CS & IT ENGINEERING

THEORY OF COMPUTATION

Mealy and Moore Machine

Lecture - 05

By- Venkat sir

Recap of Previous Lecture

Topic

Topic

{Closure properties of Regular Languages}

Topics to be Covered

Topic MEALY MACHINE

Topic ?? Moore Machine

Topic ??

Topic ??

Topic : Closure Properties of Language Families

Operation	Regular	DCFL	CFL	CSL	Recursive	RE
Union	yes	no	yes	yes	yes	yes
Intersection	yes	no	no	yes	yes	yes
Complement	yes	yes	no	yes	yes	no
Concatenation	yes	no	yes	yes	yes	yes
Homomorphism	yes	no	yes	no	no	yes
Substitution	yes	no	yes	yes	no	yes
Inverse Homomorphism	yes	yes	yes	yes	yes	yes
Reverse	yes	no	yes	yes	yes	yes
Intersection with a regular language	yes	yes	yes	yes	yes	yes

	Regular
① Subset op	✗
② Union op	✓
③ Concatenation	✓
④ Complement	✓
⑤ Kleene closure	✓
⑥ positive closure	✓
⑦ Reversal	✓

	Regular
⑧ Intersection op	✓
⑨ Difference op	✓
⑩ Prefix op	✓
⑪ Suffix op	✓
⑫ Quotient op	✓
⑬ Substitution op	✓
⑭ Homomorphism op	✓
⑮ Inverse Homomorphism	✓

L_1/L_2 = regular

Regular languages not closed Under

① Subset	X
② Infinite Union	X
③ Infinite Intersection	X

	Regular
① Subset op	✗
② Union op	✓
③ Concatenation	✓
④ Complement	✓
⑤ Kleene closure	✓
⑥ positive closure	✓
⑦ Reversal	✓

	Regular
⑧ Intersection op	✓
⑨ Difference op	✓
⑩ Prefix op	✓
⑪ Suffix op	✓
⑫ Quotient op	✓
⑬ Substitution op	✓
⑭ Homomorphism op	✓
⑮ Inverse Homomorphism	✓

L_1/L_2 = regular

$S(L)$ = always Regular.

If $L_1 \cap L_2$ is Regular, L_1 is Regular

then L_2 ?

$\{ab\} \cap \{a^n b^n\} \rightarrow \{ab\}$

L_1 reg, Non Reg., reg

(a) Always Regular

(b) need not be Regular

Home work

If L

(a) Always Regular

(b) Need not be Regular

Quotient of $L_1/L_2 = \{ x \mid xy \in L_1,\ y \in L_2 \}$

$$L_1/L_2 = \frac{0\cancel{0}}{\cancel{0}} = 0$$

$$\frac{x\cancel{y}}{\cancel{y}} = x \checkmark$$

$$\frac{00\cancel{11}}{\cancel{11}} = \underline{00}$$

$$\frac{L_1}{L_2} \quad \frac{000\cancel{01}}{\cancel{01}} = \{00\}$$

$$\frac{L_1}{L_2} \quad \frac{\epsilon\,\cancel{00}}{\cancel{00}} = \epsilon$$

$$L_1/L_2 = \frac{01}{10} = \phi$$

$$\frac{L_1}{L_2} \quad \frac{a^* b}{\boxed{b^* a}} = \phi$$

$$L_1/L_2 = \frac{a^*}{a} = \left\{ \cancel{\frac{\epsilon}{a}}, \frac{a}{a}, \frac{a^2}{a}, \frac{a^3}{a}, \frac{a^4}{a}, \frac{a^5}{a} \ldots \right\}$$

$$\epsilon, a, a^2, a^3, a^4 \ldots$$

$$L_1/L_2 = a^*$$

$$\underline{\underline{\Sigma^*}} = \frac{\Sigma^*}{\Sigma^*} = \frac{(a+b)^*}{(a+b)^*} = \underline{(a+b)^*} = \{\underline{\epsilon}_{\epsilon}, \underline{a}_{\epsilon}, \underline{b}_{\epsilon}, \underline{aa}_{\epsilon}, \underline{ab}_{\epsilon}, \underline{ba}_{\epsilon}, \underline{bb}_{\epsilon} \ldots\}$$

$$= (a+b)^* + \_ + \_ = \underline{\underline{(a+b)^*}}$$

$$\underline{\{\epsilon, a, b, aa \ldots\}}$$

$$\frac{\Sigma^+}{\Sigma^+} = \frac{(a+b)^+}{(a+b)^+} = (a+b)^*$$

$$\frac{\Sigma^+}{\Sigma^+} = \frac{\Sigma^*}{\Sigma^*} = \frac{\Sigma^*}{\Sigma^+} = \frac{\Sigma^+}{\Sigma^*} = \Sigma^* \checkmark$$

(Q) $\left.\begin{array}{l} L_1 = 0^*1 \\ L_2 = \boxed{1^*0} \end{array}\right\} = \phi$ $L_1/L_2 = ?$

