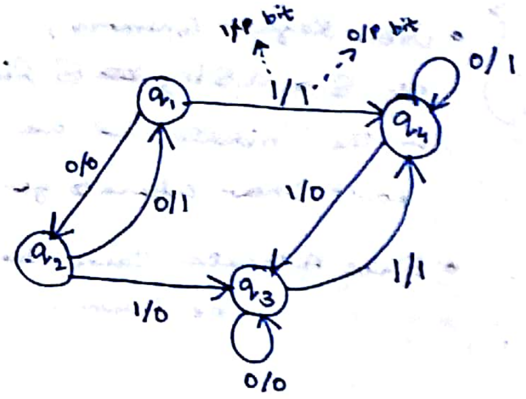
Mealy ~~Moore~~ Machine

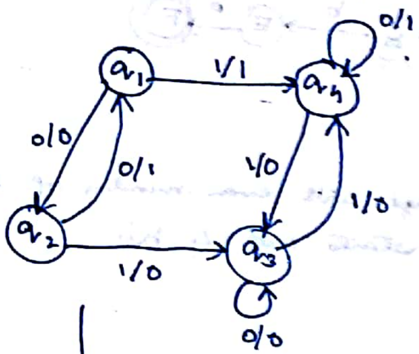
• Output → 1 → Successfull State
output → 0 → Unsuccessful State



Present State	Next State I/P = 0		Next State I/P = 1	
	State	O/P	State	O/P
q_1	q_2	0	q_4	1
q_2	q_1	1	q_3	0
q_3	q_3	0	q_4	1
q_4	q_4	1	q_3	0

Moore Machine

~~• Same diagram as Mealy Machine.~~

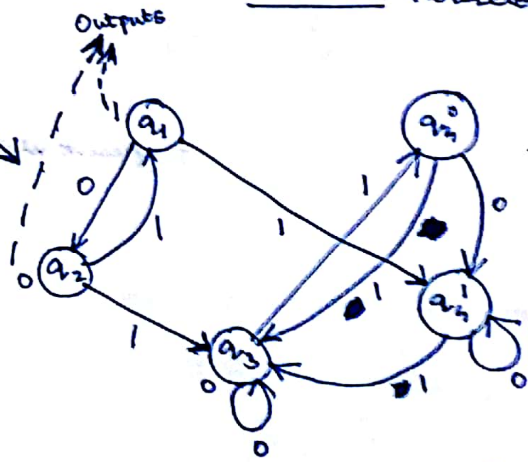


• In q_4, there are 3 incoming edges, 2 have output 1, and 1 has output 0.
∴ ambiguous.

Conversion from Mealy machine to Moore Machine

Solution: Partition

Moore Machine



• q_1, q_4 are the acceptable states, as their output is 1

Mealy → Moore

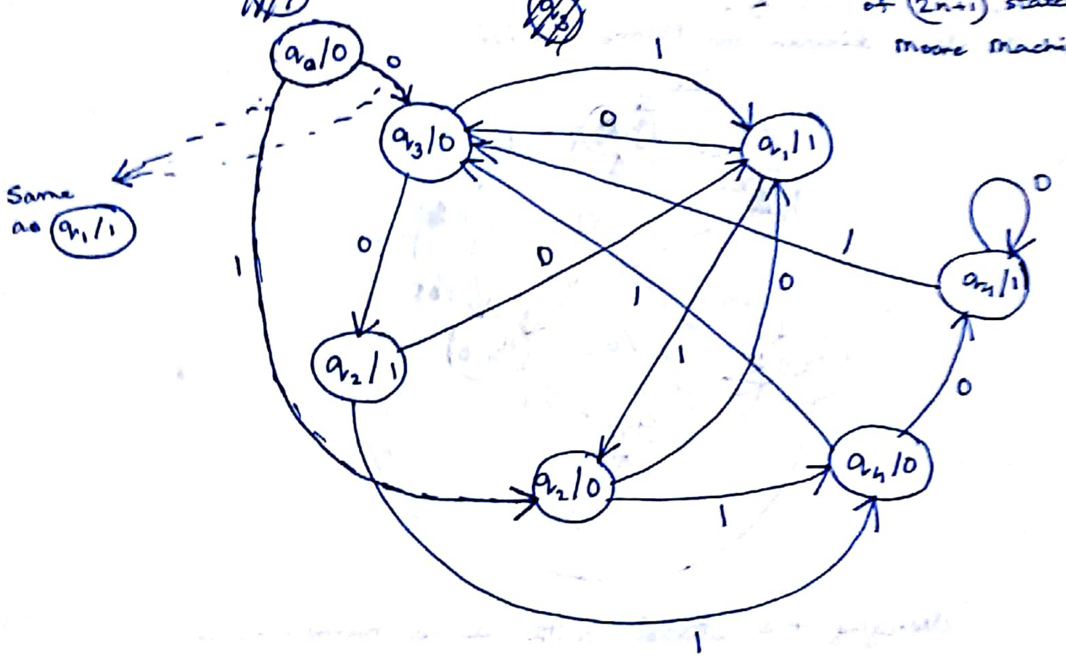
Example:

Table for Mealy Machine:

Present State	Next State			
	I/P = 0		I/P = 1	
	State	O/P	State	O/P
→ q_1	q_3	0	q_2	0
q_2	q_1	1	q_4	0
q_3	q_2	1	q_1	1
q_4	q_4	1	q_3	0

- If there are n states ~~...~~ in Mealy machine, there can be a maximum of $(2n+1)$ states in Moore Machine.

Moore Machine



∵ q_1 is the start state with output 1, λ is an acceptable state.

∴ if λ is not supposed to be included, we have to include another state $q_0/0$.

- for $L=\{\lambda, 01\} \longrightarrow \quad q_1 \xrightarrow{0} q_3 \xrightarrow{1} q_1$
- for $L=\{01\} \longrightarrow \quad q_0 \xrightarrow{0} q_3 \xrightarrow{1} q_1$

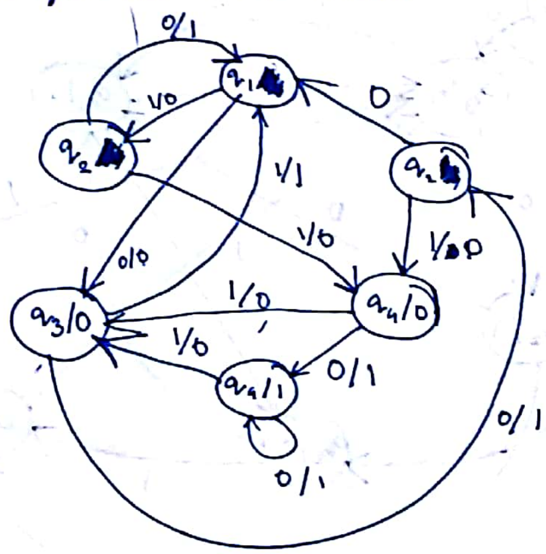
Corresponding Table for Moore Machine

Present State	Next State I/P - 0	Next State I/P - 1	O/P
q_0	q_3	$q_2/0$	0
q_1	q_3	$q_2/0$	1
$q_2/0$	q_1	$q_4/0$	0
$q_2/1$	q_1	$q_4/0$	1
q_3	$q_2/1$	q_1	0
$q_4/0$	$q_4/1$	q_3	0
$q_4/1$	$q_4/1$	q_3	1

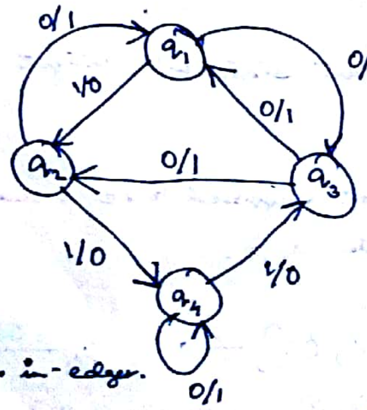
O/P → O/P of Present State

Moore → Mealy

Use the same diagram for Moore Model



Merging the states with same name —



→ Required Mealy Machine

- q_0 will never be included here, as it has only out-edges, no in-edges.

