Regular Expression & Regular Languages

Regular Language

- A language L is known as **regular if and only if** it is recognized by a finite accepter (FA).
  - **Language L is regular if and only if** it is recognized by a **DFA**. (??)
  - **Language L is regular if and only if** it is recognized by an **NFA**. (??)

- A language L is known as **regular if and only if** it is described by a **regular expression (RE)**.
- A language L is recognized by a FA **if and only if** L is described by a **regular expression**.

- NFA recognize exactly the regular languages.
- **Regular expressions** describe exactly the regular languages.

How to show that a given language is regular ?

Regular Expression

- **A regular expression consists of strings of symbols from some alphabet Σ, parentheses (), and the operators +, . and *.**

- Let Σ be a given alphabet. Then,
  - ϕ, λ *and* $a \in \Sigma$ are all regular expressions. These are known as primitive regular expressions.
  - **Recursive Definition:**
    - If r_1 and r_2 are regular expressions (REs), then the following expressions are also regular:
      - $r_1 + r_2$ ***OR*** $r_1 \mid r_2$ ➔ (r_1 or r_2)
      - $r_1.r_2$ ***OR*** $r_1 r_2$ ➔ (r_1 followed by r_2)
      - r_1^* ➔ (r_1 repeated zero or more times)
      - (r_1)

- A strings of symbols is a regular expression **if and only if** it can be derived from **primitive regular expressions** by **finite** applications of **recursive definition**.

Precedence of Operators

Rules for Specifying Regular Expressions:

	PRECEDENCE
()	HIGHEST
*	.
	.
.	.
	.
|	.
	LOWEST

Rules for REs

For r, s and t be RE over Σ

- ✓ $r + \varnothing = \varnothing + r = r$
- ✓ $r \cdot \varnothing = \varnothing \cdot r = \varnothing$
- ✓ $\varnothing^* = \Lambda$

- ✓ $r + \Lambda = \Lambda + r = r$
- ✓ $r \cdot \Lambda = \Lambda \cdot r = r$
- ✓ $\Lambda^* = \Lambda$
- ✓ $(r + \Lambda)^+ = r^*$

- ✓ $r + s = s + r$
- ✓ $r \cdot (s + t) = r \cdot s + r \cdot t$
- ✓ $r \cdot (s.t) = (r \cdot s) \cdot t$

- ✓ $r^+ = r\, r^*$
- ✓ $r^* = r^*(r + \Lambda) = r^*\, r^* = (r^*)^*$
- ✓ $(r^* s^*)^* = (r + s)^*$

Valid Regular Expressions: Example

- Let $\Sigma = \{a, b, c\}$

- ϕ, λ, *a*, b, c
- a*, b*
- a.b, b.a, a+b, (a.b)*, (b.a)*, (a+b)*
  - a + b **is equivalent to** b+a
  - a.b **is not equivalent** to b.a
- (a + b.c)*
- (c+ϕ)

Why ?

Invalid Regular Expressions: Example

- Let Σ ={a, b, c}

- *a, *b*, +a*, .b*
- +a.b, *b.a, .*a+b, (++a.b)*, (..*b.a***)*, (++a++b**)*
- (+a + b.c)*
- (c+ϕ*+*)

Why ?

Some Notations

- Parentheses in regular expressions can be omitted when the order of evaluation is clear.
  - **$((0+1)^*) = (0+1)^* \neq 0+1^*$**
  - **$((0^*)+(1^*)) = 0^* + 1^*$**
- For concatenation, $\cdot$ can be omitted.
- $r \cdot r \cdot r \ldots r$ is denoted by r^n.

n times

Simple Examples over $\Sigma = \{0,1\}$

- **$\{\alpha \in \Sigma^* \mid \alpha$ does not contain 1's$\}$**
  - **0^***
- **$\{\alpha \in \Sigma^* \mid \alpha$ contains 1's only$\}$**
  - **$1 \cdot (1^*)$ (which can be denoted by (1^+))**
- **$\{\alpha \in \Sigma^* \mid \alpha$ contains only 0's or only 1's$\}$**
  - **$(00^*)+(11^*)$**
- **Σ^***
  - **$(0+1)^*$**
- **Note: $0^* + 1^* \neq (0+1)^*$**

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

- **Strings of even length, L={00,01,10,11} ***
  - **(00+01+10+11) * or**
  - **((0+1)(0+1))***
- **Strings of length 6, L={$\alpha \in \Sigma^*$| the length of α is 6}**
  - 000000+….+111111
  - **(0+1)(0+1) (0+1)(0+1) (0+1)(0+1) =$(0+1)^6$**
- **Strings of length 6 or less, L={$\alpha \in \Sigma^*$| the length of α is less than or equal to 6}**
  - λ +0+1+00+01+10+11….+111111
  - **$(0+1+ \lambda)^6$**

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

- **$\{\alpha \in \Sigma^* \mid \alpha$ is a binary number divisible by 4$\}$**
  - **(0+1)*00**