(a) 0^*

(b) 1^*

(c) 0^*1^*

(d) ϕ ✓

(Q) $\boxed{L_1 = a^*ba^*}$ & $\boxed{L_2 = b^*a}$ What is $L_1/L_2 = ?$

ba

(a) $a^* + b^*$

(b) $a^*b^*a^*$

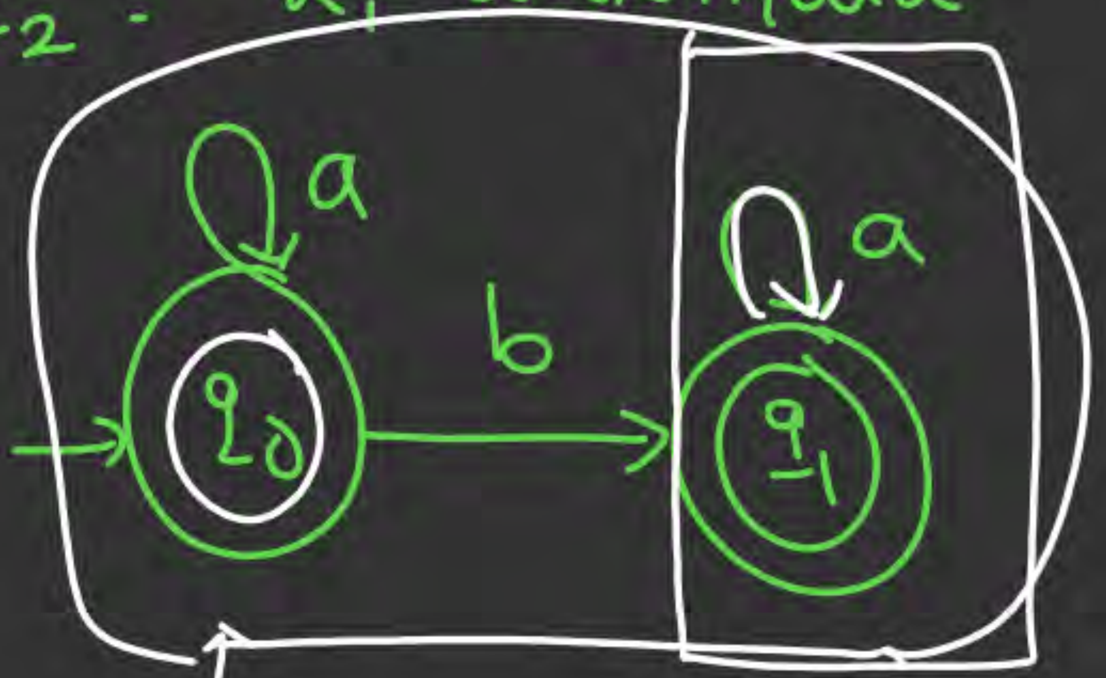
(c) $a^*ba^* + a^*$

(d) a^*b^*

$L_1/L_2 = L_1$ automata

accept

$L_1/L_2 =$



$\boxed{a^* + a^*ba^*}$

$L_1/L_2 = L_1$ finite Automata L_1/L_2 Finite Automata

1. Starting from first state Construct total automata if this automata accepts any string of L_2 then make first state as final.

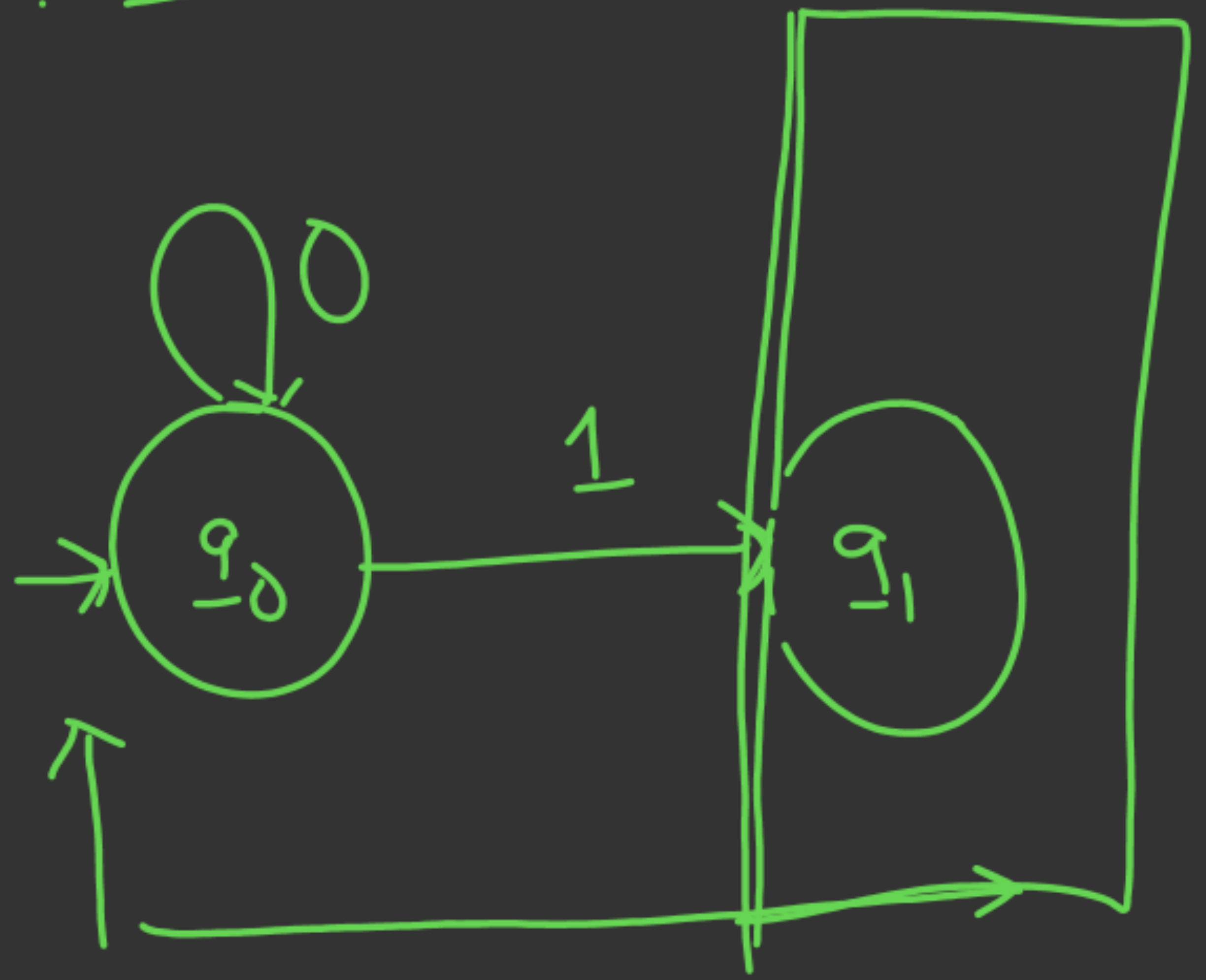
2. Starting from 2nd state Construct total automata if this automata accepts any string of L_2 then make 2nd state as final

