

Regular Expression and Languages

Notes by :- @jpwebdevelopers

→ Regular Expression is a method to represent a language.

(Regular language की represent करती)

* Regular Language :- The Language accepted by some regular expression are referred to as Regular Language.

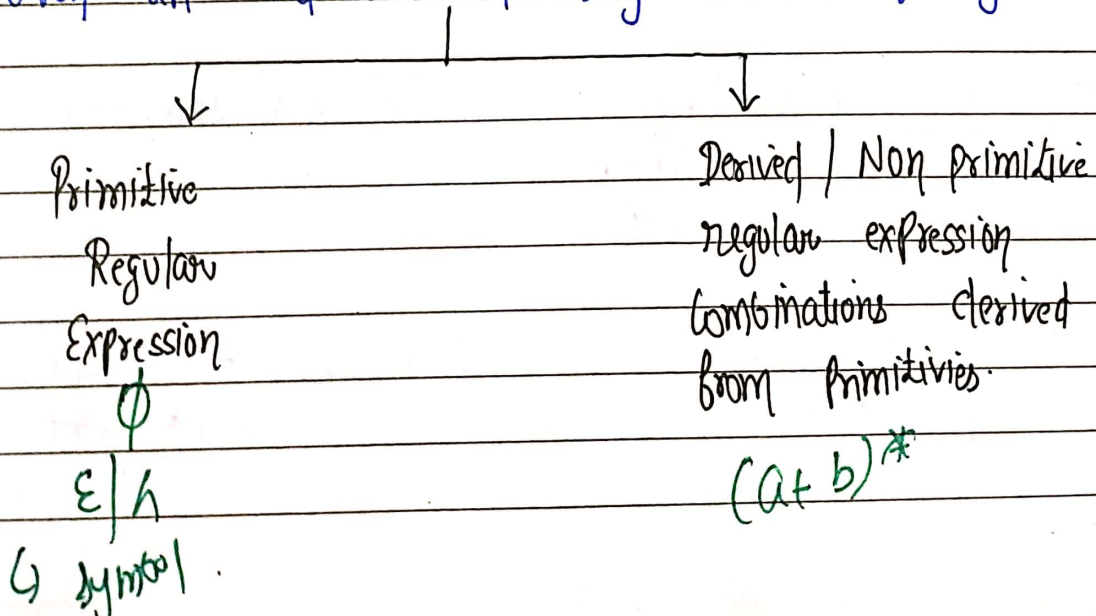
- FSM accepts करती
- TPE-3 Grammar

→ It can also be described as a sequence of pattern that defines a string.

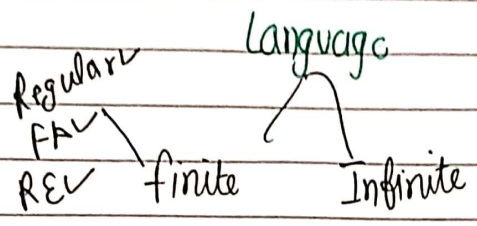
Regular Expression is a valid expression of string with operators like :-

- Kleen's closure a^*
- Positive kleen's closure a^+
- Concatenation $a-b$
- Union $a+b$

over an alphabet of any number of symbol.



Imp:- A Regular Expression is said to be valid, if it is derived from primitive Regular Expression like union, concatenation, Kleen's closure, positive closure etc.
 $(a+b)^* = \Sigma^*$



Regular Expression

Regular Language

- ✓ $\pi = \phi$ $L(\pi) = \{\emptyset\}$ (No string)
- ✓ $\pi = \epsilon$ $L(\pi) = \{\epsilon\}$ (Length 0)
- ✓ $\pi = a$ $L(\pi) = \{a \mid a \in \Sigma\}$ (Length 1)
- ✓ $\pi = a+b$ $L(\pi) = \{a, b \mid a, b \in \Sigma\}$ (" 2)
- ✓ $\pi = (a+b)(a+b)$ $L(\pi) = \{aa, ab, ba, bb\}$ (" 3)
- ✓ $\pi = (a+b)(a+b)(a+b)$ $L(\pi) = \{aaa, aab, aba, \dots\}$ (" finite)
- ✓ $\pi = a \cdot b$ $L(\pi) = \{ab\}$
 (Combination of अक्षरों है
 But $(ab)^*$ ही अक्षर, क्योंकि Power नहीं है)
- ✓ $\pi = (ab+a) \cdot b$ $L(\pi) = \{abb, ab\}$
 ($ab \cdot b$)
 ($a \cdot b$)
- ✓ $\pi = a^*$ $L(\pi) = \{\epsilon, a, aa, \dots\}$
- ✓ $\pi = (a+b)^+$ $L(\pi) = \{a^*, b^*\}$

$\pi = \epsilon \quad L(\pi) = \{\epsilon\}$

$\pi = \phi \quad L(\pi) = \{\} / \phi$

RE

$a \leftarrow a + \epsilon \rightarrow a \uparrow \epsilon$
 $a/\phi \leftarrow a + \phi = a$
 | \uparrow एक ही string में 0
 x length की

Reg. Language

$\pi = (a+b)^*$
 ↳ ab
 ↳ ba
 ↳ ϵ

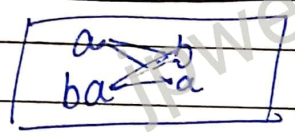
order में

$L(\pi) = \Sigma^* = \{a^*, b^*, \epsilon\}$

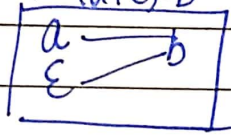
$(ab)^*$
 ↳ $ab, abab, ababab, \dots$

symbol के बीच में

$\pi = (a+ba).(b+a)$
 $L(\pi) = \{ab, aa, bab, baay\}$



$\pi = (a+\epsilon).(b+\phi)$
 $L(\pi) = \{ab, b\}$



Imp

Imp identities

same	$\pi = \Sigma^*$	$\rightarrow \{ \Sigma \}$
	$\pi = \Sigma^+$	$\rightarrow \{ \Sigma \}$
different	$\pi = \phi^*$	$\rightarrow \{ \epsilon \}$
	$\pi = \phi^+$	$= \{ \}$

Imp

$\pi^* . \pi^+ = \pi^+$

$(a+b)^* = (a^* . b^*)^*$

$(\epsilon + \pi^+) . \pi^+ \rightarrow$

$(\pi^*)^* = \pi^*$

$(\pi^+)^* = \pi^*$

NOTE :-

- Multiple regular expression can generate same language but one regular expression can not generate multiple language.

- Two regular expressions are equivalent if languages generated by them are same. For example: $(a+b^*)^*$ and $(a+b)^*$ generate same language. Every string which is generated by $(a+b)^*$ is also generated by $(a+b^*)^*$ and vice-versa.

→ ϵ is a Regular Expression indicates the language containing an empty string.
($L(\epsilon) = \{\epsilon\}$)

→ ϕ is Regular Expression denotes an empty language.
($L(\phi) = \{\}$)

→ x is Regular Exp. where $L = \{x\}$

→ R^* is a Regular Exp. corresponding to the language $L(R^*)$ where $L(R^*) = (L(R))^*$

→ The operation '*' has highest precedence over concatenation which has precedence over (+)
 $(a+(b(c^*))) = abc^*$

Notes by :- jpwebdevelopers