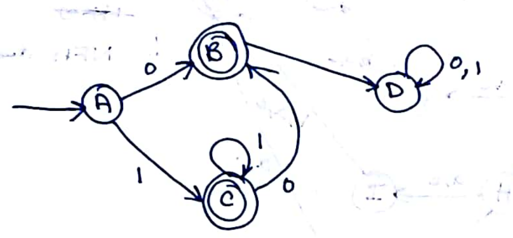
1. Give RE over $\{0,1\}$ with no leading 0's, i.e. 0, 10, 11 etc acceptable but not 01, λ, 00, and draw the DFA.

2. Draw DFA for all strings where the difference between the number of 0's and 1's is not divisible by 5.

3. $\Sigma = \{a,b,c\}$. Give NFA for strings which have a or c in atleast one of the last four positions.

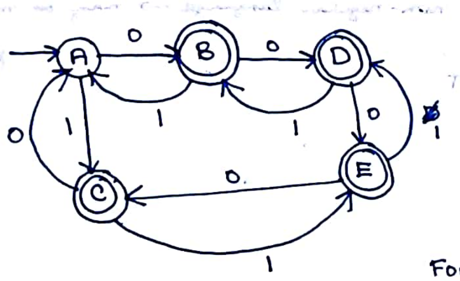
4. Prove that $((ab)^*(a^*b^*b^*)^*)^* = (a+b)^*$

1.



$RE = 1^* 0 \mid .1^+$

2.

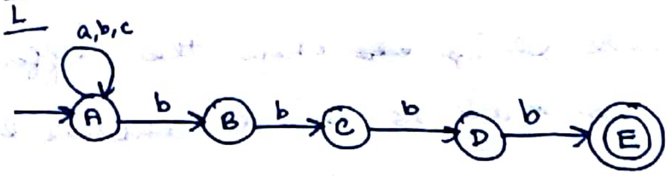


	for 0	for 1
rem 0	→ A	A
" 1	→ B	C
" 2	→ D	E
" 3	→ E	D
" 4	→ C	B

For 5, 5 remainders are possible. Hence there are 5 states.

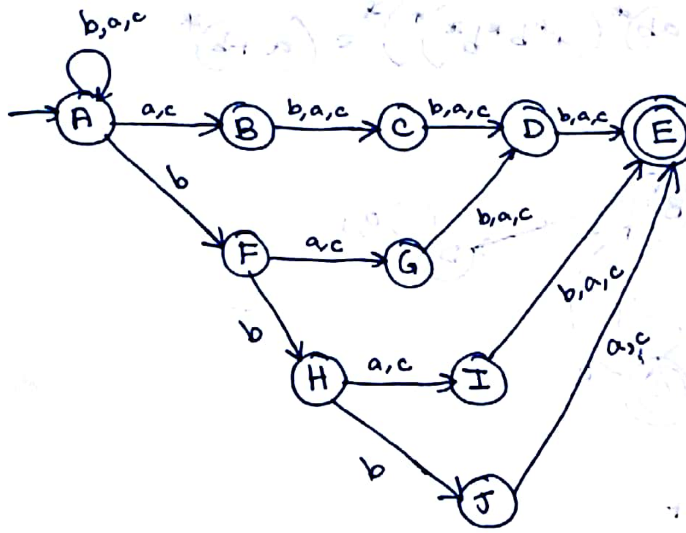
3.

$\overline{L}$



$\Downarrow$

$\overline{\overline{L}} = L \to X$



~~NFA~~

NFA due to state A.

<u>Pumping Lemma</u> → regular language → maintain pumping lemma

non-regular language → may or may not maintain pumping lemma.

- Read PPT

PPT

1) Prove that $L = \{a^p : p \text{ is a prime number}\}$ is NOT regular.

2) Let $C = \{w \mid w$ has an equal number of 0's and 1's$\}$.

3) Let $E = \{0^i 1^j \mid i > j\}$. Show ~~that~~ this is non-regular using the pumping lemma.

$$\overbrace{0000\ldots0}^{x}\,\boxed{0}_{y}\,\overbrace{111\ldots\ldots1}^{z}$$

$$\underbrace{0000\ldots0\,0}_{j+1}\;\underbrace{111\ldots\ldots1}_{j}$$

$|xy| \leq j+1$

$|y| \geq 1$

pump-down

$0000\ldots,0 \quad 1111\ldots\ldots1$

$|xy| \leq j+1$

$|y| \geq 1$

2) $L_1 = (n_0 = n_1)$

$L_2 = 0^* 1^*$

$\therefore L_1 \cap L_2$

$= \{0^n 1^n \mid n \geq 0\}$

$L_1 \cap L_2$ is regular.

