11|18

UNIT - III

Regular Grammar

Regular Grammar:-

Regular Grammar $G = (V, \Sigma, P, S)$

where V = a finite-set of variables (or) Non-Terminal

Σ = a finite set of input symbols.

S = start symbol

P = set of productions

P is in the form of either terminal followed by non-terminal (or) P is in the form of either non-terminal followed by terminal.

* $P \rightarrow T\,NT$ (or) [Right linear]

  $P \rightarrow NT\,T$ (or) [left linear]

Note:-

* ϵ is allowed.
* $S \rightarrow \epsilon$ is not allowed, if allowed S may not appear on RHS of any other production.

Equivalence between Regular Grammar and Finite Automata:-

Regular Grammar to FA:-

Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA and R be a Regular Grammar accepted by DFA.

1

Construction of DFA from Regular Grammar as follows:-

Step-1:- If There is a production $A_i \to a$ Then Add $\delta(q_i, a) = q_j$ in DFA, where q_j does not belongs to F ($q_j \notin F$).

Step-2:- If $A_i \to aA_j$ and $A_i \to a$ Then A $\delta(q_i, a) = q_j$ where $q_j \in F$.

* construct The DFA for The following R.E

$S \to aB \mid bA$
$A \to a \mid aS$ ~~bA~~
$B \to bS \mid b$.

sol:- i) consider $S \to aB$

This is in The form of $A_i \to aA_j$ Then Add $\delta(q_i, a) = q_j$

for $S \to aB$, add $\delta(S, a) = B$.

ii) consider $S \to bA$

for $S \to bA$, add $\delta(S, b) = A$.

iii) consider $A \to aS \mid a$,

for $A \to aS \mid a$, add $\delta(A, a) = S$.

Here S is final state.

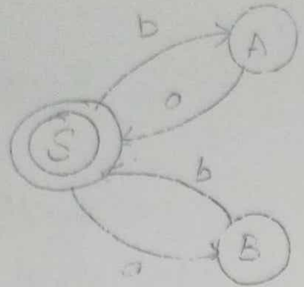
iv) consider $B \to bs \mid b$

for $B \to bS \mid b$, add $\delta(B, b) = S$.

Here S is final state.

$\delta(S, a) = B$
$\delta(S, b) = A$
$\delta(A, a) = S$
$\delta(B, b) = S$

te Relation:- The next state of an ...



Second Procedure:-

Step-1:- Let $G = (V, \Sigma, P, S)$ be a Regular Grammar we can construct DFA M whose i) Initial state corresponds to A_0. Variables are renamed to $A_0, A_1, \ldots A_n$.

ii) States corresponds to variables

Step-2:- If There is a production of The form $A_i \rightarrow a$, The corresponding transition terminates at a new state. This is The unique final state. Thus, The δ is defined for The remaining productions as

i) Each production $A_i \longrightarrow aA_j$ induces a transition from q_i to q_j with label A.

ii) Each production $A_i \rightarrow a$ induces a transition from q_i to q_f with label A.

$\therefore$ L(G) can be given by corresponding FA 'M'.

* construct a Finite Automata Recognising L(G) where G is The Grammar

$S \longrightarrow aS \mid bA \mid b$

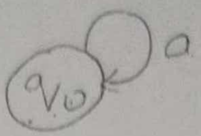
$A \longrightarrow aA \mid bS \mid a$

Sol: rename S as A_0 and A as A_1, rewrite The production

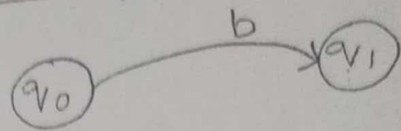
$A_0 \longrightarrow aA_0 \mid bA_1 \mid b$.

$A_1 \rightarrow aA_1 \mid bA_0 \mid a$.

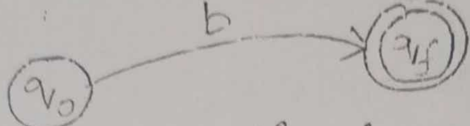
i) consider $A_0 \rightarrow aA_0$.



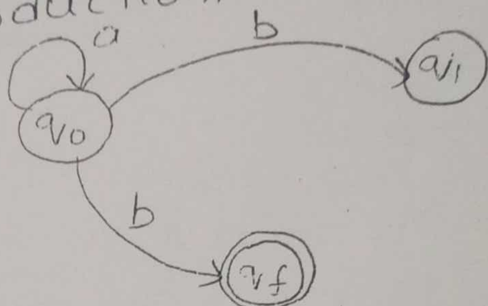
ii) consider $A_0 \rightarrow bA_1$



iii) consider $A_0 \rightarrow b$.



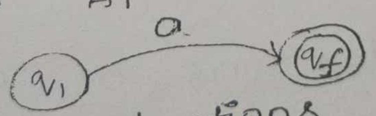
After A_0 productions FA is.



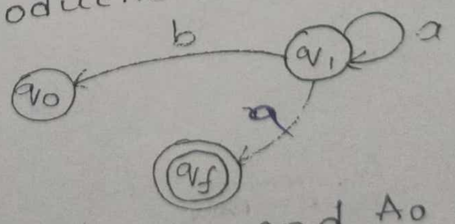
iv) consider $A_1 \rightarrow aA_1$

v) consider $A_1 \rightarrow bA_0$

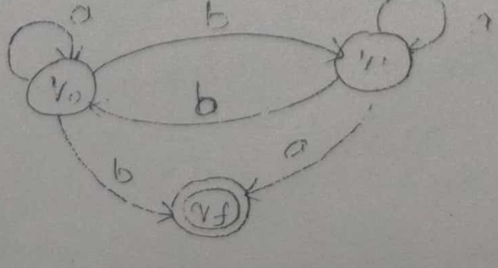
vi) consider $A_1 \rightarrow a$



After A_1 productions FA is.



After combining A_1 and A_0, we get, The D
is



(2)

$S \longrightarrow aS \mid bS \mid aA$

$A \longrightarrow bB$

$B \longrightarrow aC$

$C \longrightarrow a$

Sol:-

Rename S as A_0, A as A_1, B as A_2 and C as A_3.

