Computer Science

Theory of Computation

Regular Languages and Non-regular Languages

Lecture No.- 10

Mallesham Devasane Sir

Recap of Previous Lecture

Topic Closure Properties

Topics to be Covered

Topic — Pumping Lemma

Topic — FA with output

Pumping Lemma for Regular Languages

I) It satisfies every Regular language

II) It can be used to prove certain languages are non regular using contradiction.

III) It follows pigeon hole principle concept

Pumping Lemma

Proof for Regular Language.

If L is Regular language then

there exist pumping length (P) → minimum value of P depends on no. of states in min DFA $P \geq n(DFA)$

Such that $\forall w \in L$,

$|w| \geq P$, $\exists x, y, z \in \Sigma^*$, $xyz = w$

i) $|y| > 0$ $[y \neq \epsilon]$

ii) $|xy| \leq P$

iii) $\forall i \geq 0$ $xy^iz \in L$

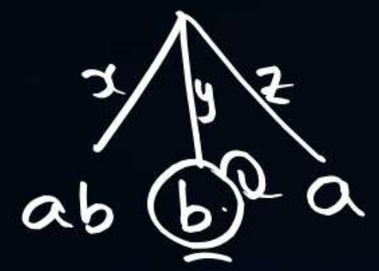
Pumping Lemma

Proof for Regular Language.

$L = ab(a+b)^*$

P=3

$w = abba$



$\forall i \; xy^iz \in L$

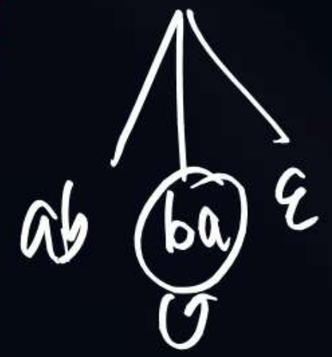
$ab.(b)^i a$

$i=0 \Rightarrow aba \in L$

$i=1 \Rightarrow abba \in L, \ldots$

P=4

$w = abba$



$ab.(ba)^i \varepsilon \in L$ ✓

P=5

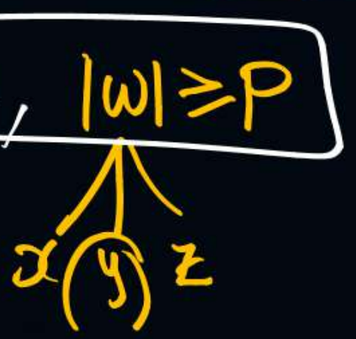
$w = abbbab$

$ab(b)^i bab \in L$

...