$\therefore$ either L_1 or L_2 or both are not regular.

But L_2 is regular.

$\therefore L_1$ is not regular

Push-down automata (PDA)

- Use a Stack

$L = \{a^n b^n \mid n \geq 0\}$

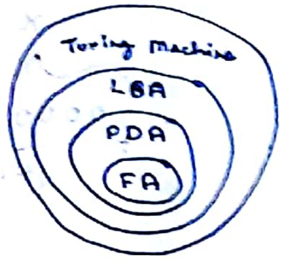
When the stack is empty and the string is ended as well, then it is valid

push()

pop()

$L = \{a^i b^j \mid i > j\}$

$L = \{a^i b^j \mid i < j\}$



Chomsky Classification

$\boxed{\alpha A B} \longrightarrow \boxed{cB\gamma}$

$B, \gamma, c\, \alpha \in (V, \Sigma)^*$
$A \in (V, \Sigma)^*$
$B \in (V, \Sigma)^*$
} No Restriction → Turing Machine

$|\alpha A \beta| < |cB\gamma|$ } LBA (Linear Bounded Automata)

$|\alpha A \beta| = 1$ } Context free Language → PDA

- DPDA → Deterministic Push-down Automata

Examples:

1) To check if parenthesization is balanced or not :

(())()

2) $0^n 1^n$

3) $\omega c \omega^R$

4) $\omega \omega^R$

5) $\omega \omega$; Ex: 00110011 → Not a Context free Language

3) $\Sigma = \{0, 1, c\}$

δ → transition functions

$\delta \rightarrow \{q_0, 001c100, \#\}$ → There is nothing in stack

$\{q_0, 01c100, p\#\}$

$\{q_0, 1c100, pp\#\}$

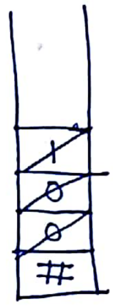
$\{q_0, c100, ppp\#\}$

Change the state, after this state we will pop. → $\{q_1, 100, ppp\#\}$

$\{q_1, 00, pp\#\}$

$\{q_1, 0, p\#\}$

$\{q_1, \varepsilon, \#\}$ → This state means that the string has been accepted



δ → At which state I am, my input, my stack top

$M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

- M → machine
- Q → set of states $\{q_0, q_1\}$
- Σ → alphabet of the input string $\{"(", ")"\}$
- Γ → stack symbol/contents $\{\#, p\}$
- δ → transition function
- q_0 → start state (q_0)
- Z_0 → start symbol of the stack $(\#)$
- F → Final states

$q_0 \in Q$

$Z_0 \in \Gamma$

- A PDA is stronger than FA.
- If we have a FA, then we can find the PDA also. But not the converse.

1. Write a grammar for balanced parentheses.

2. " " " " palindrome of odd length, even length.

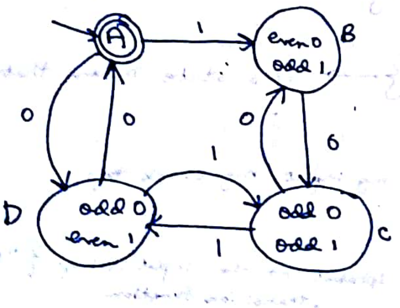
$\Sigma\{0,c,1\}^*$ $\Sigma\{0,1\}^*$

3. " " " " all strings containing even number of 0's and even number of 1's

4. " " " " $0^i 1^i \mid i \geq 0$

5. " " " " $L = \{w \mid n_a = n_b\}$

3.