$A_0 \longrightarrow aA_0 \mid bA_0 \mid aA_1$ 10/2/18

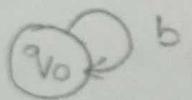
$A_1 \longrightarrow bA_2$

$A_2 \longrightarrow aA_3$

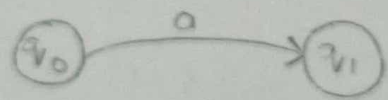
$A_3 \longrightarrow a$.

i) consider $A_0 \longrightarrow aA_0$

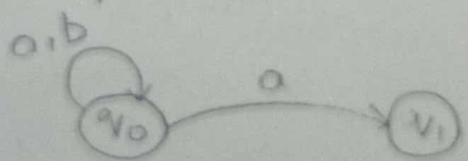
ii) consider $A_0 \longrightarrow bA_0$.



iii) consider $A_0 \longrightarrow aA_1$



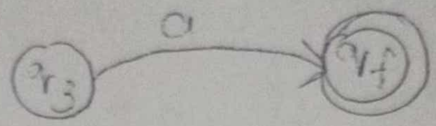
After A_0 productions of FA is



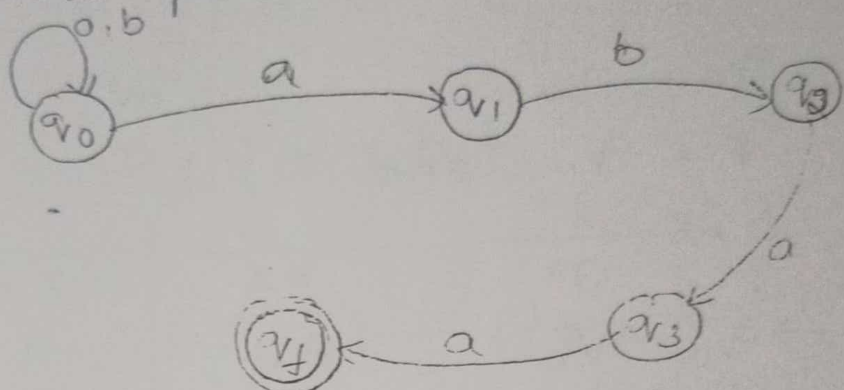
iv) consider $A_1 \longrightarrow bA_2$.

v) consider $A_2 \longrightarrow aA_3$.

vi) consider $A_3 \longrightarrow a$



The complete production is.



10/2/18 **Regular Grammar to Finite Automata :-**

Procedure :-

* Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA The equivalent Grammar 'G' can be constructed from This DFA such That, productions should corresponds to transitions.

Procedure :-

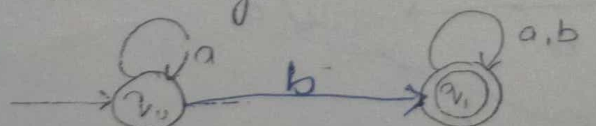
* P is The set of production Rules can be defined by following Rules :-

i) $A_i \longrightarrow aA_j$ is a production add to 'G' if $\delta(q_i, a) = q_j$. does not belongs to $F (q_j \notin F)$.

ii) $A_i \longrightarrow aA_j$ and $A_i \rightarrow a$ are The production, added to 'G' iff $\delta(q_i, a) = q_j \in F$.

Example :-

* construct equivalent Regular Grammar for The following FA ?



Relation:- The next state of an instant of time is

Given data,

$M = (Q, \Sigma, \delta, q_0, F)$

where $Q = \{q_0, q_1\}$

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

$q_0 = q_0$

$F = q_1$ and

δ is

	a	b
$\rightarrow q_0$	q_0	q_1
q_1	q_1	q_1

* $\delta(q_0, a) = q_0 \notin F$ Next line
* $\delta(q_0, b) = q_1 \in F$
* $\delta(q_1, a) = q_1 \in F$
* $\delta(q_1, b) = q_1 \in F$

i) Add The production $A_0 \longrightarrow aA_0$.

ii) Add The production $A_0 \longrightarrow bA_1$ and $A_0 \rightarrow b$.

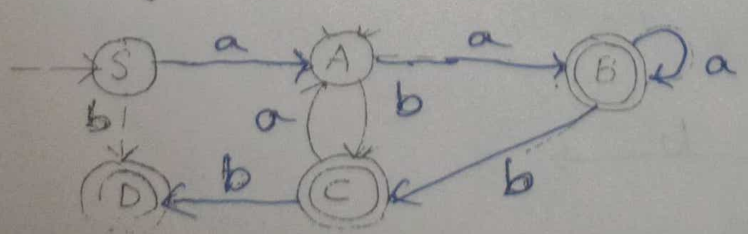
iii) Add The production $A_1 \longrightarrow aA_1$ and $A_1 \rightarrow a$

iv) Add The production $A_1 \rightarrow bA_1$ and $A_1 \rightarrow b$.

Now The Regular Grammar productions are

$$\boxed{\begin{array}{l} A_0 \longrightarrow aA_0 \\ A_0 \longrightarrow bA_1 / b \\ A_1 \longrightarrow aA_1 / a \\ A_1 \longrightarrow bA_1 / b. \end{array}}$$

* construct The Regular Grammar for The following FA?



sol^n: Given data,

$M = (Q, \Sigma, \delta, q_0, F)$

where $Q = \{S, A, B, D, C\}$

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

$q_0 = S$

$F = \{B, C, D\}$.

δ is

	a	b
$\rightarrow S$	A	D
A	$\{B, \}$	$\{\ , C\}$
B	$\{B, \}$	C
C	A	D
D	ϕ	ϕ

* $\delta(S, a) = A \notin F$

  Add The production $A_0 \rightarrow a A_1$

* $\delta(S, b) = D \in F$

  Add The production $A_0 \rightarrow b A_4$ & $A_0 \rightarrow b$

* $\delta(A, a) = B \in F$

  Add The production $A_1 \rightarrow a A_2$ and $A_1 \rightarrow a$

* $\delta(A, b) = C \in F$

  Add The production $A_1 \rightarrow b A_3$ and $A_1 \rightarrow b$

* $\delta(B, a) = B \in F$

  Add The production $A_2 \rightarrow a A_2$ and $A_2 \rightarrow a$

* $\delta(B, b) = C \in F$

  Add The production $A_2 \rightarrow b A_3$ and $A_2 \rightarrow b$

* $\delta(C, a) = A \notin F$

  Add The productions $A_3 \rightarrow a A_1$

* $\delta(C, b) = D \in F$

  Add The productions $A_3 \rightarrow b A_4$ and $A_3 \rightarrow b$

$\rightarrow$ b.
$\rightarrow$ a
b.
are

The total productions are

$A_0 \longrightarrow aA_1 | bA_4 | b$

$A_1 \longrightarrow aA_2 | a | bA_3 | b$

$A_2 \longrightarrow aA_2 | a | bA_3 | b$

$A_3 \longrightarrow aA_1 | bA_4 | b$

Context free Grammar :-

Let $G = (V, \Sigma, P, S)$ is context free if the productions of the form $\alpha \rightarrow \beta$ follows the following rules.

i) $|\alpha| \leq |\beta|$

ii) $|\alpha| = 1$, $\alpha \in V$

iii) Null (ϵ) productions are allowed.