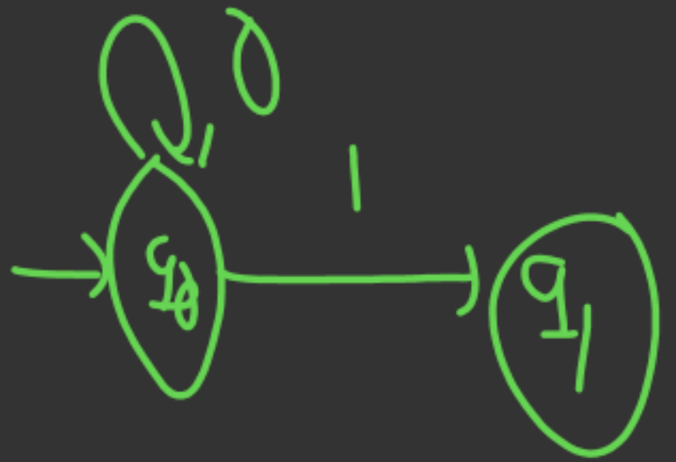
3. Repeat this process for every state to decide that state is final (or) not.

$$\frac{L_1}{L_2} = \frac{0^*1}{1^*0} = \phi$$

N.F

$L_1 / L_2 = 0^*1 =$





$= \phi$

Substitution op :-

Substitution is a mapping from $\Sigma \to \Delta$ where each symbol of Σ is Replaced by Regular Language over the alphabet Δ.

1. $S(\phi) = \phi$
2. $S(\epsilon) = \epsilon$
3. $S(a+b) = S(a) + S(b)$
4. $S(a \cdot b) = S(a) \cdot S(b)$
5. $S(a^*) = (S(a))^*$

(Q) $\boxed{L = 0^*}$

$$\boxed{S(0) = (a+b)(a+b)}$$

$S(0) =$ length exactly 2

✓ closed

$S(L) = ?\ [(a+b)(a+b)]^*$

length div by 2

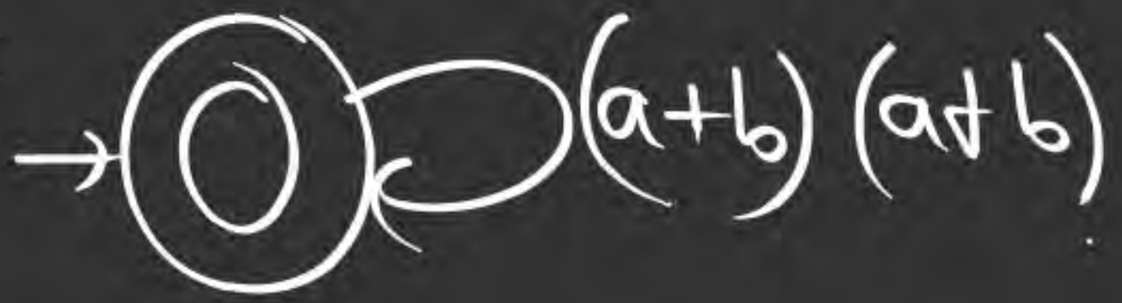
$S(0^*)$

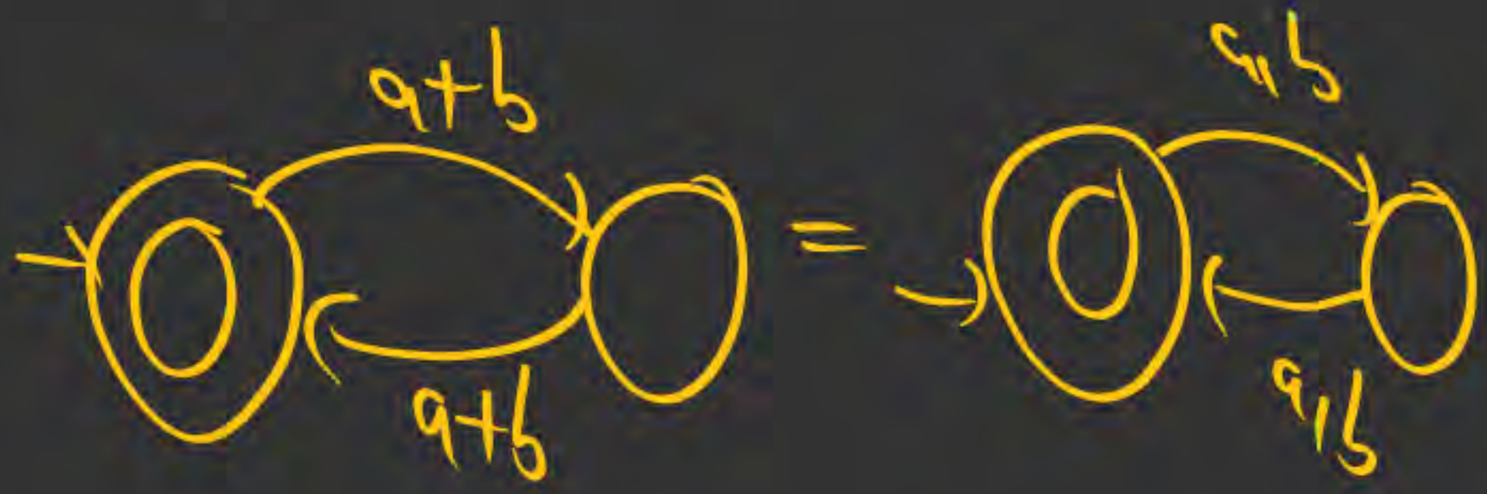
$[S(0)]^*$

$$[(a+b)(a+b)]^*$$

(1) $L = 0^*$ = 

regular

= 

$S(L) =$ = 

Always Regular

Homomorphism op :-

Homomorphism is a Special case Substitution where each symbol of Σ is Replaced by a Single string over the alphabet Δ.