4. $S \to 0S1 \mid \lambda$

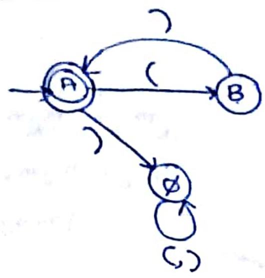
~~$G = \{\{0,1\},$~~

$G = \{\{S\},\{0,1\}, S \to 0S1 \mid \lambda, S\}$ (Ans)

- **Note:** If $0^i 1^i \mid i \geq 1$,

$G = \{\{S\},\{0,1\}, S \to 0S1 \mid 01, S\}$.

1.



• DFA not possible

$S \to (S)S \mid \lambda$

OR

$S \to SS \mid (S) \mid \lambda$

~~0110~~

0101|1010

2.

$\cancel{S \to 0 \mid S S^R \mid 0}$

wcw^R

ww^R

Even:

$S \to 0S0 \mid 1S1 \mid \lambda$

0110

10001

Odd:

$\cancel{S \to ScS}$

ScS

$S \to 0S0 \mid 1S1 \mid c$

5.

$S \to aSbS \mid bSaS \mid \lambda$

aSb

abSab

abbSaab

abb a

aSb

aSb

$a^m b^n c^m d^n \longrightarrow$ Not Possible

$a^m b^n c^n d^m \longrightarrow$ Possible

$| m,n \geq 1$

Grammer

$S \to aAd$

$A \to aAd \mid bBc$

$B \to bBc \mid \lambda$

aabcdd

aAd

aaAdd

aabBC

OR

$S \to \cancel{ASd}\ aSd \mid bBd$

$B \to bBc \mid bc$

aAd

abBcd

1) closed union $\longrightarrow L_1$ CFL
L_2 CFL

2) closed ~~con~~ concatenation $L_1 \cup L_2$ CFL

3) closed star closure $L_1 . L_2$ CFL

Write grammers for the following —

i) $L = \{ a^i b^j \mid i > j \}$

ii) $L = \{ a^n b^{2n} \mid n \geq 1 \}$

ii) $S \to aAbb \mid abb$

i) $S \to a \mid aS \mid aSb$

i) $i = j + k$

$a^{j+k} b^j$

$\underbrace{a^k}_{L_1} \; \underbrace{a^j b^j}_{L_2}$

$S \to \underline{S_1 S_2}$

$\cancel{S_1 \to \alpha S_1 \mid \alpha}$

$\cancel{S_2 \to aS_2 b}$

$S_1 \to aS_1 \mid a$

$S_2 \to aS_2 b \mid \lambda$

$\cancel{\alpha}$

$\underline{aab}$

$S \to S_1 S$

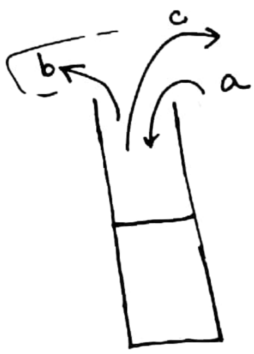
$\underline{aS}\; a\underline{S} b$

$aaaab$

$\cancel{S_1 S}$

- $a^i b^j c^k$

  $i + k = j$



Creating Push-Down Automata

$$M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$$

1. $L = \{0^n 1^n \mid n \geq 0\}$

transition function

① $\delta(q_0, 0, z_0) = (q_0, p z_0)$

$\Gamma = \{z_0, p\}$

② $\delta(q_0, 1, z_0) = \phi$ (reject)

③ $\delta(q_0, \lambda, z_0) =$ accept

④ $\delta(q_0, 0, p) = (q_0, pp)$

⑤ $\delta(q_0, 1, p) = (q_1, \varepsilon)$ → There is nothing (It signifies that we have popped)

⑥ $\delta(q_1, \lambda, z_0) =$ accept

⑦ $\delta(q_1, 0, p) =$ reject

⑧ $\delta(q_1, 1, p) = (q_1, \varepsilon)$

- Find if it is acceptable or not — 000111.

$\delta(q_0, 000111, z_0)$

$\delta(q_0, 00111, p z_0)$

$\delta(q_0, 0111, pp z_0)$

$\delta(q_0, 111, ppp z_0)$

$\delta(q_1, 11, pp z_0)$

$\delta(q_1, 1, p z_0)$

$\delta(q_1, \lambda, z_0)$

$\Downarrow$

accept

- Balanced Parenthesis.