Example:-

* write a context free Grammar for the string no. of a's followed by no. of b's. 14/2/18, 14M {imp. *****

Sol:- $L = \{\epsilon, ab, aabb, aaabbb, aaaabbbb \ldots$

The Grammar is

$S \longrightarrow aSbaSb / \epsilon$

* $S \rightarrow \epsilon$ is a production.

$S \longrightarrow aSb$

* Substitute $S \rightarrow \epsilon$ on RHS of $S \rightarrow aSb$

$S \rightarrow ab$.

* Substitute $S \rightarrow aSb$ in RHS of $S \rightarrow aSb$.

$S \rightarrow aaSbb \rightarrow$ Substitute $S \rightarrow \epsilon$
($\because S \rightarrow \epsilon$)

$S \rightarrow aabb$.

* Design context free grammar for pal drome?

sol:- $S \rightarrow aSa / bSb / a / b$

Substitute $S \rightarrow a$ in $S \rightarrow aSa$.
($\because S \rightarrow a$).

Then $S \rightarrow aaa$

* Design CFG to accept $a^n b^{2n}$?

sol:- $L = \{\epsilon, abb, aabbbb, aaabbbbbb$ ....

$S \rightarrow aSbb / \epsilon$

sub. $S \rightarrow \epsilon$ in $S \rightarrow aSbb$

$S \rightarrow abb$.

Substitute $S \rightarrow aSbb$ in $S \rightarrow aSbb$.

$S \rightarrow aaSbbbb$

Now $S \rightarrow \epsilon$, Then

$S \rightarrow aabbbb$.

14/2/18 Leftmost derivation And Rightmost derivati

14M Syntax tree (or) parse (or) derivation tree
Imp
****

Definition:- The derivation process ca be shown pictorially as a tree, such trees are called derivation trees. Thes trees shows clearly how the symbols of terminal strings are grouped into substrings, each of which belongs to the language of one of the variables of grammer

* For constructing a parse tree for a grammer $G = (V, \Sigma, P, S)$.

i) root is the start symbol

ii) Each internal node is marked by variaty in V.

Relation: The ... of ...

iii) Each node is labelled by either a variable, terminal (or) Epsilon (ϵ).

iv) if an internal node is labelled A, and its children are labelled $X_1, X_2, \ldots X_n$, respectively. From the left, $A \rightarrow X_1, X_2 \ldots X_n$ is a production in P.

Left most derivation:- The leftmost non-terminal in a working string is the first non-terminal that we encounter, when we scan the string from left to right.

Right most derivation:- The Rightmost non-terminal in a working string is the first non-terminal that we encounter, when we scan the string from ~~left~~ Right to left.

Note:- Some grammars are implemented (or) derived using LMD only.

* Some grammars are implemented (or) derived using RMD only.

* Some grammars are implemented (or) derived using LMD & RMD.. such grammars are called Ambiguous grammars.

Ambiguous Grammar:-

A CFG Grammar is said to be Ambiguous if and only if

i) if there exists more than one LMD.

ii) if there exists more than one RMD

iii) if there exists LMD and RMD

Sol:-

iv) if There exists more Than one syntax tree.

Problem:-

* construct a derivation tree for the following CFG to derive The string baaba.

$S \rightarrow \cancel{AAA} / AA$

$A \rightarrow AA / aA / Ab / a / b.$

Sol:- Given data,

$G = (V, \Sigma, P, S)$ where

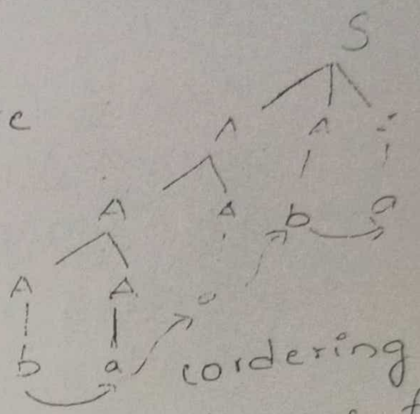
$V = \{S, A\}$

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

$S = S$

$P = S \longrightarrow AAA / AA$

$A \longrightarrow AA / aA / Ab / a / b.$



cordering of leaves from left to right

Leftmost derivation:-

(diagram ..p).

$S \longrightarrow \underline{A}AA$

$\longrightarrow \underline{A}AAA \quad [\because A \longrightarrow AA]$

$\longrightarrow \underline{A}AAAA \quad [\because A \longrightarrow AA]$

$\longrightarrow b\underline{A}AAA \quad [\because A \rightarrow b]$

$\longrightarrow ba\underline{A}AAA \quad [\because A \rightarrow aA]$

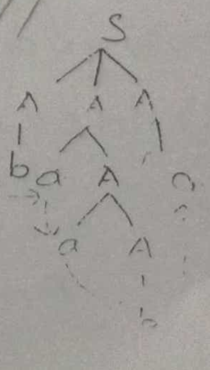
$\longrightarrow baa\underline{A}A \quad [\because A \rightarrow a]$

$\longrightarrow baab\underline{A} \quad [\because A \rightarrow b]$

$\longrightarrow baaba \quad [\because A \rightarrow a]$ //

another method.

LMD.



$S \longrightarrow \underline{A}AA$

$\longrightarrow b\underline{A}A \quad [\because A \rightarrow b]$

$\longrightarrow ba\underline{A}A \quad [\because A \rightarrow aA]$

$\longrightarrow baa\underline{A}A \quad [\because A \rightarrow aA]$

$\longrightarrow baab\underline{A} \quad [\because A \rightarrow b]$

Relation:- The next state of a ... of time is

Right Most derivation:-

S ⟶ AAA

S ⟶ AAa [∵ A → a]

S ⟶ Aba [∵ A → b]

S ⟶ AAAba [∵ A → AAA]

S ⟶ AAaba [∵ A → a]

S ⟶ Aaaba [∵ A → a]

S ⟶ baaba [∵ A → b].

