Theory of Automata & Formal Languages

(Theory of Computation)

Compiled By

Prof. M. S. Bhatt

Outline

- Regular Expressions
- Regular Languages

Regular Expressions

- $\varnothing$ is a regular language corresponding to the regular expression $\varnothing$.
- $\{\lambda\}$ is a regular language corresponding to the regular expression λ.

- For any symbol $a \in \Sigma$, $\{a\}$ is a regular language corresponding to the regular expression a.

- If L_1 and L_2 are regular languages corresponding to the regular expression r_1 and r_2, then
  - $L_1 \cup L_2$, $L_1 \cdot L_2$, and L_1^* are regular languages corresponding to $(r_1 + r_2)$, $(r_1 \cdot r_2)$, and (r_1^*).

Example. The following are some regular expressions over alphabet $A = \{a, b\}$, and the corresponding regular languages:

<u>Expression</u>	<u>Language</u>
$a + ab$	$\{a, ab\}$
$(a + b)^*bb$	$\{a, b\}^*\{bb\}$
$ba(a + b)^*ab$	$\{ba\}\{a, b\}^*\{ab\}$
λ	$\{\lambda\}$

Regular Expressions:

To simplify the notations of regular languages, and drawing analogy to arithmetic expressions used in algebra, we could replace the union symbol $\cup$ with the plus sign +, drop the braces "{" and "}" for sets, and use parentheses "(" and ")" for grouping when necessary

$$0^* + 1^* \neq (0+1)^*$$

Examples

Find a string of minimum length in {0,1}* not in the language corresponding to given regular expressions.

1) $1^*(01)^*0^*$

2) $(0^*+1^*)(0^*+1^*)(0^*+1^*)$

3) $0^*(100^*)^*1^*$

Examples

Find a string of minimum length in {0,1}* not in the language corresponding to given regular expressions.

1) 1*(01)*0* → 001

2) (0*+1*) (0*+1*) (0*+1*) → 0101

3) 0*(100*)*1* → 110

More examples

R= 0* + 1*

S= 01*+10*+1*0 + (0*1)*

1)Find a string corresponding to r but not is s

2)Find a string corresponding to s but not is r

3)Find a string corresponding to both r and s

More examples

R= 0* + 1*

S= 01*+10*+1*0 + (0*1)*

1) Find a string corresponding to r but not is s

00

2) Find a string corresponding to s but not is r

01

3) Find a string corresponding to both r and s

0

Simple examples

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

- $\{\alpha \in \Sigma^* | \alpha$ does not contain 1's$\}$
- $\{\alpha \in \Sigma^* | \alpha$ contains 1's only$\}$
- Σ^*
- $\{\alpha \in \Sigma^* | \alpha$ contains only 0's or only 1's$\}$

Simple examples

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

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

More examples

(a)

The language of integer constants as in C:

(b)

The set of strings over {a, b} that contain the substring aa:

(c)

The set of strings over {a, b} that contain exactly two occurrences of symbol a:

More examples

(a)

The language of integer constants as in C:

$$(0+1+2+3+4+5+6+7+8+9)\,(0+1+2+3+4+5+6+7+8+9)^*$$

(b)

The set of strings over {a, b} that contain the substring aa:

$$(a + b)^*aa\,(a + b)^*$$

(c)

The set of strings over {a, b} that contain exactly two occurrences of symbol a:

$$b^*ab^*ab^*$$

More complex examples

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

- $\{\alpha \in \Sigma^* |\ \alpha$ contains odd number of 1's$\}$
  - $0^*(10^*10^*)^*10^*$
- $\{\alpha \in \Sigma^* |$ any two 0's in α are separated by atleast three 1's$\}$
  - $1^*(0111)^*1^*01^* + 1^*$

More complex examples

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

- $\{\alpha \in \Sigma^* | \alpha$ is a binary number divisible by $4\}$
  - $(0+1)^*00$
- $\{\alpha \in \Sigma^* | \alpha$ does not contain $11\}$
  - $0^*(10^+)^* (1+ \lambda)$ or $(0+10)^* (1+ \lambda)$