2/8/2023
Wednesday.

Symbol: Any character that belongs to a to z (or) 0-9 (or) any special characters like , , #, id, (,) ...

Alphabet (Σ): It is denoted by 'Σ'. It is a finite set of elements where those elements are nothing but symbols

Eg: $\Sigma_1 = \{a,b\}$, $\Sigma_2 = \{0,1\}$, $\Sigma_3 = \{(,)\}$.

String (W):
- It is a sequence of characters formed by given alphabet.
- Input symbols are alphabet elements
  ↳ set of symbols
- It is denoted by 'W'.

Note: $|W|$ ⇒
- Magnitude of string (or) length of string.
- It returns no. of. characters in the string

Empty String (ϵ): A string whose length is equal to 'zero'.

i.e., $|W| = 0$.

- It is represented by "ϵ" → read as (Epsilon)

Note: Q) W = abaab then $|W| = ?$

A) length of string (W) = 5

$\Rightarrow \boxed{|W| = 5}$

Language (L): It is represented by 'L'. and It is a set of strings formed over given alphabet symbol (Σ).

Eg1: $\Sigma = \{0,1\}$. design a language 'L' which contains strings of length exactly 2. A) $L = \{00, 01, 10, 11\}$.

Eg2: $\Sigma = \{a,b\}$ design a Language which contains strings starting with 'a'.

Sol: $L = \{ a, aa, ab, aaa, aab, aba, abb, aaaa, \dots \}$

NOTE: Language set can be a finite set (or) infinite set.

Eg3: Design a Language which contains even length strings. ($\Sigma = \{a,b\}$)

Sol: $L = \{\epsilon, aa, ab, ba, bb, aaaa, abaa, abba, abbb, bbbb, \dots\}$.

$\epsilon \rightarrow$ empty string $\longrightarrow |w| = 0$ even.

CLOSURES

1) Positive closure (+) : $\Sigma = \{a,b\}$

$\Sigma^+ = \{a, b, aa, ab, ba, bb, \dots\}$

- It is combination of all strings that are formed by using given alphabet symbols.

2) Kleene closure (*) : $\Sigma = \{a,b\}$

$\Sigma^* = \{\epsilon, a, b, aa, ab, ba, bb \dots\}$

- It is combination of all strings that are formed by using given alphabet symbols including "ϵ".

NOTE: Positive closure never contains an empty string i.e., 'ϵ'. but Kleene closure must contain an empty string (ϵ).

* Relationship b/w L^+ and L^* :

$$\boxed{L^* = L^+ \cup \{\epsilon\}}$$

- $L^* = L^+ + \{\epsilon\}$. , Here $+$ and $\cup$ are same.

→ $L_1 = \{a, b\}$ $L_2 = \{c, d\}$

$L_1 \cdot L_2 = \{ac, ad, bc, bd\}$

→ $L = \{a, b\}$.

$L^2 = L \cdot L = \{aa, ab, ba, bb\}$. similarlly $L^3 = L^2 \cdot L$ and so on.

Q) $\Sigma = \{a, b\}$. Design a language which contains strings starting with b.

sol: $L_1 = \{bab, ba, baa, bb, bbb, baaa, baba, \dots\}$

Q) $\Sigma = \{a, b\}$, Design a Language which contains strings ending with 'b'.

sol: $L_2 = \{ab, bb, aab, abb, bbb, baab, \dots\}$

Q) Design a language which contains strings ~~whic~~ 'b' ($\Sigma = \{a, b\}$).

sol: $\boxed{L = L_1 \cup L_2}$

→ Exercise :

$L_1 = \{abc, bba, abb, abab\}$

$L_2 = \{aa, bc, dd, bb\}$

$L_1 \cdot L_2 = \{abcaa, abcbc, abcdd, abcbb, bbaaa, bbabc, bbadd, bbabb, abbaa, abbbc, abbdd, abbbb, ababaa, ababbc, ababdd, ababbb\}$

$L_2 \cdot L_1 = \{aaabc, aabba, aaabb, aaabab, bcabc, bcbba, bcabb, bcabab, ddabc, ddbba, ddabb, ddabab, bbabc, bbbba, bbabb, bbabab\}$

$L_1^+ =$

$L_1^* =$

Grammar (G):

- It is a four tuple
- It is represented by G(V, T, P, S)

Here; V = finite set of variables (or) non-Terminals.

T = Finite set of Terminals.

P = Finite set of Productions .. where it is of the form. $\alpha \rightarrow \beta$

$\Downarrow$

Production (or) Grammar rule.

S = It is starting variable in the given Productions

Eg:

$A \rightarrow abB$

$B \rightarrow bD / f$

$D \rightarrow d$.

$B \rightarrow bD / f$

$\Rightarrow B \rightarrow bD$

$\Rightarrow B \rightarrow f$

$V = \{A, B, D\}$

$T = \{a, b, d, f\}$

$P = \{A \rightarrow abB, B \rightarrow bD, B \rightarrow f, D \rightarrow d\}$

$S = A$.

NOTE :-

- The ultimate goal of any grammar is to produce strings. If in case a Grammar is not able to produce a string, then G is called Useless grammar