S → AAA

S → AAAA [∵ A → AA]

S → AAAAA [∵ A → AA]

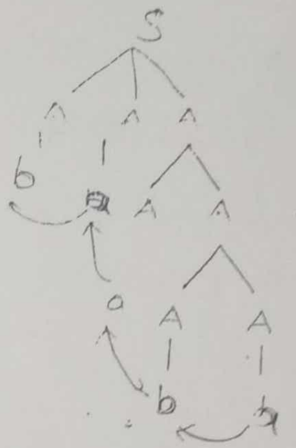
S → AAAAa [∵ A → a]

S → AAAba [∵ A → b]

S → AAaba [∵ A → a]

→ Aaaba [∵ A → a]

→ baaba.. [∵ A → b]



118 **Ordering The leaves:-**

This technique is used when The derivation tree associated with more Than two children. Here first it visits The leftmost leaf and visited The next leafs from Next leafs from left to right and it does not depends on levels of tree.

Prove The following Grammar is Ambiguous for The strings A) id+id*id B) id*id+id

$E \longrightarrow E+T / T$

$T \rightarrow T*F / F$

$F \rightarrow id$.

$G = (V, \Sigma, P, S)$

where

$V = \{E, T, F\}$

$\Sigma = \{id, +, *\}$

$S = E$

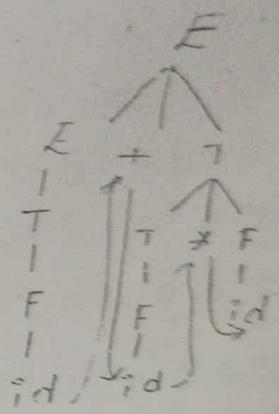
$P = E \rightarrow E+T \,/\, T$

$T \rightarrow T*F \,/\, F$

$F \rightarrow id.$

Left Most derivation:-

a) $id + id * id.$



$E \rightarrow \underline{E}+T$

$\rightarrow \underline{T}+T \quad [E \rightarrow T]$

$\rightarrow \underline{F}+T \quad [T \rightarrow F]$

$\rightarrow id + \underline{T} \quad [F \rightarrow id]$

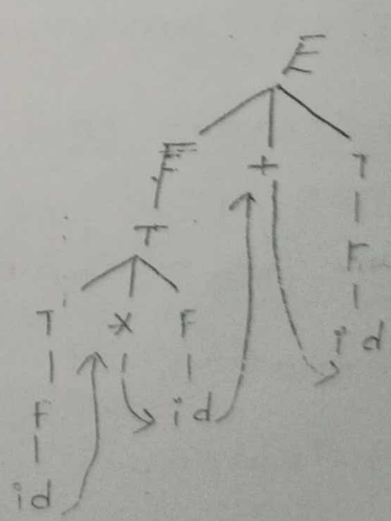
$\rightarrow id + \underline{T}*F \quad [T \rightarrow T*F]$

$\rightarrow id + \underline{F}*F \quad [T \rightarrow F]$

$\rightarrow id + id * \underline{F} \quad [F \rightarrow id]$

$\rightarrow id + id * id.$

b) $id * id + id$



$E \rightarrow \underline{E}+T$

$\rightarrow \underline{T}+T \quad (\because E \rightarrow T)$

$\rightarrow \underline{T}*F+T \quad [\because T \rightarrow T*F]$

$\rightarrow \underline{F} * F+T \quad [T \rightarrow F]$

$\rightarrow id * \underline{F} + T \quad [F \rightarrow id]$

$\rightarrow id * id + \underline{T} \quad [F \rightarrow id]$

$\rightarrow id * id + \underline{F} \quad [T \rightarrow F]$

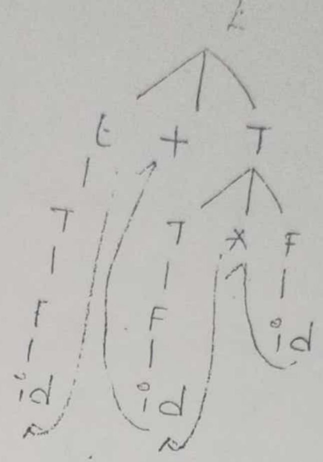
$\rightarrow id * id + id \quad [F \rightarrow id]$

$\rightarrow id * id + id.$

Right most derivation:-

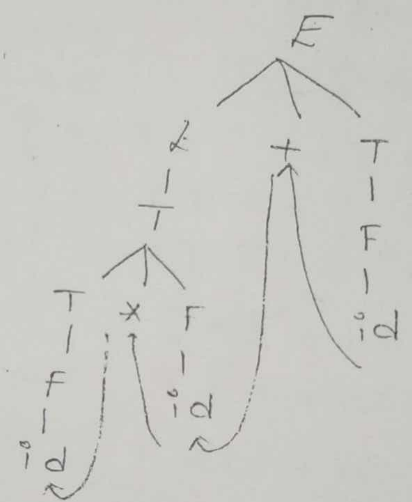
a) id + id * id

E → E + T [T → T*F]
→ E + T * F [F → id]
→ E + T * id [T → F]
→ E + F * id [F → id]
→ E + id * id [E → T]
→ T + id * id [T → F]
→ F + id * id [F → id]
→ id + id * id.



b) id * id + id.

E → E + T [T → F]
→ E + F [F → id]
→ E + id [E → T]
→ T + id [T → T*F]
→ T * F + id [T → F]
→ F * id + id [F → id]
→ id * id + id [F → id]
→ id * id + id.



Conclusion:- Here there exist left most derivation and Rightmost derivation for the given grammar. So, it is ambiguous.

Minimisation (or) Reduction of a Grammar:-

* Elimination of useless symbols
* Elimination of ε- productions
* Elimination of unit productions.
* Equivalence.

16/2/18 Elimination of useless symbols:-

* consider the grammar

$S \longrightarrow aA \mid bB$

$A \longrightarrow ba$.

Here the non-terminal B is used in R.H.S of other production but it has production. such type of symbols are called useless symbols.

useless symbol B.

After Elimination of useless symbol grammar is

$S \longrightarrow aA$

$A \rightarrow ba$.

* Elimination of useless production:-

* The non-terminal has production but it is not used in R.H.S of any other production. Then, such a production is called useless production.

Eg:-

$S \longrightarrow aA \mid bB$

$A \longrightarrow a$

$B \longrightarrow b$

$C \rightarrow c$

After elimination of c production, the grammar is

$S \longrightarrow aA \mid bB$

$A \longrightarrow a$

$B \rightarrow b$.

* Eliminate useless productions & useless symbols from the following Grammar.