choose P → ≥ no. of states in min DFA exclude dead state

$\forall w \in L, \; |w| \geq P$



$|y| > 0$

$|xy| \leq P$

$\forall i \quad xy^iz \in L$

Pumping Lemma

Which of the following cannot be pumping length for $L = ab(a+b)^*$

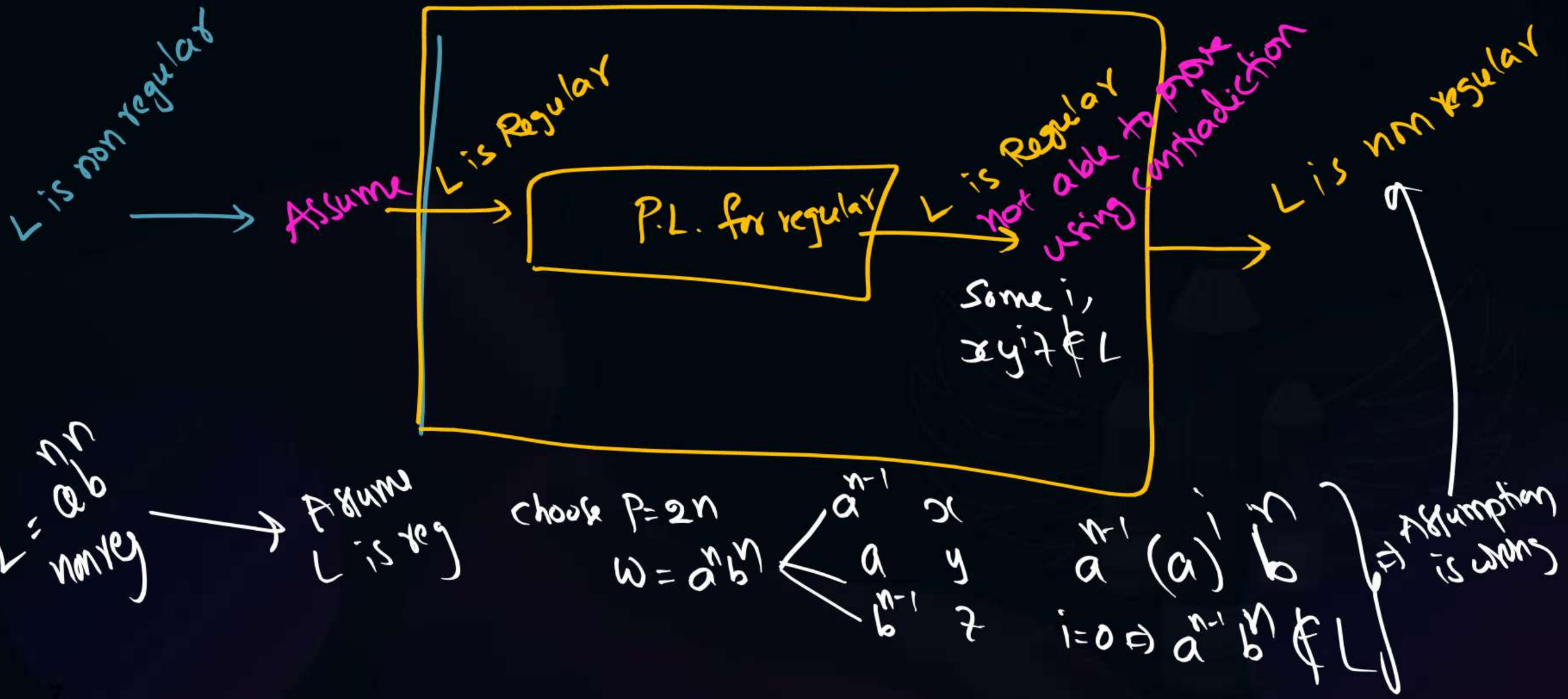
A) 2

B) 3

C) 50

D) 100

Pumping Lemma for Regular Languages



Pumping Lemma

Note: I) P.L. for regulars satisfies regular language

II) P.L. for regulars can prove nonregular using contradiction

III) Min Pumping length for regular $\geq$ no. of states min DFA