(Q)

$\Sigma = \{0,1\}$ $\Delta = \{a,b\}$

closed

$h(0) = aa$

$h(1) = \epsilon$

$L_1 = 1^*01^*$

$h(L_1) = ?$

$h(L_1) = h(1^*01^*)$

$= h(1)^* h(0) h(1)^*$

$= \epsilon^* aa \epsilon^* = aa$

$h(L_1) = aa$

Inverse Homomorphism:

Applying homomorphism in reverse way is known as inverse homomorphism

(string is replaced by symbol)

$$h(0) = aa$$

(Q) $\Sigma = \{a, b, c\}$ $\Delta = \{0, 1\}$

$$\begin{cases} h(a) = 0 \\ h(b) = 1 \\ h(c) = 10 \end{cases}$$

$$\boxed{L = \overset{b\,a\,b\,a}{1010}_{\;c\;b\,a}}$$

✓ closed

$h^{-1}(L) = ?$

$h^{-1}(L) = \{cc, baba, cba, bac\}$

[MCQ]

L

$\Sigma^* - L$

#Q. Let R_1 and R_2 be regular sets defined over the alphabet then

regular

A $R_1 \cap R_2$ is not regular $\rightarrow$ false

regular

B $R_1 \cup R_2$ is not regular $\rightarrow$ false

C $\Sigma^* - R_1$ is regular $\rightarrow$ true

Complement

Regular

D R_1^* is not regular $\rightarrow$ false

regular

[MCQ]

regular a^* a^{10} regular b^* b^{30}

#Q. If $L_1 = \{a^n \mid n \geq 0\}$ and $L_2 = \{b^n \mid n \geq 0\}$, consider

I. $L_1 \cdot L_2$ is a regular language a^*b^* → true

II. $L_1 \cdot L_2 = \{a^n b^n \mid n \geq 0\}$ → false

Which one of the following is CORRECT?

$L_1 \cdot L_2$ $\{a^n b^n\}$

$L_1 \cdot L_2 = a^* b^*$

$\{a^n b^m\}$

A Only I

B Only II

C Both I and II

D Neither I nor II

Q

Consider the following two statements:

DFA

I. If all states of an NFA are accepting states then the language accepted by the NFA is Σ^*.

false

II. There exists a regular language A such that for all language B, $A \cap B$ is regular.

true

[2016-Set2: 2 Marks]

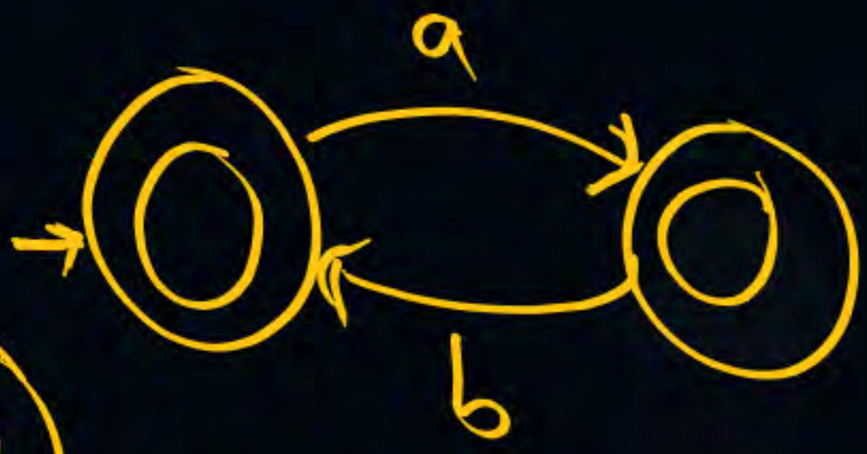
Which one of the following is CORRECT

(A) Only I is true

(B) Only II is true

(C) Both I and II are true

(D) Both I and II are false



$A \wedge B \rightarrow$ regular

$\{\epsilon\} \wedge \text{anything} \rightarrow \{\epsilon\} / \phi$

Q

Consider the following statements: $(a+b)^* \cup \{a^n b^n\} \rightarrow$ regular

$(a+b)^*$

I. false If $L_1 \cup L_2$ is regular, then both L_1 and L_2 must be regular.

II. The class of regular languages is closed under infinite union. } false

Which of the above statements is/are TRUE?

[2020: 1 Mark]

A. Neither I nor II

B. II only

C. I only

D. Both I and II

~~not closed~~

Infinite Union

$\{L_1, L_2, L_3, L_4 \ldots\}$ regular

$L_1 \cup L_2 \cup L_3 \cup L_4 \cup L_5 \cdots = $ may (or) may not regular

$$\underline{\{ab\}} \cup \underline{\{a^2b^2\}} \cup \underline{\{a^3b^3\}} \cup \underline{\{a^4b^4\}} \cup \cdots = \underline{\{a^nb^n \mid n \geq 1\}}$$