$S \longrightarrow aA / bB$

$A \longrightarrow BE / a$

$B \longrightarrow C / b$

$E \longrightarrow d / \epsilon$.

tate Relation:- The next state of an instant of time is

useless symbol is ε
useless production is ε.
After Elimination The Grammar is

$S \longrightarrow aA \mid bB$

$A \longrightarrow a$

118 $B \longrightarrow b$.

Elimination of Null productions:-

* If The CFL consists of a word Epsilon Then, Add Epsilon productions in CFG.
* If CFG has null production and CFL does not need Epsilon Then Eliminate Epsilon productions from CFG.

Eg:-

$S \longrightarrow aX$

$X \longrightarrow \epsilon$.

* The derived language is {a}.
* For deriving This language only $S \to a$ is enough.

Procedure:- Step-1:-

* Identify The nullable variable.

Eg:- $S \longrightarrow AB$

$A \longrightarrow \epsilon$

$B \longrightarrow \epsilon$

* In a Given CFG a non-terminal 'X' is nullable if
* There is a production $X \longrightarrow \epsilon$
* There is a derivation That starts at X and leads to ε.

Eg:- S in The above Grammar.

Step-2: Nullable variable Substitution

* if $A \rightarrow \epsilon$ is a null production, A is nullable variable. Substitute The Nullable Variable by Epsilon if it is available on RHS of other productions and add all possible combinations on RHS.

1) Eg:- * Eliminate ϵ - productions in The Grammar.

$S \rightarrow ABac$
$A \rightarrow BC$
$B \rightarrow b/\epsilon$
$C \rightarrow D/\epsilon$
$D \rightarrow d.$

Sol:- * Identification of nullable Variables.

$B \rightarrow \epsilon$ and $C \rightarrow \epsilon$ are null productions.

So, nullable variables are = {B, C}

* After Substitution of B, C in A - production Then A becomes ϵ. So Add A to The nullable set.

nullable variables = {A, B, C}

* Substitution of nullable variables

a) $S \rightarrow ABac \mid \underset{A\rightarrow\epsilon}{Bac} \mid \underset{B\rightarrow\epsilon}{Aac} \mid \underset{A\rightarrow\epsilon, B\rightarrow\epsilon}{ac}$

b) $A \rightarrow BC \mid B \mid C$

c) $B \rightarrow b$

d) $C \rightarrow D$

e) $D \rightarrow d.$

2) * Eliminate ϵ productions in The Grammar.

$S \rightarrow ABA$
$A \rightarrow aA \mid \epsilon$
$B \rightarrow bB \mid \epsilon.$

Relation:- The next state of an ... instant of time is

" $A \to \epsilon$ and $B \to \epsilon$ are null productions
so nullable variables = {B, C}.
After substitution of A, B in S-production
Then S becomes ϵ. So Add A to nullable set.

nullable variables = {S, A, B}.

* Substitution of nullable variables

$S \to ABA \mid BA \mid AB \mid AA \mid A \mid B$

$A \to aA \mid a$

$B \to bB \mid b$.

Elimination of unit productions:-

* A single non-terminal on LHS is Replaced with another single non-terminal on RHS.

Eg:- $A \to B$ is a unit production.

Step-1:- For each pair of non-terminals A and B, such that there is a production $A \to B$ and the non-unit productions from B are $B \to \alpha_1 \mid \alpha_2 \mid \text{------} \mid \alpha_n$.
where $\alpha_i \in (V \cup \Sigma)^*$ are strings of terminals and non-terminals then create the new productions as $A \to \alpha_1 \mid \alpha_2 \mid \text{------} \mid \alpha_n$.

Step-2:- Repeat step-1 for all unit productions.

* Eg:-

$S \to A$

$A \to B$

$B \to b$

Given data,
The non unit production in the Grammar is $B \to b$.

Sol:-

Substitute This in $A \to B$, Then we will get $A \to b$. substitute $A \to b$ in $S \to A$

Then we will get $S \to b$.

* The Grammar after The elimination of unit productions is

$$\boxed{\begin{array}{l} S \to b \\ A \to b \\ B \to b \end{array}}$$

* Eliminate unit productions in The Grammar

$S \to A|bb$

$A \to B/b$.

$B \to a$

sol:- Substitute $B \to a$ in A Then we will get $A \to a|b$. substitute A in S

Then we will get

$S \to a|b|bb$.

~~Fore~~ After elimination of unit productions The Grammar is

$$\boxed{\begin{array}{l} S \to a|b|bb \\ A \to a|b \\ B \to a \end{array}}$$

Cyclic unit productions:-

Eg:- $A \to B$

$B \to C$

$C \to A$.

* The LHS ~~of a~~ variable is repeated on RHS of any production, such a type of productions are called cyclic productions.

The next ... time is

Eg:- Eliminate cyclic productions in the Grammar?

$S \rightarrow A | bb$

$A \rightarrow B | b$

$B \rightarrow A | a.$

Substitute $B \rightarrow a$ in A Then $A \rightarrow a | b$

Substitute A in $B \rightarrow A$. Then $B \rightarrow a | b$.

Substitute A in S, Then $S \rightarrow a | b | bb$

After elimination of cyclic production Then grammar is

$$\boxed{\begin{array}{l} S \rightarrow a|b|bb \\ A \rightarrow a|b \\ B \rightarrow a|b \end{array}}$$

118 * Eliminate cyclic productions in the grammar?

$S \rightarrow A | bb$

$A \rightarrow B | b$

$B \rightarrow S | a.$

Substitute $B \rightarrow a$ in A Then $A \rightarrow a | b$

Substitute $A \rightarrow a | b$ in S Then

$S \rightarrow a | b | bb.$

Substitute $S \rightarrow a | b | bb$ in $B \rightarrow S$ Then

$B \rightarrow a | b | bb.$

Substitute B in $A \rightarrow B$. then we get

$A \rightarrow a | b | bb.$

After elimination of cyclic unit Production Then grammar is

$$\boxed{\begin{array}{l} S \rightarrow a|b|bb \\ A \rightarrow a|b|bb \\ B \rightarrow a|b|bb. \end{array}}$$

Normal Forms:-

According to chomsky hierarchy The CFG is acceptable by PDA but in implementation it is not directly acceptable by PDA. So, we need to convert this CFG into some other form called Normal form. There are number of Normal forms among those we will discuss only two Normal forms because there are No. of restrictions imposed on RHS of production. The two Normal Forms are

* chomsky Normal form and
* Greibach Normal Form.

chomsky Normal Form:-

The production of the form $A \rightarrow \alpha$, where α consists of two non-terminals (or) single terminal.