- **$\{\alpha \in \Sigma^* \mid \alpha$ does not contain 11$\}$**
  - **(0+10)* (1+ λ)**

- **$\{\alpha \in \Sigma^* \mid \alpha$ contains odd number of 1's$\}$**
  - **0*(10*10*)*10***

- **$\{\alpha \in \Sigma^* \mid$ any two 0's in α are separated by three 1's$\}$**
  - **1*(0111)*01* + 1***

Regular Expressions: Example

- All strings of 1s and 0s
  $(0 \mid 1)^*$
- All strings of 1s and 0s beginning with a 1
  $1\ (0 \mid 1)^*$
- All strings containing two or more **0**s
  $(1|0)^*\mathbf{0}(1|0)^*\mathbf{0}(1|0)^*$
- All strings containing an even number of **0**s
  $(1^*\mathbf{0}1^*\mathbf{0}1^*)^* \mid 1^*$

Regular Expressions : Example

- All strings containing an **even** number of **0**s and **even** number of **1**s

  Assume that (0 0 | 1 1) is X

  X* | (X* (0 1 | 1 0) X* (0 1 | 1 0) X*)*

  OR

  $(00 \mid 11)^*((01 \mid 10)(00 \mid 11)^*(01 \mid 10)(00 \mid 11)^*)^*$

- All strings of alternating **0**s and **1**s

  $(\lambda \mid 1)(01)^*(\lambda \mid 0)$

- Strings over the alphabet {a, b} in which substrings **ab** and **ba** occur an unequal number of times
  - $(a^+b^+)^+ \mid (b^+a^+)^+$

Regular Expressions : Example

- Strings over the alphabet {0, 1} with **no consecutive 0's**
  - $(1 \mid 01)^* (0 \mid \varepsilon)$
  - $1^*(01^+)^* (0 \mid \varepsilon)$
  - $1^*(011^*)^* (0 \mid \varepsilon)$

- Strings over the alphabet {a, b} with **exactly three b's**
  - $a^*ba^*ba^*ba^*$

- Strings over the alphabet {a, b, c} containing (**at least once**) **bc**
  - $(a|b|c)^*bc(a|b|c)^*$

Regular Expressions : Example

- $(1 \mid 10)^*$
    - all strings starting with “1” and containing no “00”
- $(0 \mid 1)^*011$
    - all strings ending with “011”
- 0^*1^*
    - all strings with no “0” after “1”
- 00^*11^*
    - all strings with at least one “0” and one “1”, and no “0” after “1”

Regular Expressions : Example

- **What languages do the following RE represent?**
  - $((0 \mid 1)(0 \mid 1))^* \mid ((0 \mid 1)(0 \mid 1)(0 \mid 1))^*$

Regular Languages

- **Each RE has an equivalent regular language (RL).**

- A language L is regular if there is a regular expression r such that L = L(r).

- The language L(r) denoted by any regular expression r is defined by the following rules.
  - **Φ is a regular expression.** **$L(\Phi) = \{\} = \Phi$**
  - **λ is a regular expression.** **$L(\lambda) = \{\lambda\}$**
  - **$a \in \Sigma$ are all regular expressions.** **$L(a) = \{a\}$**

Regular Languages: Cont..

- If r_1 and r_2 are regular expressions (REs).
  - $r_1 + r_2$ is R.E., then $L(r_1 + r_2) = L(r_1) \cup L(r_2)$

    $L(r_1) \cup L(r_2) = \{w \mid w \in L(r_1) \text{ or } w \in L(r_2)\}$
  - $r_1.r_2$ is R.E., then $L(r_1 \cdot r_2) = L(r_1)\, L(r_2)$

    $L(r_1).L(r_2) = \{w_1.w_2 : w_1 \in L(r_1) \text{ and } w_2 \in L(r_2)\}$
  - r_1^* is R.E., then $L(r_1^*) = (L(r_1))^*$

    $(L(r_1))^* = L(r_1)^0 \cup L(r_1)^1 \cup L(r_1)^2 \cup L(r_1)^3 \cup \ldots$
  - (r_1) is R.E., then $L((r_1)) = L(r_1)$

Regular Expression to Regular Language

Regular Expression: $(a+b)\cdot a^*$

$$
\begin{aligned}
L((a+b)\cdot a^*) &= L((a+b))\,L(a^*) \\
&= L(a+b)\,L(a^*) \\
&= (L(a)\cup L(b))\,(L(a))^* \\
&= (\{a\}\cup\{b\})\,(\{a\})^* \\
&= \{a,b\}\,\{\lambda,a,aa,aaa,\ldots\} \\
&= \{a,aa,aaa,\ldots,b,ba,baa,\ldots\}
\end{aligned}
$$