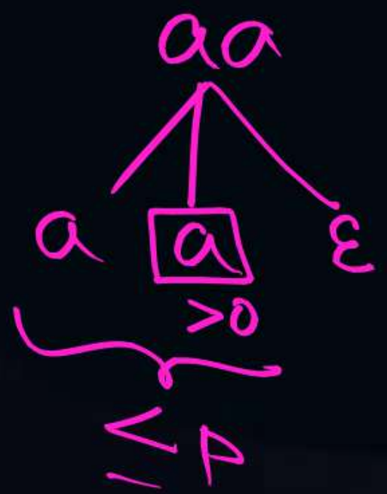
Pumping Lemma

$L = (a+b)^*$

$P = 2$

$P \geq 1$

$|w| \geq P$



$\forall i \; a(a)^i \varepsilon \in L$

$i=0 \Rightarrow a \in L$

$i=1 \Rightarrow aa \in L$

$i=2 \Rightarrow aaa \in L$

$\vdots$

FA with output

$L = a^n b^n$

$P = 2n$

$w = a^n b^n$

$i=0 \Rightarrow a^{n-1}(ab)^0 b^{n-1}$

$a^{n-1} b^{n-1} \in L$

$i=1 \Rightarrow a^{n-1}(ab)^1 b^{n-1}$

$a^n b^n \in L$

$i=2 \Rightarrow a^{n-1}(ab)^2 b^{n-1} \notin L$

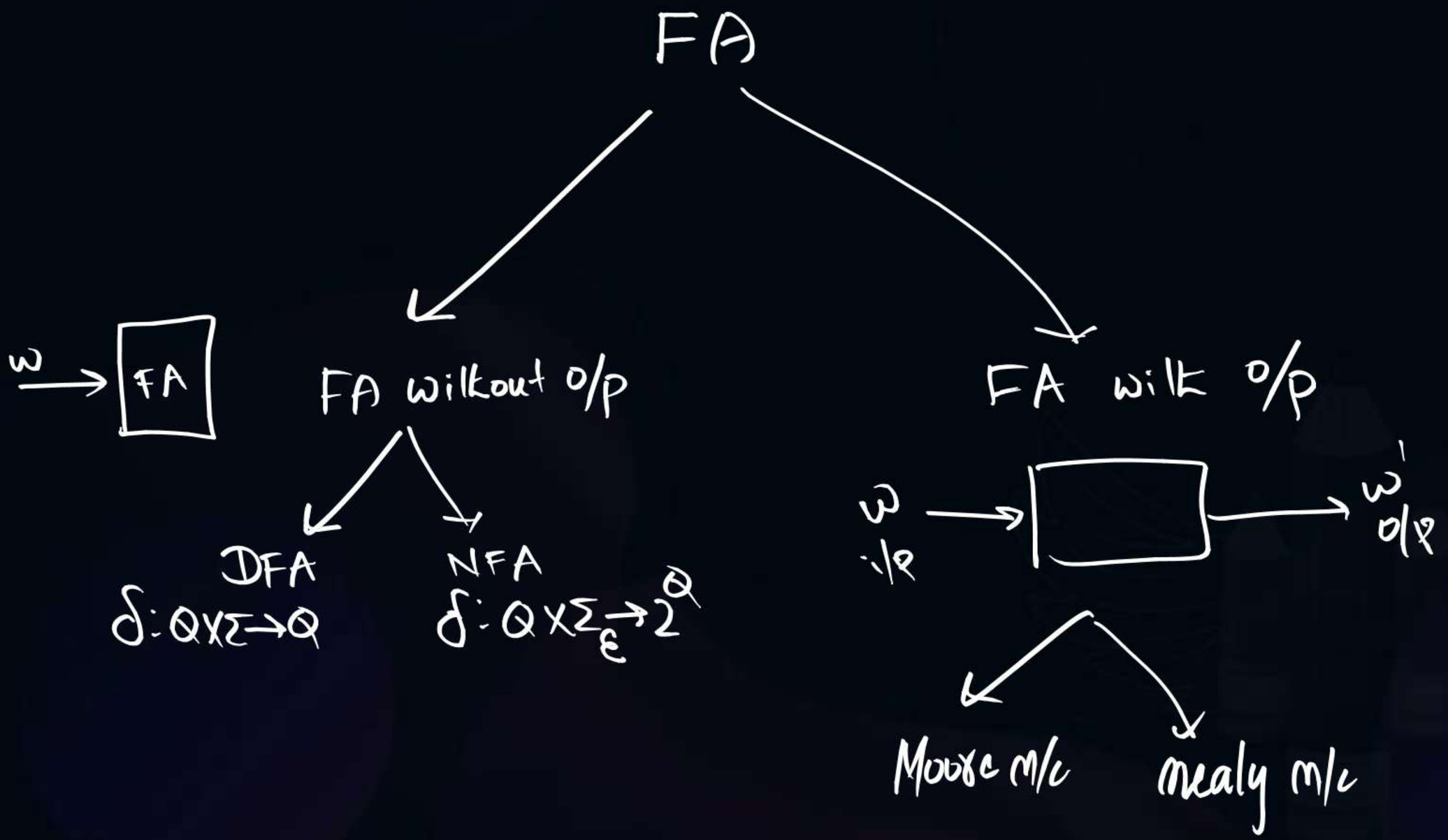
$i=3 \Rightarrow a^{n-1}(ab)^3 b^{n-1} \notin L$