Eg:- $A \rightarrow BC | a$

Procedure:- In this procedure we will have 3 steps.

Step-1:- Eliminate NULL and unit productions.

Step-2:- Elimination of terminals.

Introduce a new variable (or) non-terminal C_{ai} for terminal a_i as

$C_{a1} \rightarrow a_1$

$C_{a2} \rightarrow a_2$ -

Step-3:- Restriction of length.

To Restrict The no of variables on RHS, Introduce new variables and separate Them as follows.

Step-4:- Include The production of The form $A \rightarrow BC \mid a$ in the CNF.

118 * Convert The following CFG into CNF?

$$S \rightarrow bA \mid aB$$
$$A \rightarrow bAA \mid aS \mid a$$
$$B \rightarrow aBB \mid bS \mid a.$$

Step-1:- Eliminate null and unit productions. There are no null and unit productions in The given grammar.

Step-2:- Elimination of terminals.

i) consider $S \rightarrow bA$

replace The terminal b with C_b and add $C_b \rightarrow b$. Now The productions are

$$S \rightarrow C_bA.$$
$$C_b \rightarrow b.$$

$S \rightarrow bA$
$C_b \rightarrow b$
$S \rightarrow C_bA$
$C_b \rightarrow b$

ii) consider $S \rightarrow aB$. $S \rightarrow C_aB$ $S \rightarrow aB$

replace The terminal a with C_a and add $C_a \rightarrow a$. Now The productions are

$$S \rightarrow C_aB$$
$$C_a \rightarrow a.$$

iii) consider $A \rightarrow bAA$.

replace The terminal b with C_b. Now The Production is $A \rightarrow C_bAA$.

iv) consider $A \rightarrow aS$

replace The terminal a with C_a. Now The Production is $A \rightarrow C_aS$.

v) consider $B \rightarrow aBB$

replace the terminal a with C_a. Now the production is

$B \rightarrow C_a BB$.

vi) consider $B \rightarrow bS$ with C_b. Now replace the terminal b

The production is $B \rightarrow C_b S$.

After the elimination of terminals. The grammar is

$S \rightarrow C_b A \mid C_a B$

$A \rightarrow C_b AA \mid C_a S \mid a$

$B \rightarrow C_a BB \mid C_b S \mid a$

$C_a \rightarrow a$

$C_b \rightarrow b$.

Step-3:- Restriction of length. $A \rightarrow C_b AA$ and

All are in CNF except

$B \rightarrow C_a BB$.

i) consider $A \rightarrow C_b AA$. Replace AA with C_1 and add C_1 to AA. Now the producti are

$A \rightarrow C_b C_1$

$C_1 \rightarrow AA$.

ii) consider $B \rightarrow C_a BB$. Replace BB with C_2 and add C_2 to BB. Now the producti are

$B \rightarrow C_a C_2$

$C_2 \rightarrow BB$

The grammar in CNF is

$$\boxed{\begin{array}{l} S \rightarrow C_b A \mid C_a B \\ A \rightarrow C_b C_1 \mid C_a S \mid a \\ B \rightarrow C_a C_2 \mid C_b S \mid a \\ C_a \rightarrow a \\ C_b \rightarrow b \qquad C_2 \rightarrow BB \\ C_1 \rightarrow AA. \end{array}}$$

The next state of a ... time is

* convert The following CFG into CNF

$S \rightarrow ABA$

$A \rightarrow aA \mid \epsilon$

$B \rightarrow bB \mid \epsilon$

Step-1:- After The elimination of ϵ productions The grammar is

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid A \mid B.$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b.$

After The elimination of unit productions The grammar is

$S \rightarrow ABA \mid BA \mid AB \mid AA \mid aA \mid a \mid bB \mid b.$

$A \rightarrow aA \mid a$

$B \rightarrow bB \mid b.$

Step-2:- Elimination of terminals.

i) consider $S \rightarrow aA$
replace The terminal a with C_a and add C_a ~~to~~ to a. Now The productions are

$S \rightarrow C_a A$

$C_a \rightarrow a.$

ii) consider $S \rightarrow bB$
replace The terminal b with C_b and add $C_b \rightarrow b$. Now The productions are

$S \rightarrow C_b B$

$C_b \rightarrow b.$

iii) consider $A \rightarrow aA$
replace The terminal a with C_a. Now The production is

$A \rightarrow C_a A$

26/2/18 (14M) - V.V.V.Imp ****

iv) consider $B \rightarrow bB$

replace the terminal b with C_b. Now the production is

$$B \rightarrow C_b B.$$

After the elimination of terminals the grammar is

$$\boxed{\begin{array}{l} S \rightarrow ABA \mid BA \mid AB \mid AA \mid C_a A \mid a \mid C_b B \mid \\ A \rightarrow C_a A \mid a \\ B \rightarrow C_b B \mid b \\ C_a \rightarrow a \\ C_b \rightarrow b \end{array}}$$

Step-3:- Restriction of length.

All are in CNF except ~~A~~ $S \rightarrow ABA$.

consider $S \rightarrow ABA$.

Replace BA with C_1. Add the production $C_1 \rightarrow BA$. Now the Grammar is

$$S \rightarrow AC_1 \quad , \quad C_1 \rightarrow BA.$$

Now the Grammar in CNF is

$$\boxed{\begin{array}{l} S \rightarrow AC_1 \mid BA \mid AB \mid AA \mid C_a A \mid a \mid C_b B \mid b. \\ A \rightarrow C_a A \mid a \\ B \rightarrow C_b B \mid b \\ C_a \rightarrow a \\ C_b \rightarrow b \\ C_1 \rightarrow BA. \end{array}}$$

26/2/18 (14M) V.V.V.Imp ****

Greibach Normal form:-

Greibach Normal Form is also focused on RHS of the production. The production of the form $A \rightarrow ax \mid a$ is said to be in GNF.

The next state of an ... time is

Procedure:-

Step-1:- Eliminate Null and unit productions and convert it into CNF.

Step-2:- Rename the variables as $A_1, A_2 \ldots\ldots A_n$.

Eg:-

$S \longrightarrow AA$

$A \longrightarrow BS$

$B \longrightarrow b.$

Rename the variables S with A_1,
A with A_2 and
B with A_3
and Reconstruct the productions. with new variables.

$A_1 \longrightarrow A_2 A_2$

$A_2 \longrightarrow A_3 A_1$

$A_3 \longrightarrow b.$

Step-3:- if the production is in the form of $A_i \longrightarrow A_j \alpha$. Then,
compare i and j values.

i) if $j > i$, leave production as it is

ii) if $j = i$, apply Lemma 2.

iii) if $j < i$; apply Lemma 1.