$\Sigma = \{ "(", ")" \}$ $\quad Q = \{q_0\}$

$\Gamma = \{ z_0, P \}$

$\delta(q_0, (, z_0) = (q_0, Pz_0)$

$\delta(q_0,), z_0) = \phi$

$\delta(q_0, (, P) = \{q_0, PP\}$

$\delta(q_0,), P) = \{q_0, \varepsilon\}$

$\delta(q_0, \lambda, z_0) = \text{accept}$

- $L = \{ \omega\omega^R \mid \omega \in \{a,b\}^* \}$

$\Gamma = \{A, B, z_0\}$

$\delta(q_0, a, z_0) = (q_0, Az_0)$

$\delta(q_0, b, z_0) = (q_0, Bz_0)$

$\underbrace{baab}_{\omega}\underbrace{baab}_{\omega^R}$

$\underbrace{ba}_{\omega}\underbrace{ab}_{\omega^R}$

$\delta(q_0, a, A) = \{(q_0, AA), (q_1, \varepsilon)\}$

$\delta(q_0, b, A) = (q_0, BA)$

$\delta(q_0, a, B) = (q_0, AB)$

$\delta(q_0, b, B) = \{(q_0, BB), (q_1, \varepsilon)\}$

$\delta(q_1, \lambda, z_0) = \text{accept}$

$\delta(q_0, \lambda, z_0) = \text{accept}$

λ is accepted

Scanned by CamScanner

__Example:__

baab

$(q_0, baab, z_0)$

(q_0, aab, Bz_0)

(q_0, ab, ABz_0)

$(q_0, b, AABz_0)$ $\qquad$ (q_1, b, Bz_0)

$(q_0, \lambda, BAABz_0)$ $\qquad$ (q_1, λ, z_0)

$\therefore$ Non-deterministic.

1. $S \to a \mid \cancel{Aab}\ aAb \mid abSb$

   $A \to aAAb \mid bS$

   Find the parse tree for ababab for LMD (left most derivation) and RMD.

2. $S \to SS \mid aABb \mid a$

   $A \to aB$

   $B \to b$

   Derive the word a^3b^3a in LMD and RMD.

3. Write CFG for the following — (Write in 4 type form)

   a) strings end with a '0'

   b) " containing even number of 1's

   c) " that is not in the form 0^i1^j, $i,j>0$

d) {w | w contains at least three 1's}

e) {w | w starts and ends in same symbol,

f) {w | |w| is odd} $\Sigma = \{0,1\}^*$ }

g) {w | |w| is odd and the middle one is 0}

3. a) S → A0 | 0

A → 0A | 1A | 0 | 1

b) S → 0S | λ | S1S1S

S → S1 0S S1S |

c) S → A | B | C

A → 0A | 0

B → 1B | 1

C → D1 0D

D → 0D | 1D | 0 | 1

S → A | B | C

A → 0A | 0

B → 1B | 1

C → D1 0D

D → 0D | 1D | 0 | 1

d) S → A1A1A1

A → 0T | 1T | λ

~~100111~~

e) S → 1A1 | 0A0 | λ

A → 0A | 1A | λ

1A1

0A

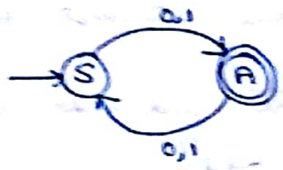
1A

1 0A0

1A1

- In CFG, the left hand side can't contain more than one variable.

f)



2

$S \to 0A \mid 1A$

$A \to 0S \mid 1S \mid \lambda$

g)