$a^{n-1}\ ab\ b^{n-1}$

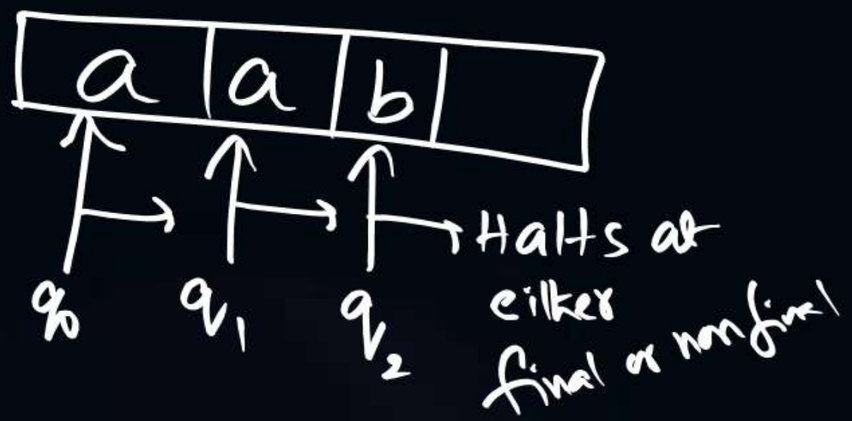
If $xy^i z \in L$ for which value of i, it proves non reg.