non regular ✓

Regular languages closed under finite Union

not closed under infinite Union

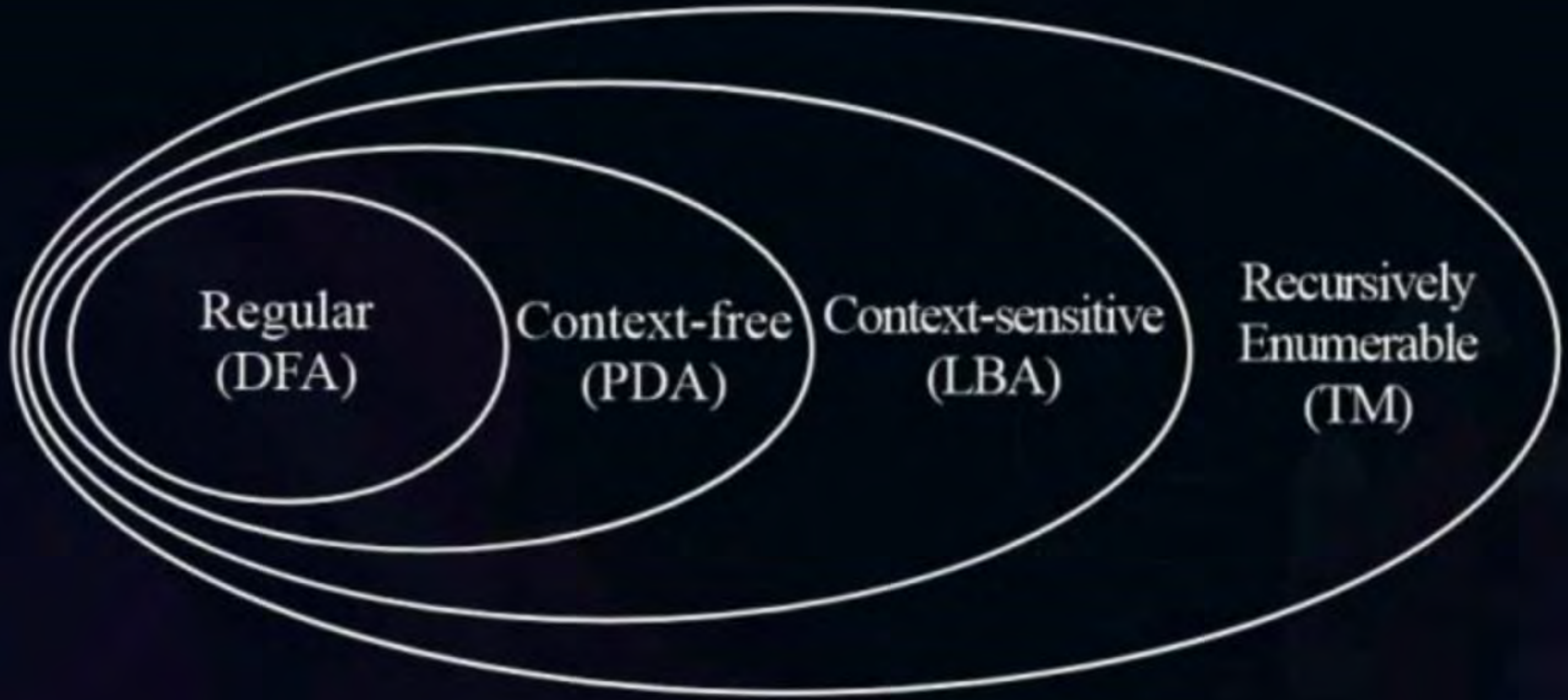
Infinite Intersection op (not closed)

$$L_1 \cap L_2 = \overline{L_1^c \cup L_2^c}$$

$$L_1 \cap L_2 \cap L_3 \cap L_4 \cap \cdots = \overline{L_1^c \cup L_2^c \cup L_3^c \cup L_4^c \cdots}$$