Step-4:- Repeat step-3 for all productions which are in the form of $A_i \longrightarrow A_j \alpha$.

Step-5:- if any production is not in GNF, This for bringing those productions into GNF. Apply Lemma 1. i.e., ~~if~~ in the production of the form $A_i \rightarrow A_j\alpha$ if A_j is in GNF which is substituted in A_i for bringing A_i into GNF.

Lemma 1:- Lemma 1 (or) Substitution Rule

if any production is in the form of $A \rightarrow B\alpha$ And $B \rightarrow \beta_1 / \beta_2 / \ldots / \beta_n$

~~Then $A \rightarrow \beta_1\alpha / \beta_2\alpha / \ldots / \beta_n\alpha$~~

Then B is substituted in A production

i.e., $A \rightarrow B\alpha$, $B \rightarrow \beta_1 / \beta_2 / \ldots / \beta_n$

$A \rightarrow \beta_1\alpha / \beta_2\alpha / \ldots / \beta_n\alpha$.

Lemma 2 (or) Elimination of Left Recursi

~~if~~ The production is in the form of $A \rightarrow A\alpha / \beta$ is said to be Left Recursive production. After the eliminatio of Left Recursion the productions are

$A \rightarrow \beta A'$ $\Rightarrow$ $A \rightarrow \beta A' / \beta$

$A' \rightarrow \alpha A' / \epsilon$ $\qquad$ $A' \rightarrow \alpha A' / \alpha$

Problem:-

* Convert The following CFG into GNF.

$S \rightarrow AA / a$

$A \rightarrow SS / b$.

Sol: step 1:- There are no null And unit productions and the given grammer is in CNF.

time is

Step 2 :- Rename S with A_1 and A with A_2. Reconstruct the functions.

$$A_1 \to A_2A_2 \mid a$$

$$A_2 \to A_1A_1 \mid b.$$

Step-3 :-

i) consider the production $A_1 \to A_2A_2$. is in the form of $A_i \to A_j\alpha$.

$i=1,\ j=2,\ j>i$

Leave the production as it is.

ii) consider the production $A_2 \to A_1A_1$. is in the form of $A_i \to A_j\alpha$

$i=2,\ j=1\ ;\ j<i$

Apply lemma 1.

$A_2 \to A_1A_1$ $\qquad$ $A_1 \to \underset{\beta_1}{A_2A_2} \mid \underset{\beta_2}{a}$

$A \to B\alpha$ $\qquad$ $A_1 \to \beta_1\alpha \mid \beta_2\alpha.$

$$A_2 \to A_2A_2A_1 \mid aA_1.$$

consider $A_2 \to A_2A_2A_1$ is in the form of $A_i \to A_j\alpha$, $i=2, j=2, i=j$

Apply lemma 2.

$\underset{A}{A_2} \to \underset{A}{A_2}\underset{\alpha}{A_2A_1} \mid \underset{\beta_1}{aA_1} \mid \underset{\beta_2}{b}.$ $\qquad$ $A \to A\alpha \mid \beta_1 \mid \beta_2$

$A \to \beta_1A' \mid \beta_2A'$

$A' \to \alpha A' \mid \alpha.$

$A_2 \to aA_1A_2' \mid bA_2'$

$A_2' \to A_2A_1A_2' \mid A_2A_1$

*

~~step~~ 5 :- $A_1 \to A_2A_2 \mid a$

$\therefore A_2 \to aA_1A_2' \mid bA_2' \mid aA_1 \mid b.$

$A_2' \rightarrow A_2 A_1 A_2' \mid A_2 A_1.$

Step-5:- A_2 is in GNF, But A_1 and are not in GNF.

To bring A_1 and A_2' into GNF, Apply lem

consider $A_1 \longrightarrow A_2 A_2 \mid a$, apply lemmas

$A \rightarrow B\alpha$, $A_2 \longrightarrow aA_1A_2' \mid bA_2' \mid aA_1 \mid$

$B \rightarrow \beta_1 \mid \beta_2 \mid \text{-----} \mid \beta_n$

$A_1 \rightarrow \beta_1\alpha \mid \beta_2\alpha \mid \text{----} \mid \beta_n\alpha$

substitute A_2 in A_1

$A_1 \rightarrow aA_1A_2'A_2 \mid bA_2'A_2 \mid aA_1A_2 \mid bA_2 \mid$

consider $A_2' \rightarrow A_2A_1A_2' \mid A_2\underline{A_1}.$

substitute A_2 in A_2'

$A_2' \rightarrow aA_1A_2'A_1A_2' \mid bA_2'A_2A_1A_2' \mid aA_1A_1A_2' \mid$
$bA_1A_2' \mid aA_1A_2'A_1 \mid bA_2'A_1 \mid aA_1A_1 \mid b.$

Now The Grammar in GNF is

$$\boxed{\begin{array}{l} A_1 \rightarrow aA_1A_2'A_2 \mid bA_2'A_2 \mid aA_1A_2 \mid bA_2 \mid a \\ A_2 \rightarrow aA_1A_2' \mid bA_2' \mid aA_1 \mid b \\ A_2' \rightarrow aA_1A_2'A_1A_2' \mid bA_2'A_2A_1A_2' \mid aA_1A_1A_2' \mid \\ \quad bA_1A_2' \mid aA_1A_2'A_1 \mid bA_2'A_1 \mid aA_1A_1 \mid bA_1 \end{array}}$$

* $S \longrightarrow XA \mid b$

$A \longrightarrow a \mid \cancel{B} \; BS$

$B \longrightarrow b$

$X \longrightarrow a.$

convert The Grammar into GNF.

Step-1:- There are no null and unit productions in the grammar and the grammar is in the CNF.

Step-2:- Rename S with A_1, A with A_2, B with A_3, X with A_4. Reconstruct the Productions.

$A_1 \longrightarrow A_4 A_2 | b$

$A_2 \longrightarrow a | A_3 A_1$

$A_3 \longrightarrow b$

$A_4 \longrightarrow a.$

Step-3:-

i) consider the production $A_1 \longrightarrow A_4 A_2$
it is in the form of $A_i \longrightarrow A_j \alpha$.

$i = 1, j = 4, j > i$

leave the production as it is.

ii) consider the production $A_2 \longrightarrow A_3 A_1$
it is in the form of $A_i \longrightarrow A_j \alpha$

$i = 2, j = 3, j > i$

leave the production as it is.

