \Sigma^*$, $L_1 \circ L_2 = L_1 L_2 = \{w \in \Sigma^* : w = xy, x \in L_1, y \in L_2\}$
  - Example: $\{a, ab\}\{b, bb\} = \{abb, ab, abbb\}$

Language operations

- Complement
  - For $L \in \Sigma^*, \overline{L} = \Sigma^* - L$

Language operations

- Complement

  - For $L \in \Sigma^*, \bar{L} = \Sigma^* - L$

- Kleene star

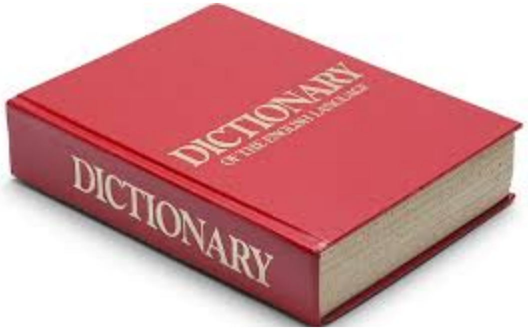
  - L^*: the set of all strings obtained by concatenating zero or more strings of L

  - $L^* = \{w \in \Sigma^* : w = w_1 w_2 \cdots w_k \text{ for some } k \geq 0, w_i \in L, 1 \leq i \leq k\}$

  - $\Sigma^* = \{w \in \Sigma : w = w_1 w_2 \cdots w_k \text{ for some } k \geq 0, w_i \in \Sigma, 1 \leq i \leq k\}$ = all the strings over alphabet Σ

Regular expression

- We want to express interesting languages
  - In a succinct way, if possible
  - can express languages of infinite size



Regular expression

- Regular expressions are an algebraic way to describe languages.

Say that R is a ***regular expression*** if R is

1. a for some a in the alphabet Σ,
2. ε,
3. $\emptyset$,
4. $(R_1 \cup R_2)$, where R_1 and R_2 are regular expressions,
5. $(R_1 \circ R_2)$, where R_1 and R_2 are regular expressions, or
6. (R_1^*), where R_1 is a regular expression.

Regular expression

- Regular expressions are defined recursively
  - Base case – simple regular expressions
  - Recursive case – how to build more complex regular expressions from simple regular expressions

Regular expression

- The regular expressions of Σ^* are all strings over $\Sigma \cup \{(,), \emptyset, +, \star\}$ that can be obtained through the following operations:
  - $\emptyset$ and every member of Σ is a regular expression
  - If α and β are regular expressions, then so is $(\alpha\beta)$
  - if α and β are regular expressions, then so is $(\alpha \cup \beta)$
  - if α is a regular expression, then so is α^*
  - Nothing else is a regular expression

Regular expression

- Examples
  - $(((a \cup b)(b^*))a)$

    $\{aa, ba, aba, bba, abba, bbba, abbba, bbbba, \ldots\}$

Regular expression

- Examples
  - $(((a \cup b)(b^*))a)$

    $\{aa, ba, aba, bba, abba, bbba, abbba, bbbba, \ldots\}$
  - $((a((a \cup b)^*))a)$

    $\{aa, aaa, aba, aaaa, aaba, abaa, abba, \ldots\}$

Regular expression

- Examples
  - $(((a \cup b)(b^*))a)$

    $\{aa, ba, aba, bba, abba, bbba, abbba, bbbba, \ldots\}$
  - $((a((a \cup b)^*))a)$

    $\{aa, aaa, aba, aaaa, aaba, abaa, abba, \ldots\}$
  - $((a^*)(b^*))$

    $\{e, a, b, aa, ab, bb, aaa, aab, abb, bbb, \ldots\}$

Regular expression

- Examples
  - $(((a \cup b)(b^*))a)$
    $\{aa, ba, aba, bba, abba, bbba, abbba, bbbba, \ldots\}$
  - $((a((a \cup b)^*))a)$
    $\{aa, aaa, aba, aaaa, aaba, abaa, abba, \ldots\}$
  - $((a^*)(b^*))$
    $\{e, a, b, aa, ab, bb, aaa, aab, abb, bbb, \ldots\}$
  - $((ab)^*)$
    $\{e, ab, abab, ababab, abababab, \ldots\}$

Regular expression

- ( ) can sometimes be dropped
  - $(((ab)b)a) = abba$
- Sometimes () cannot be dropped
  - $a \cup (b(b^*))a, (a \cup b)(b^*)a, \left(a \cup (bb)\right)^* a$ are different

Regular expression

- More examples

1. $0^*10^* = \{w \mid w \text{ contains a single } 1\}$.
2. $\Sigma^*1\Sigma^* = \{w \mid w \text{ has at least one } 1\}$.
3. $\Sigma^*001\Sigma^* = \{w \mid w \text{ contains the string } 001 \text{ as a substring}\}$.
4. $1^*(01^+)^* = \{w \mid \text{every } 0 \text{ in } w \text{ is followed by at least one } 1\}$.
5. $(\Sigma\Sigma)^* = \{w \mid w \text{ is a string of even length}\}$.[5]
6. $(\Sigma\Sigma\Sigma)^* = \{w \mid \text{the length of } w \text{ is a multiple of three}\}$.
7. $01 \cup 10 = \{01, 10\}$.
8. $0\Sigma^*0 \cup 1\Sigma^*1 \cup 0 \cup 1 = \{w \mid w \text{ starts and ends with the same symbol}\}$.

Regular expression

- All strings over $\{a, b\}$ that start with an a
  - $a(a \cup b)^*$

Regular expression

- All strings over $\{a, b\}$ that start with an a
  - $a(a \cup b)^*$
- All strings over $\{a, b\}$ that are even in length
  - $\left((a \cup b)(a \cup b)\right)^*$

Regular expression

- All strings over $\{a, b\}$ that start with an a
  - $a(a \cup b)^*$
- All strings over $\{a, b\}$ that are even in length
  - $\big((a \cup b)(a \cup b)\big)^*$
- All strings over {0,1} that have an even number of 1’ s.
  - $0^* \, (10^*10^*)^*$

Regular expression

- All strings over $\{a, b\}$ that start with an a
  - $a(a \cup b)^*$
- All strings over $\{a, b\}$ that are even in length
  - $\left((a \cup b)(a \cup b)\right)^*$
- All strings over {0,1} that have an even number of 1’ s.
  - $0^* \, (10^*10^*)^*$
- All strings over a, b that start and end with the same letter
  - $a(a \cup b)^* a \cup \ b(a \cup b)^* b \cup \ a \cup \ b$

Regular language

- A language is regular if and only if some regular expression describes it.

Recall a language is defined as regular if a DFA (NFA) recognizes it.