# Unit 3 Grammer

## Classification of Grammer

Noam chomsky classification

Type 0 → Unrestricted Grammer → Recurrsive Enumerable language → Turing Machine.

Type 1 → Content sensitive Grammer → Content sensitive language → Linear bounded Autometa.

Type 2 → Content free Grammer → Content free language → Push Down Autometa

Type 3 → Regular Grammer → R.L → FSM (Finite State Machine)

**Grammer:** A grammer can be formally defined by 4 tuples

Tuples are V, T, S, P.

V = set of variable or nonterminals.

T = set of terminal symbols.

S = start symbol

P = production Rule for terminals & nonterminals. $\alpha \rightarrow \beta$

- **Non-Terminal symbol** are those symbols which take part in the generation of sentence but it is not the component of sentence. There are represented by Capital letters (eg: A, B, C, X, Y, Z)
- **Terminal symbol** are the component of sentence & represented by small letters. (eg: a, b, c, ... z)

Q) $G_1 = (\{S, A, B\}, \{a, b\}, S, \{S \rightarrow AB, A \rightarrow a, B \rightarrow b\})$

$V = V_n = \{S, A, B\}$

$T = \Sigma = \{a, b\}$

$S = S$

$P = \alpha \rightarrow \beta$

$S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow b$

$L(G_1) = \{ab\}$

$S \rightarrow AB$

$\rightarrow aB$ [By $A \rightarrow a$]

$\rightarrow ab$ [By $B \rightarrow b$]

*A production has a form $\alpha \to \beta$ where $\alpha$ & $\beta$ are strings on V ∪ T and atleast one symbol of $\alpha \in V$.

**Regular Grammer:-** That generate regular language.

It has 2 types. They are

1) Left linear Grammer
2) Right linear Grammer.

**Left linear Grammer**

$A \to Bx$

$A, B \in V$

$x \in T$

Non terminal is on left hand side so it is L.L.G. i.e 'B'

**Right linear Grammar**

$A \to xB$

Non terminal is on right hand side so it is R.L.G. i.e 'B'

**Derivation of Grammer:**

Set of all stream that can be derived from a grammer is said to be language generated by a grammer.

Ex: $G = (\{S, A\}, \{a, b\}, S, S \to aAb$

$aA \to aaAb$

$A \to \epsilon$

$V = \{S, A\}$

$T = \{a, b\}$

$S \to aAb$

$\to a\epsilon b$ [By $A \to \epsilon$]

$\to ab$

$S \to aAb$

$\to aaAbb$ [By $aA \to aaAb$]

$\to aaaAbbb$ [By $aA \to aaAb$]

$\to a^n A b^n$

$\to a^n \epsilon b^n$ [By $A \to \epsilon$]

$L(G) = \{a^n b^n \mid n \geq 1\}$

## Recursive & Enumerable set

A set 's' is said to be recursive enumerable if there is a procedure to find elements belongs to 's'.

Ex:- $S \rightarrow OSA12$

$S \rightarrow 012$

$2A1 \rightarrow A12$

$1A1 \rightarrow 11$

Check whether it is recursive enumerable for

(A) 00112

(B) 001122

Ans $T = \{0, 1, 2\}$

$V = \{S, A\}$

$S \rightarrow OSA12$

$\rightarrow 0012A12$ [S → 012]

$\rightarrow 001A122$ [2A1 → A12]

$\rightarrow 001122$ [1A1 → 11]

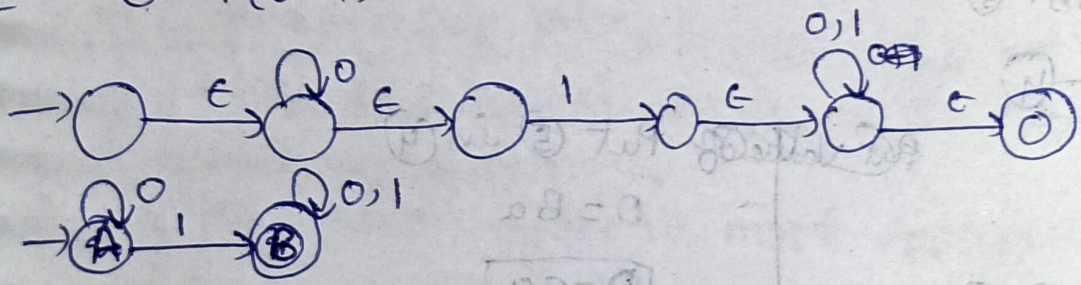
## RE to RG

Steps

1) Convert R.E to FA by Elimination method.

2) Convert F.A to R.G.

Ex:- $0^* 1 (0+1)^*$

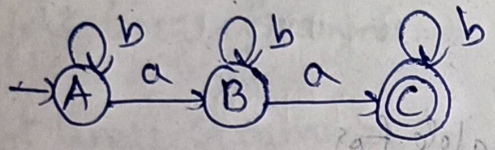



$S = \{A\}$

$A \rightarrow 0A$
$A \rightarrow 1B$
$A \rightarrow 1$

$B \rightarrow 0B$
$B \rightarrow 1B$
$B \rightarrow 0$
$B \rightarrow 1$

Ex. $b^*ab^*ab^*$




$A \rightarrow bA$ $B \rightarrow bB$ $C \rightarrow bC$

$A \rightarrow aB$ $B \rightarrow aC$ $C \rightarrow b$

$B \rightarrow a$

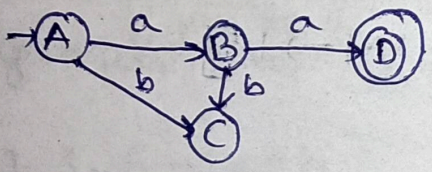
## RG to RE

1) Convert RG to FA

2) Convert FA to R.E by Arden's Theorem.

Q) $G = \{\{A, B, C\}, \{a, b\}, \{A\}\}$

$A \rightarrow aB$
$A \rightarrow bC$
$B \rightarrow a$
$B \rightarrow bC$.




aa

$A = \epsilon$ —①

$B = Aa$ —②

$C = Ab + Bb$ —③

$D = Ba$ —④

①in② Put ⑤ in ④

$B = Aa$

$B = \epsilon a$ [$\epsilon R = R$]

$B = a$ —⑤

$D = Ba$

$\boxed{D = aa}$