Infinite Union

Topic : Regular expression



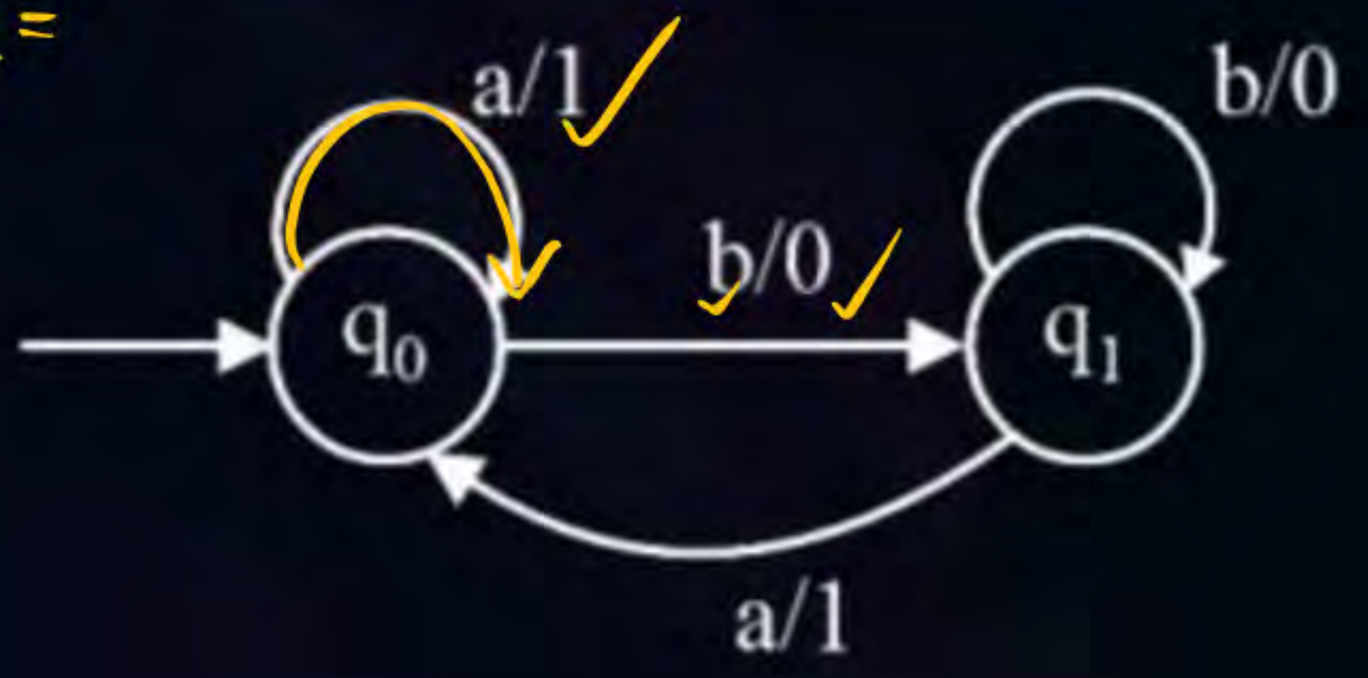
Circuit Design

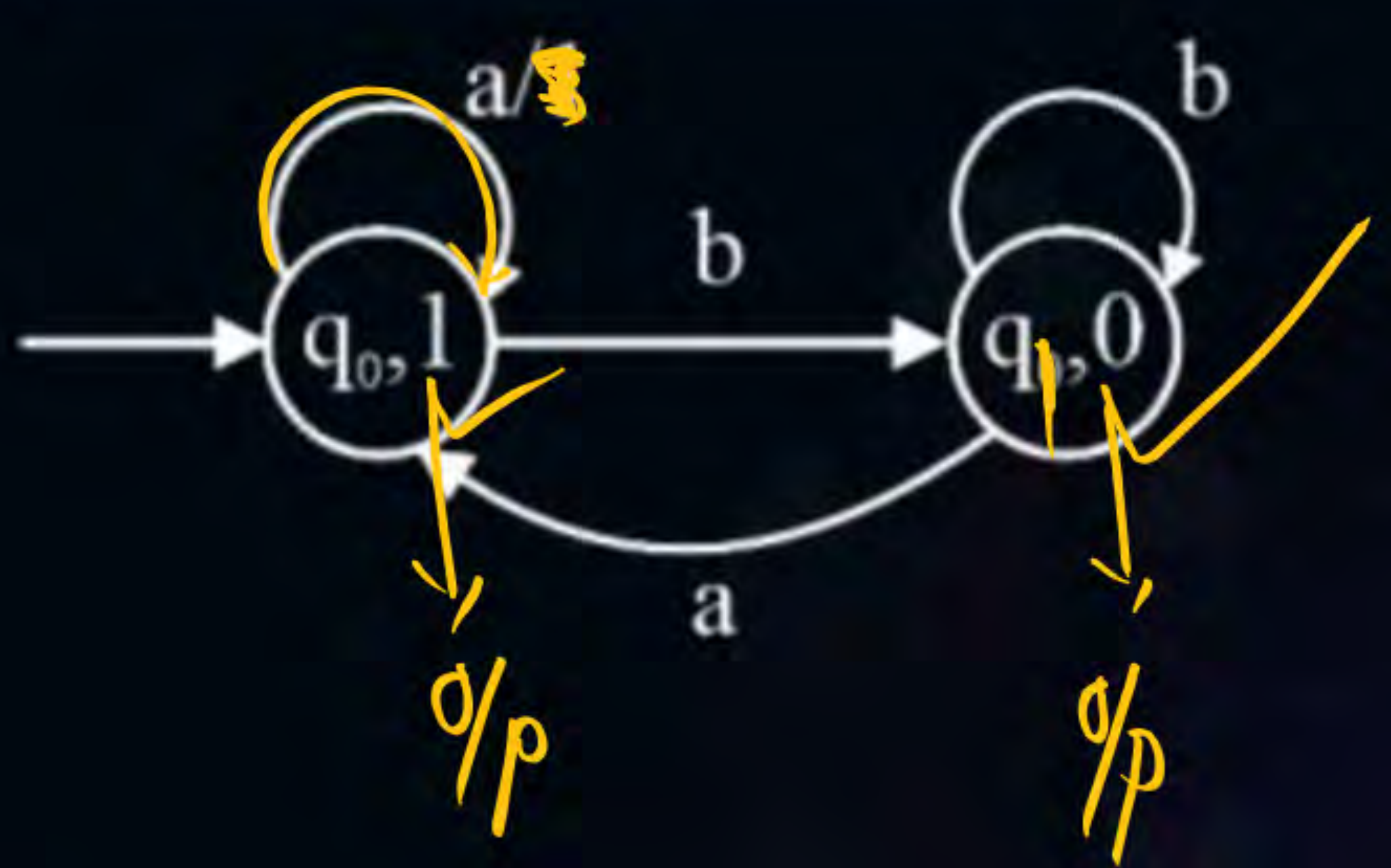
0101 → F.A with output → 1010

Mealy Machine

Moore Machine

a =





- Mealy Machine:
- It is a mathematical model in which output is associated
- with transition.

- Moore Machine:
- It is a mathematical model in which output is associated
- with state.

Formal Definition

$$\boxed{(Q, \Sigma, q_0, \delta, \Delta, \lambda)} \Bigg\} \text{Mealy \& Moore}$$

- Q : finite no. of states
- Σ : input alphabet
- q_0 : initial state
- δ : transition function: $\boxed{Q \times \Sigma \to Q}$
- Δ : output alphabet
- λ : output function