RE to RL

$$r = (a+b)^*(a+bb)$$

$$L(r) = \{a, bb, aa, abb, ba, bbb, \ldots\}$$

$$r = (aa)^*(bb)^*b$$

$$L(r) = \{a^{2n}b^{2m}b : \quad n, m \geq 0\}$$

$$r = (0+1)^*00(0+1)^*$$

$L(r)$ = { all strings containing substring 00 }

RE & RL: Example

- λ^* is RE, then the language

  $$L(\lambda^*) = \{\lambda\}^* = \{\lambda\}$$

- ϕ^* is RE, then the language

  $$L(\phi^*) = \{\phi\}^* = \{\ \}$$

- 0^* is RE, then the language

  $$L(0^*) = \{0\}^* = \{\lambda, 0, 00, 000, 0000, \ldots\}$$

- $(0+1).(00+11)$ is RE, then the language

  $$L(\,(0+1).(00+11)\,) = \{0, 1\}\{00, 11\} = \{000, 011, 100, 111\}$$

- $(10+01)^*$ is RE, then the language

  $$L((10+01)^*) = \{10, 01\}^* = \{\lambda, 10, 1010, 101010, \ldots, 01, 0101, 010101, \ldots, 1001, 100101, 10010101, \ldots, 0110, 011010, 01101010, \ldots\}$$

RE & RL: Example

- Let *L* be a language over {a, b}, each string in *L* contains the substring **bb**
  - $L = \{a, b\}^*\{bb\}\{a, b\}^*$

- L is regular language (RL). Why?
  - {a} and {b} are RLs
  - {a, b} is RL
  - $\{a, b\}^*$ is RL
  - {b}{b} = {bb} is also RL
  - Then $L = \{a, b\}^*\{bb\}\{a, b\}^*$ is RL

RE & RL: Example

- Let L be a language over {a, b}, each string in L
  - begins and ends with an **a** **AND** contains at least one **b**
  - $L = \{a\}\{a, b\}^*\{b\}\{a, b\}^*\{a\}$

- L is regular language (RL). Why?
  - {a} and {b} are RLs
  - {a, b} is RL
  - $\{a, b\}^*$ is RL
  - Then $L = \{a\}\{a, b\}^*\{b\}\{a, b\}^*\{a\}$ is RL

RL - Example

- The RE (b + ab*a)*ab* represents the strings over {a, b} with an odd number of a’s
- Note: this is a set equality; to prove it you have to show the following:
  - strings with an odd number of a’s are in this language; and
  - any string in this language has an odd number of a’s.

RE & RL: Example

- Let $\sum$= {a, b}
  - RE $a|b$ ➔ L = {a, b}
  - RE $(a|b)(a|b)$ ➔ L = {aa, ab, ba, bb}
  - RE $aa|ab|ba|bb$ same as above
  - RE a^* ➔ L = {λ, a , aa, aaa, …}
  - RE $(a|b)^*$ ➔ L = set of all strings of a's and b's including λ
  - RE $(a^*b^*)^*$ ➔ same as above
  - RE $a|a^*b$ ➔ L = {a,b,ab,aab,aaab, …}

RE & RL

- 01^*
  - {0, 01, 011, 0111, …..}
- $(01^*)(01)$
  - {001, 0101, 01101, 011101, …..}
- $(0 \mid 1)^*$
  - {0, 1, 00, 01, 10, 11, …..}
  - i.e., all strings of 0 and 1
- $(0 \mid 1)^*\ 00\ (0 \mid 1)^*$
  - {00, 1001, …..}
  - i.e., all 0 and 1 strings containing a "00"

EQUIVALENT REs

- Two regular expressions **r and s** are equivalent (r=s), **if and only if** r and s represent/generate the same language.

- Example-:
  - r = a|b, s = b|a → r = s   Why?
  - Since L(r) = L(s) = {a, b}

- RE = (a)|((b)*(c)) is equivalent to a + b*c

EQUIVALENT REs

- Examples,
  - $(a^*b^*)^* = (a+b)^*$
  - $(a+b)^*ab(a+b)^*+b^*a^* = (a+b)^*$
- First equality rather clear.
- For the second equality, note that $(a+b)^*$ denotes strings over a and b, that a string either contains ab or it doesn’t; the first half of the left-hand expression describes the strings that contain the substring *ab* and the second half describes those that don’t; the + says “take the union”.

Regular Expressions: Exercise

- Construct a RE over $\Sigma=\{0,1\}$ such that
  - It does not contain any string with two consecutive “0”s
  - It has no prefix with two or more “0”s than “1” nor two or more “1”s than “0”
  - The set of all strings ending with “00”
  - The set of all strings with 3 consecutive 0’s
  - The set of all strings beginning with “1”, which when interpreted as a binary no., is divisible by 5
  - The set of all strings with a “1” at the 5th position from the right
  - The set of all strings not containing 101 as a sub-string

- Construct a RE for the set $\{a^n b^m: n >= 3, m \text{ is even}\}$.
- Construct a RE for the set $\{a^n b^m: n >= 4, m <= 3\}$.
- Construct a RE for the set $\{w: |w| \bmod 3 = 0\}$.
- Construct a RE for the set $\{w: |w| \bmod 3 = 1\}$