A) 0 B) 1 ~~C) 2~~ ~~D) 3~~

FA with output



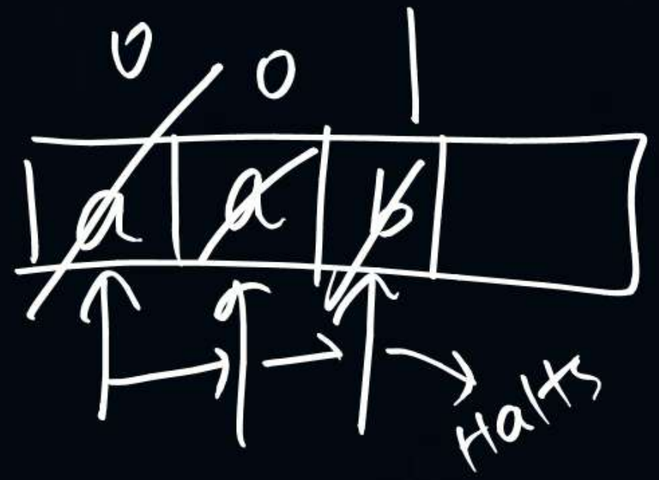
FA with output



$q_0 \xrightarrow{a} q_1 \xrightarrow{a} q_2 \xrightarrow{b} q_1$

q

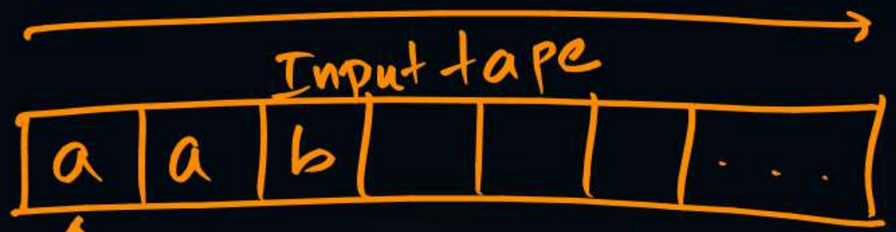
FA without o/p



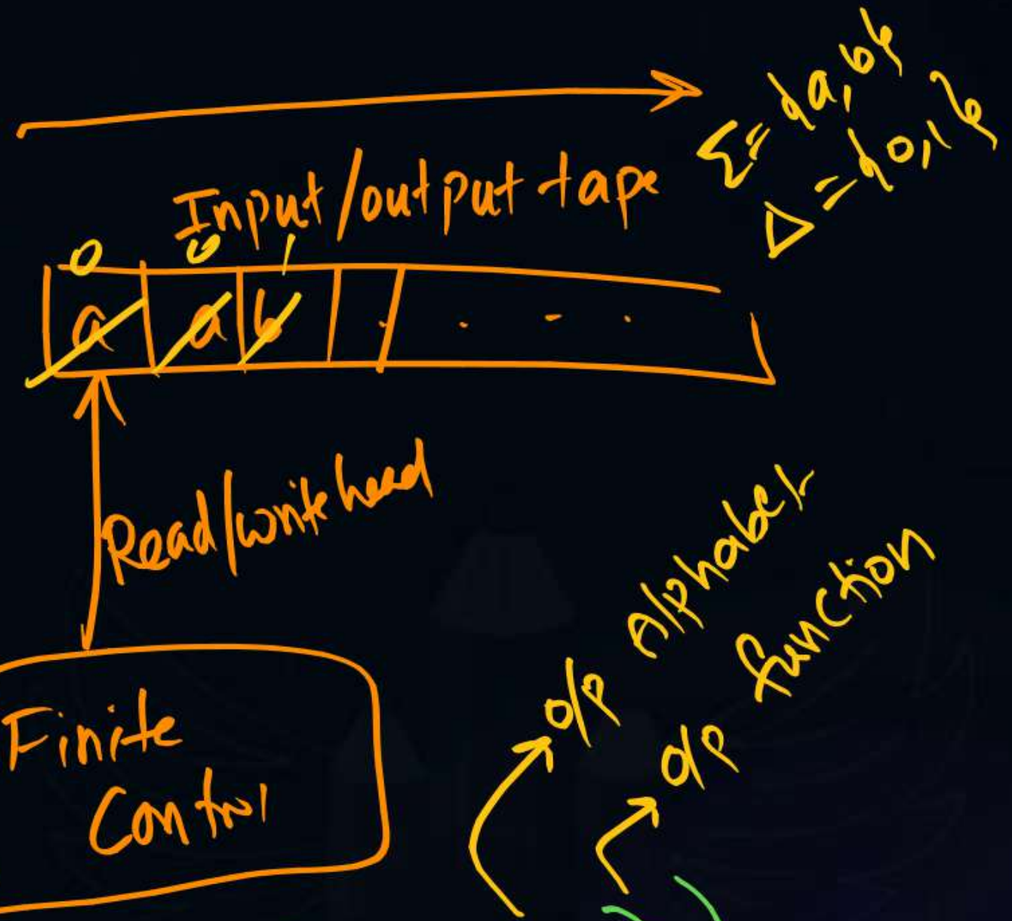
FA with o/p

FA with output

FA without O/P



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



$FA = (Q, \Sigma, \delta, q_0, \Delta, \lambda)$

$\delta : Q \times \Sigma \to Q$

FA with output

Moore M/C

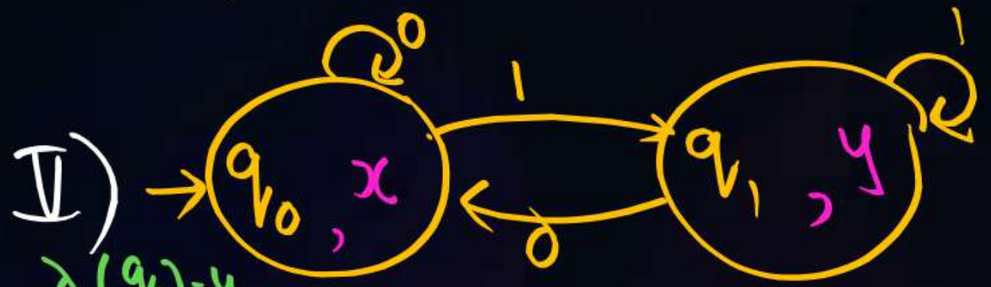
I) $\delta: Q \times \Sigma \to Q$

II) It is DFA

III) o/p is associated with state

IV) $\lambda: Q \to \Delta$

V)



