# Unit-10 Recursively Enumerable Language.

## (10.1) Definition:

⇒ A Language L is Recursively enumerable, if there existing Turing Machine can accept the Language L.

⇒ REL Divide in two categories.

① In this category TM always halt on valid input and enter in accept state but on invalid i/p it halts without entering in halt state. This type of Language called as recursive Language. (RL)

② In this category, the Language can be modeled by TM but there is no guarantee that the TM will eventually halt state.

⇒ We cannot predict the TM will go into halt state or enter in an infinite loop for the certain i/p. Language of this category are called Recursively Enumerable Language (REL)

## (10.2) Properties of RL & REL

① The complement of Recursive Language is recursive.

② The union of two recursive Language is recursive.

③ The union of two recursively enumerable Language is recursively enumerable.

④ If Language L and its complement $\bar{L}$ are both recursively enumerable, then L & $\bar{L}$

(2) are Recursive.

(5) If L is recursive Language than $\Sigma^* - L$ is Recursive.

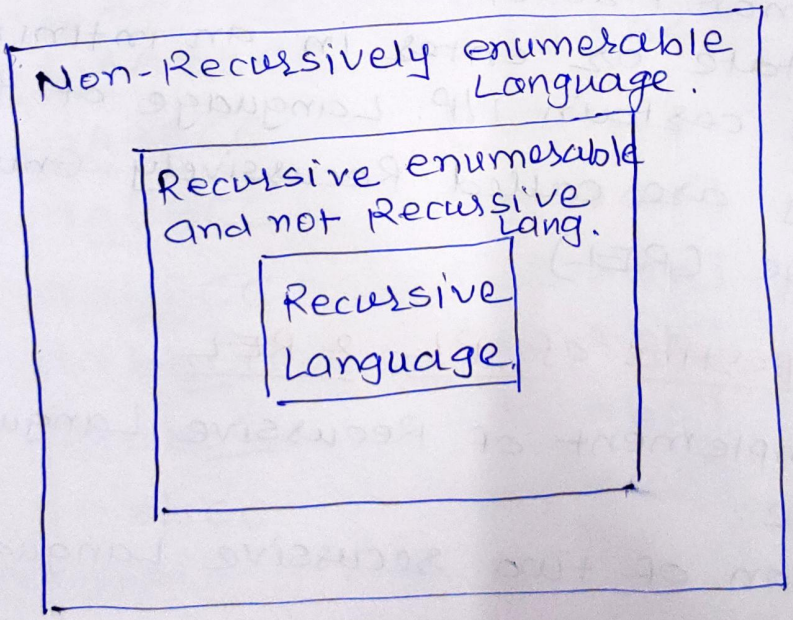
## (10.3) Decidable and undecidable Language

⇒ If a Language is recursive then it is called decidable Languages and if the Language is not recursive then such a Language is called undecidable Language.

⇒ These are three categories of the Languages

(1) Recursive Language
(2) Recursive enumerable and not Recursive Language.
(3) Non-Recursively enumerable Language.

⇒ Relationship betⁿ 3 Languages.



## * Examples

**Ex:1** Find the unrestricted grammar generating Language. $L = L(ABC)^n$

**Ans** for $n=1$ $\Rightarrow L = L(ABC)^1$

$S \rightarrow \underline{LA}BC$
$\rightarrow a\underline{BC}$ [LA → a]
$\rightarrow \underline{ab}C$ [aB → ab]
$\rightarrow abc$ [bC → bc]
Accepted.

for $n=2$.

$S \rightarrow L(ABC)^2$
$\rightarrow L(ABC)(ABC)$
$\rightarrow L\,AB\underline{CA}BC$
$\rightarrow L\,A\underline{BA}CBC$ [CA → AC]
$\rightarrow L\,AA\underline{BC}BC$ [BA → AB]
$\rightarrow \underline{LA}ABBCC$ [CB → BC]
$\rightarrow a\underline{AB}BCC$ [LA → a]
$\rightarrow aa\underline{BB}CC$ [aA → aa]
$\rightarrow aab\underline{bC}C$ [aB → ab]
$\rightarrow aabb\underline{cC}$ [bB → bb]
$\rightarrow aabbcc$ [bC → bc]
[cC → cc]
Accepted.

$\Rightarrow$ The final grammar

$S \rightarrow SABC \mid LABC$
$CA \rightarrow AC$
$BA \rightarrow AB$

(4)

$CB \rightarrow BC$

$LA \rightarrow a$

$aA \rightarrow aa$

$aB \rightarrow ab$

$bB \rightarrow bb$

$bC \rightarrow bc$

$cC \rightarrow cc$

Ex:2 A csg genesate for a Language

$L = \{a^n b^n c^n \mid n \geq 1\}$

⇒ string Acceptance → aabbcc

⇒ start with start symbol

$S \rightarrow SABC$

$\rightarrow SABCABC$ [S → SABC]

$\rightarrow aBCABC$ [SA → a]

$\rightarrow aBACBC$ [CA → AC]

$\rightarrow aABCBC$ [BA → AB]

$\rightarrow aABBCC$ [CB → BC]

$\rightarrow aaBBCC$ [aA → aa]

$\rightarrow aabBCC$ [aB → ab]

$\rightarrow aabbCC$ [bB → bb]

$\rightarrow aabbcC$ [bC → bc]

$\rightarrow aabbcc$ [cC → cc]

The CFG is

| | |
|---|---|
| $S \rightarrow SABC$ | $aA \rightarrow aa$ |
| $SA \rightarrow a$ | $aB \rightarrow ab$ |
| $CA \rightarrow AC$ | $bB \rightarrow bb$ |
| $BA \rightarrow AB$ | $bC \rightarrow bc$ |
| $CB \rightarrow BC$ | $cC \rightarrow cc$ |

Ex:3 Find the unrestricted grammar generating Language.

$$L = \{a^{2k} \mid k \in N\}$$

⇒ Let $L = \{a^{2^k} \mid k \in N\}$ be a Recursively defined Language by

$a \in L$ and for every $n \geq 1$ if

$a^n \in L$

then $a^{2n} \in L$

⇒ $S \to LaR$

$L \to LD$ [D means Double, $D = Da$]

$Da \to aaD$

$DR \to RA$

$L \to \Lambda$

$R \to \Lambda$