Step-5:-

Since A_3 and A_4 are in GNF but $A_1 \longrightarrow A_4 A_2$ and $A_2 \longrightarrow A_3 A_1$.
To bring A_1 and A_2 in GNF Apply Lemma 1

consider $A_1 \longrightarrow \underbrace{A_4}_{B} \underbrace{A_2}_{\alpha}$

$A \rightarrow B\alpha$

$B \rightarrow \beta_1 | \beta_2 | \cdots \beta_n.$

$A \rightarrow \beta_1 \alpha | \beta_2 \alpha | \cdots | \beta_n \alpha$.

Substitute A_4 in A_1 Then

$A_1 \rightarrow aA_2$.

consider $A_2 \rightarrow A_3 A_1$

Substitute A_3 in A_2

$A_2 \rightarrow bA_1$

$A \rightarrow B\alpha$

$\rightarrow B \rightarrow \beta_1 | \beta_2 | \cdots | \beta_n$

$A \rightarrow \beta_1 \alpha | \beta_2 \alpha | \cdots$

Now The Grammar in GNF is

$$\boxed{\begin{array}{l} A_1 \rightarrow aA_2 | b \\ A_2 \rightarrow a | bA_1 \\ A_3 \rightarrow b \\ A_4 \rightarrow a. \end{array}}$$

* convert The following CFG into GNF.

$S \rightarrow XA | BB$

$B \rightarrow b | SB$

$X \rightarrow b$.

$A \rightarrow a$.

Sol: Step-1:- There are no null and unit productions and The grammar is in CNF.

Step-2:- Rename S with A_1, B with A_2, X with A_3 and A with A_4.

Reconstruct The productions.

$A_1 \rightarrow A_3 A_4 | A_2 A_2$

$A_2 \rightarrow b | A_1 A_2$

$A_3 \rightarrow b$

$A_4 \rightarrow a$.

Step 3:

i) consider The production $A_1 \to A_3A_4$, it is in The form of $A_i \to A_j\alpha$

$i = 1,\ j = 3$, $j > i$

leave The production as it is

ii) consider The production $A_1 \to A_2A_2$, it is in The form of $A_i \to A_j\alpha$

$i = 1,\ j = 2$, $j > i$

leave The production as it is

iii) consider The production $A_2 \to A_1A_2$, it is in The form of $A_i \to A_j\alpha$

$i = 2,\ j = 1$ $\quad j < i$

Apply lemma 1.

$A_2 \to \underbrace{A_1}_{B}\ \underbrace{A_2}_{\alpha}$.

$A \to B\alpha$

$B \to \beta_1 | \beta_2 | \cdots | \beta_n$

$A \to \beta_1\alpha | \beta_2\alpha | \cdots | \beta_n\alpha$

Substitute $A_1 \to \underbrace{A_3A_4}_{\beta_1} | \underbrace{A_2A_2}_{\beta_2}$

$A_2 \to A_3A_4A_2 | A_2A_2A_2$:

Now, consider $A_2 \to A_3A_4A_2$ it is in The form $A_i \to A_j\alpha$.

$i = 2,\ j = 3$, $j > i$

leave The production as it is.

consider $A_2 \to A_2A_2A_2$. it is in The form of $A_i \to A_j\alpha$

$i = 2,\ j = 2$ $\quad i = j$.

apply Lemma 2.

$A_2 \to A_2A_2A_2 | b$

$\underbrace{A_2}_{A} \to \underbrace{A_2}_{A}\underbrace{A_2A_2}_{\alpha} | \underbrace{b}_{\beta}$

$A \to \beta A'$

$A' \to \alpha A' | \alpha$

* The productions are

$A_2 \rightarrow bA_2' \mid b$

$A_2' \rightarrow A_2A_2A_2' \mid A_2A_2$

Now consider $A_2' \rightarrow A_2 A_2 A_2'$

* The productions of A_2' are

$A_2' \rightarrow bA_2' A_2 A_2' \mid bA_2 A_2'$

consider the production

$A_2' \rightarrow A_2 A_2$

* we get $A_2' \rightarrow bA_2' A_2 \mid bA_2$

on combining we get

$\boxed{A_2' \rightarrow bA_2'A_2A_2' \mid bA_2A_2' \mid bA_2'A_2 \mid bA_2}$

$\boxed{A_2 \rightarrow bA_2' \mid b}$

* we need to get the production A_1.

so consider the production

$A_1 \rightarrow A_3 A_4$

By applying lemma 1

$\boxed{A_1 \rightarrow bA_4}$ and

consider the production $A_1 \rightarrow A_2 A_2$

on Applying lemma 1

$\boxed{A_1 \rightarrow bA_2'A_2 \mid bA_2}$

Now the Grammar in GNF is

$A_1 \rightarrow bA_4 \mid bA_2' A_2 \mid bA_2$

$A_2 \rightarrow bA_2' \mid b$

$A_3 \rightarrow b$

$A_4 \rightarrow a$

$A_2' \rightarrow bA_2'A_2A_2' \mid bA_2A_2' \mid bA_2'A_2 \mid bA_2$

stack.

UNIT - IV

Push Down Automata.

* Let us consider $L = \{a^n b^n / n \geq 1\}$. This is a context free language but not Regular.

* A Finite Automata cannot Accept 'L' i.e. strings of the form $a^n b^n$ as it has to remember the no. of a's in a string

2/18 * $(q_0, x, z_0) \vdash (q_f, \epsilon\ \gamma)$ for Final state

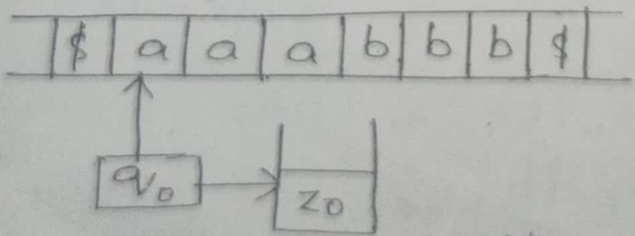
* $(q_0, x, z_0) \vdash (q, \epsilon, \epsilon)$ for empty stack.

1/18 Design a PDA which accepts $L = \{a^n b^n / n \geq 1\}$.

$L = \{ab, aabb, aaabbb, \text{---------}\}$.

$w = aaabbb$.

1) Reading of 1^{st} a.



push a into stack.

$\delta(q_0, a, z_0) = (q_0, az_0)$

2) Reading of 2^{nd} a