$\lambda(q_1) = y$
$\lambda(q_0) = x$

$\Delta = \{x, y\}$

Mealy M/C

I) $\delta: Q \times \Sigma \to Q$

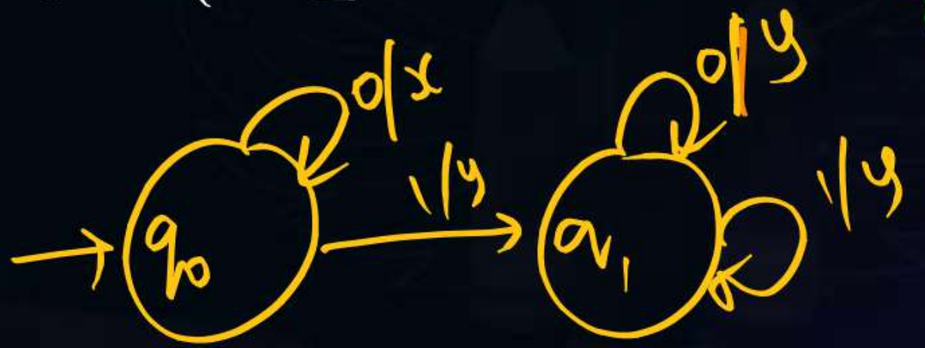
II) It is DFA

III) o/p is associated with transition

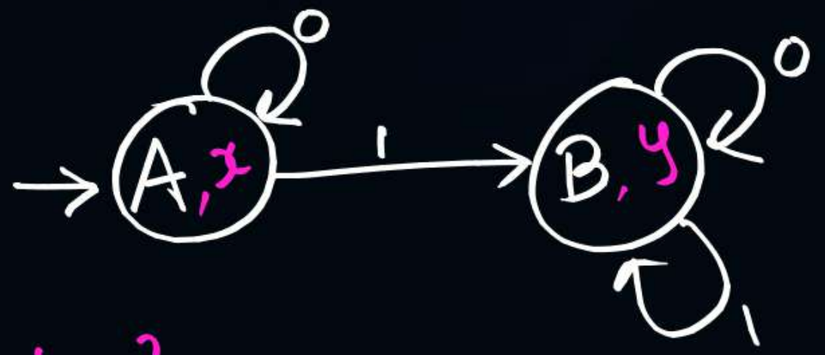
IV) $\lambda: Q \times \Sigma \to \Delta$

$\lambda(q_0, 0) = x$
$\lambda(q_1, 0) = y$
$\lambda(q_0, 1) = y$
$\lambda(q_1, 1) = y$

V)



FA with output



$\Sigma = \{0,1\}$

$\Delta = \{x, y\}$

$Q = \{A, B\}$

I/p = 01101 (5 len)

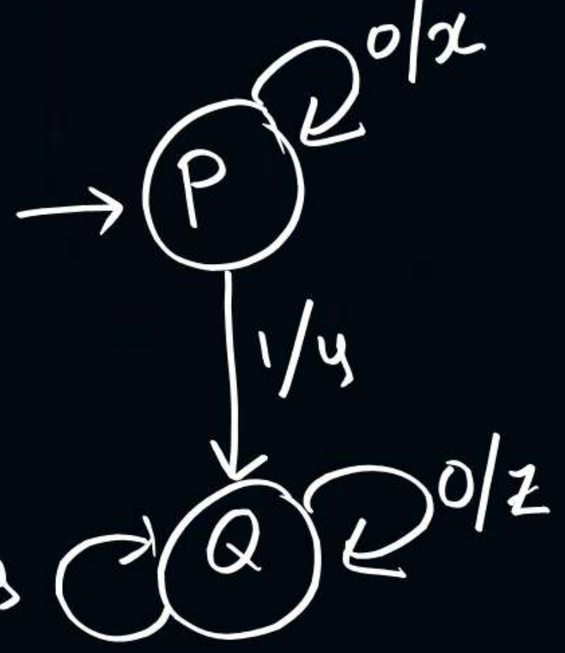
Path : $A \xrightarrow{0} A \xrightarrow{1} B \xrightarrow{1} B \xrightarrow{0} B \xrightarrow{1} B$

(A: x, A: x, B: y, B: y, B: y)

Default o/p

O/p = x x y y y y

(6 len; we will always ignore)



$\Sigma = \{0,1\}$

$\Delta = \{x, y, z\}$

I/p: 01101 (5 length)

Path : $P \xrightarrow{0/x} P \xrightarrow{1/y} Q \xrightarrow{1/y} Q \xrightarrow{0/z} Q \xrightarrow{1/y} Q$

O/p = x y y z y

(5 length)

Moore :

What should be o/p length ?

Assume for each i/p symbol, o/p length 1

If Input length = n

⇕

o/p length = n+1

Mealy :

What should be o/p length ?

If Input length = n

⇕

o/p length = n

2 mins Summary

Topic

- P.L. ✓
- FA with o/p.

THANK - YOU