**REGULAR LANGUAGE & FINITE AUTOMATA**

**Regular Expression**

* **Expressions** over alphabet which uses operators **(\*)**, **(.)** & **(+)**.
* These symbols are called ***regular operators***.
* For every **regular language**, there can be a **regular expression**.
* **(\*)** is **kleen closure** operator.
* **(.)** is **concatenation** operator.
* **(+)** is **union** operator.

Operator precedence:-

**\* > . > +**

Types of regular expressions:-

* **Restricted regular expression** [(\*), (.), (+)].
* **Semi-restricted regular expression** [(\*), (.), (+), (n)].
* **Unrestricted regular expression** [(\*), (.), (+), (n), (~)].

Regular expressions identities:-

**r = ф**

**r\* = €**

**Then, r+ = ф**

**r = €**

**r\* = €**

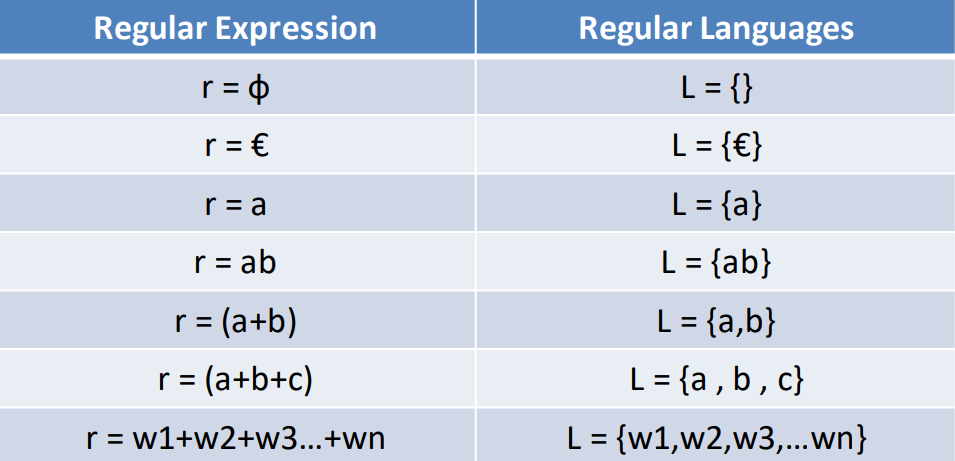
**Then, r+ = €**

**r\* + r + = r\***

**r\* . r = r+**

**(r\*)\* = r\***

Regular expressions for some regular languages:-



Actual Meaning

a\* = Zero or more occurrence of 'a'

a+ = One or more occurrence of 'a'

Example 1:-

Regular expression over Σ for string of length 3:

R.E. = (a|b|c) (a|b|c) (a|b|c)

Example 2:-

0 or more occurrence of either 'a' or 'b':

R.E. = (a|b)\* [Includes €]

Example 3:-

1 or more occurrence of either 'a' or 'b' & ending with 0:

R.E. = (a|b)+0 [Excludes €]

Finite Automata