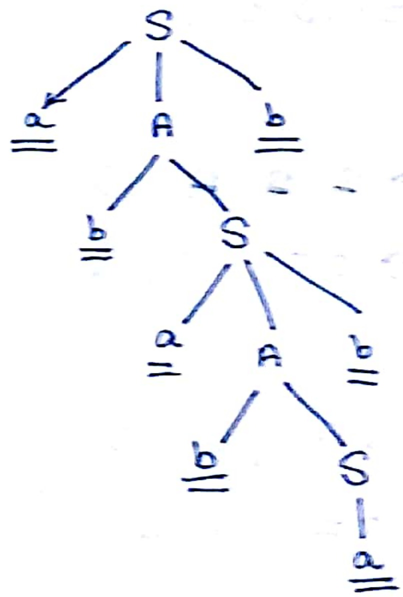
~~$S \to A\,0\,A$~~

~~$A \to 0B \mid 1B \mid \lambda$~~

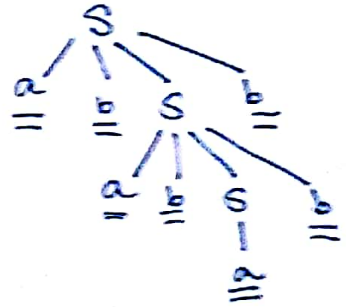
~~$B \to 0A \mid 1A \mid \lambda$~~

$S \to 0S1 \mid 0S0 \mid 1S0 \mid 1S1 \mid 0$

1. ababab b



- Inorder Traversal



∴ 2 parse Tree are possible.

Hence, this grammar is ambiguous

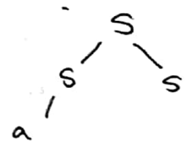
∴ There is only 1 Variable on the R.H.S of S. ∴ the LMD & RMD are same.

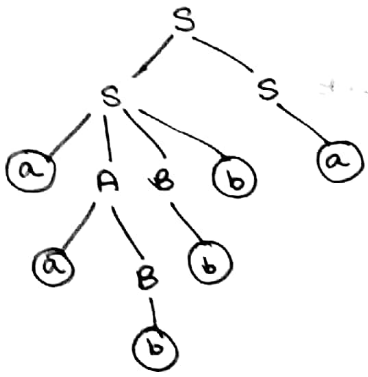
2. $\underline{a^2b^3a}$

$S \rightarrow SS \mid aABb \mid a$

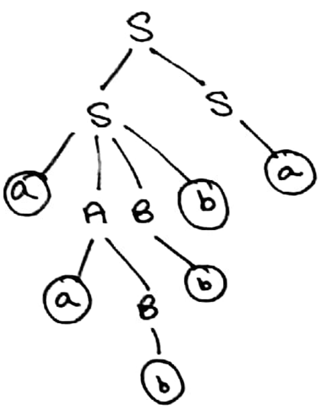
$A \rightarrow aB$

$B \rightarrow b$





LMD



RMD

∴ Same

If we start with ~~S~~ $S \rightarrow aABb$, then it not possible.

∴ the grammar is unambiguous

1. $S \to AB$
   $A \to a \mid \lambda$
   $B \to b$

2. $S \to AaB \mid aaB$
   $A \to \lambda$
   $B \to bbA \mid \lambda$

3. $S \to Aa \mid B$
   $B \to A \mid bb$
   $A \to a \mid bc \mid B$

4. $S \to aA$
   $A \to BB$
   $B \to aBb \mid \lambda$

5. $S \to ABaC$
   $A \to BC$
   $B \to b \mid \lambda$
   $C \to D \mid \lambda$
   $D \to d$

6. $S \to aA \mid aBB$
   $A \to aAA \mid \lambda$
   $B \to bB \mid bbC$
   $C \to B$

i) Useless symbols → Symbol which cannot be reached from the start state

ii) Null production

iii) Unit production → $A \rightarrow B$, $B \rightarrow \ldots$

$S \downarrow \times A$ $\quad$ $A \downarrow \times \Sigma^*$ (terminal)

Reachability Graph

i.

∴ A, B are reachable ~~to~~ from S.

∴ A, B are not useless.

- $S \rightarrow AB$

  $A \rightarrow cA$

  $C \rightarrow a$ ×

  $B \rightarrow b$

Here C is not reachable.

Hence, simply remove $C \rightarrow a$.

ii) Nullable Variable: Which produce λ directly or indirectly.

1. $S \rightarrow AB$, $A \rightarrow a \mid \lambda$, $B \rightarrow b$ $\Rightarrow$ $S \rightarrow AB \mid \cancel{B}b$, $A \rightarrow a$, $B \rightarrow b$

$\Downarrow$ $S \rightarrow B$ (Unit Production)

$S \rightarrow AB \mid b \mid a$

$A \rightarrow a$

$B \rightarrow b \mid a$