Mealy : $\lambda : Q \times \Sigma \to \Delta$

Moore : $\lambda : Q \to \Delta$

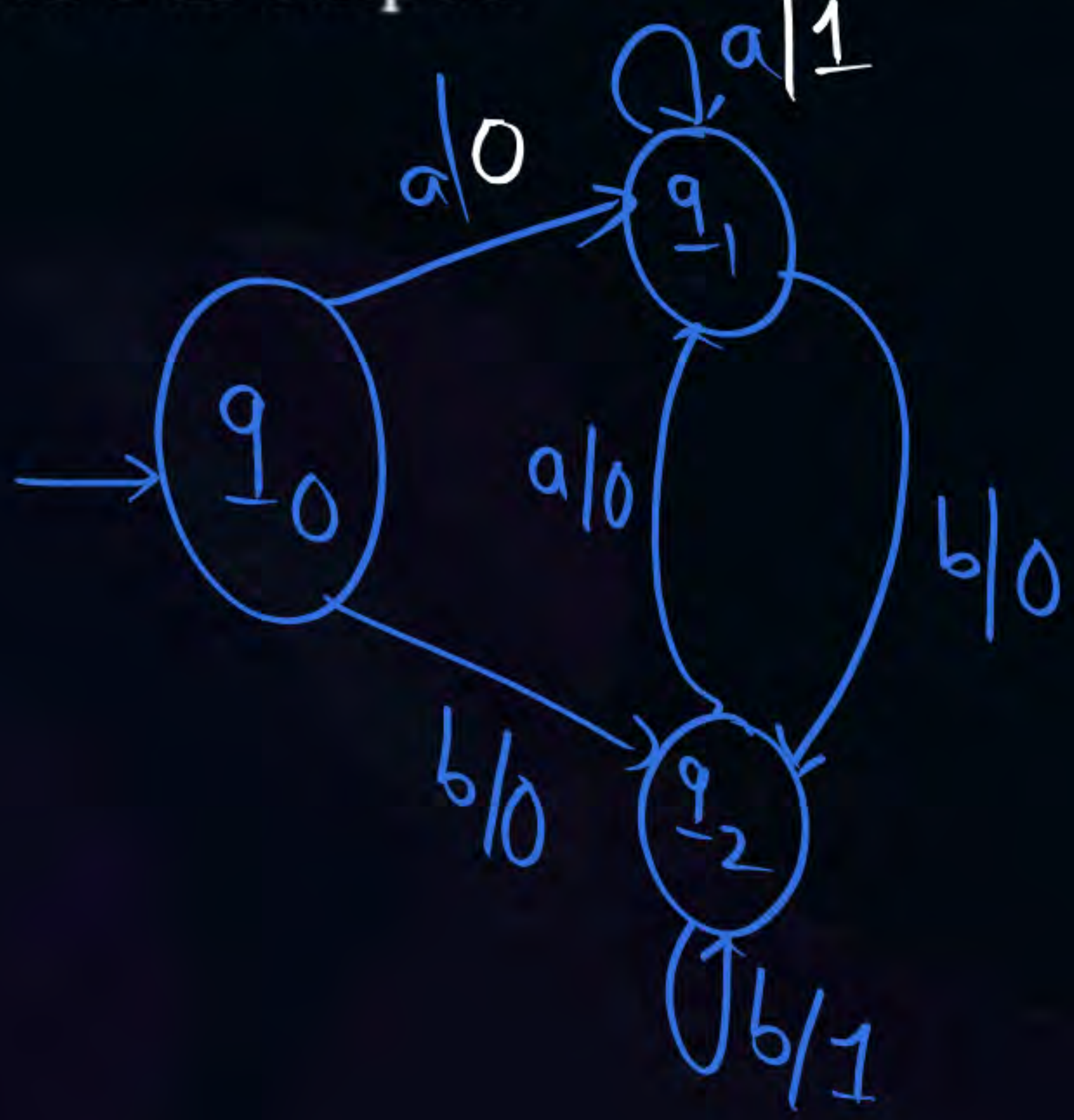
[NAT]

$Q \times \Sigma \rightarrow Q$

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

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

#Q. Construct mealy machine that takes all strings of a's and b's as input and produces 1 as output if last two symbols in the input are same otherwise produces 0 as output.



aaaa → 1

a, b, ab } → 0

bbbaa

aaaabb

0

[NAT]

Home Work

#Q. Construct mealy machine that takes all strings of a's and b's as input and produces 1 as output if last two symbols in the input are different otherwise produces 0 as output.

[NAT]

Home Work

#Q. Construct mealy machine that takes all strings of 0's and 1's as input and produces A as output if input ending with 10 or produces B as output if input ending with 11 otherwise produces output C.

[NAT]

Home Work

#Q. Construct mealy machine that produces 1's complement of given binary number as output.

[NAT]

#Q. Construct mealy machine that produces 2's complement of given binary number as output.(assume we are reading string from LSB to MSB)

[NAT]

#Q. Construct Moore machine that takes all binary strings as input and produces Residue modulo 4 as output.

[NAT]

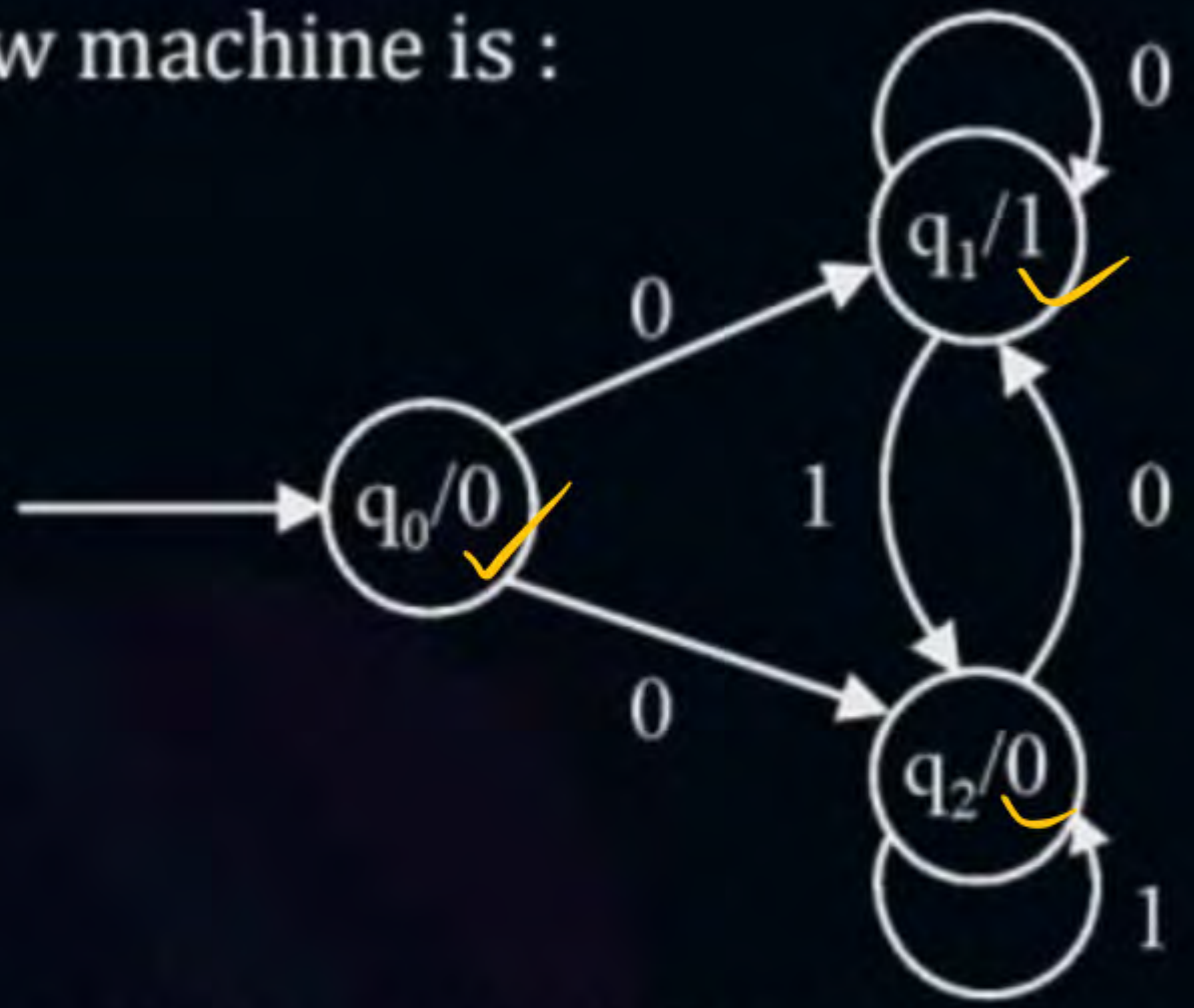
#Q. Construct Moore machine that takes all binary strings as input and produces Residue modulo 5 as output.

[NAT]

#Q. Construct Moore machine that takes all base 3 numbers as input and produces Residue modulo 4 as output.

[MCQ]

#Q. The below machine is :



A. A Mealy machine to find 2's complement of a number

B. A Moore machine to find 2's complement of a number

C. A Mealy machine to find 1's complement of a number

D. A Moore machine to find 1's complement of a number

[MCQ]

#Q. A finite state machine with the following state table has a single input x and a single output z.

	Present state x = 1	Next state, z x = 0
A	D, 0	B, 0
B	B, 1	C, 1
C	B, 2	D, 1
D	B, 1	C, 0

If the initial state is unknown , then the shortest input sequence to reach the final state C is :

A. 0 1

B. 10

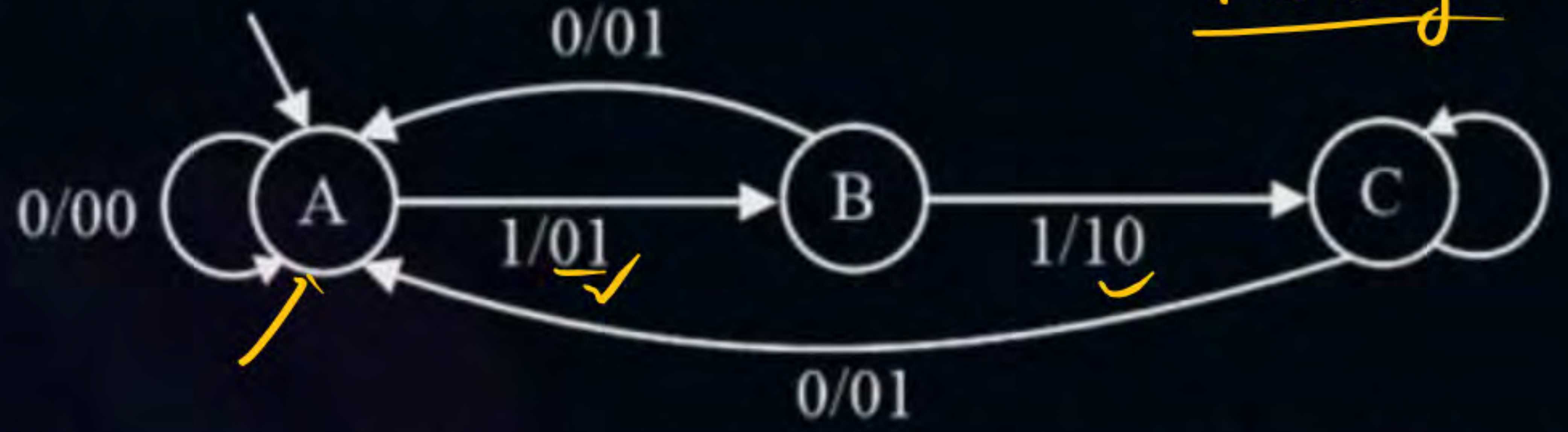
C. 101

D. 110

[MCQ]

#Q. The Finite state machine described by the following state diagram with A as starting state, where an arc label is x/y and x stands for 1-bit input and y stands for 2-bit output.

Mealy



A. Outputs the sum of the present and the previous bits of the input

B. Outputs a "01" whenever the input sequence contain "11"

C. Outputs a “00” whenever the input sequence contains "10"

D. None of the above

$$\delta(0) = a^{*} \quad (\underline{\text{language}})$$

$$H(0) = \underline{a9} \quad (\text{string})$$

2 mins Summary

Topic One

Topic Two

Topic Three

Topic Four

Topic Five

THANK - YOU