CS & IT ENGINEERING

Theory of Computation

NFA

Lecture No.- 01

By- Venkat sir

Recap of Previous Lecture

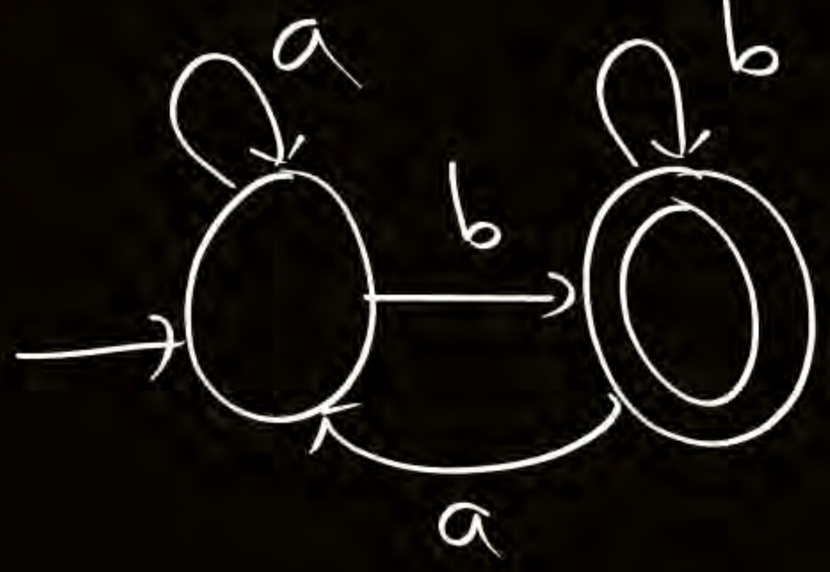
Topic — Length Condition DFA

Topic — (AND) / (OR) × product DFA

Topics to be Covered

- **Topic** Finite Automaton & Regular Languages.
- **Topic** Pushdown Automata & Context free Languages.
- **Topic** Turing Machine & Recursive Enumerable Languages.
- **Topic** Undecidability.

$$\begin{pmatrix} a\cdot S \\ b\cdot S \end{pmatrix} aba$$



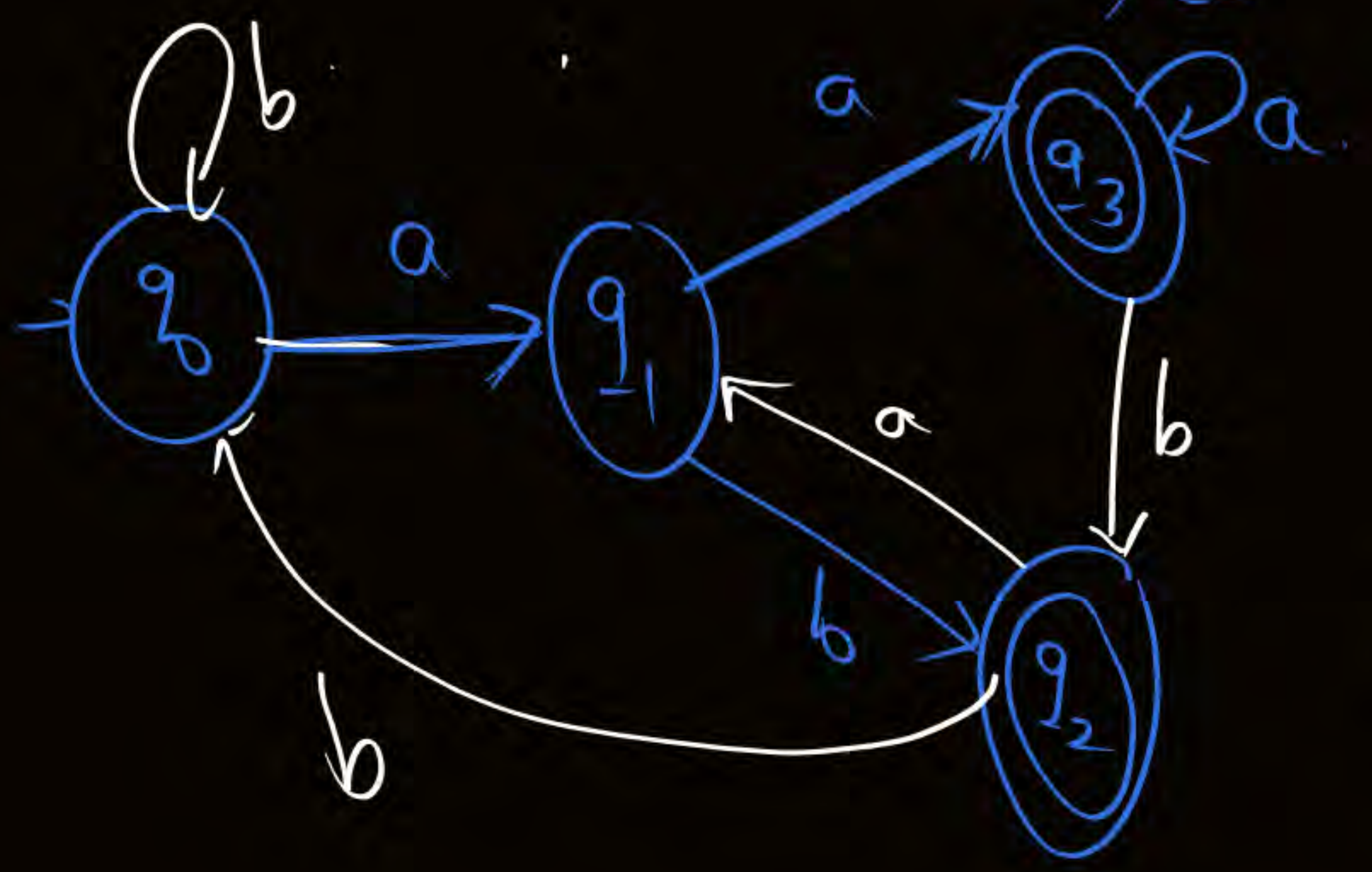
$$\left\{ \begin{array}{l} \begin{pmatrix} aS \\ bS \end{pmatrix} a \\ (\ )\, ab \\ (\ )\, aba \\ (\ )\, abab \end{array} \right\} \text{DFA}$$

(a.s b.s) a a|b

(Q) Construct min DFA that accepts all strings where Second input symbol is (a) from end

[Second symbol is a reading from R.H.S]

a|b



(a.s b.s) a a|b

(a.s b.s) a → a, → b

(a.s b.s) (aa) (ab)

aa

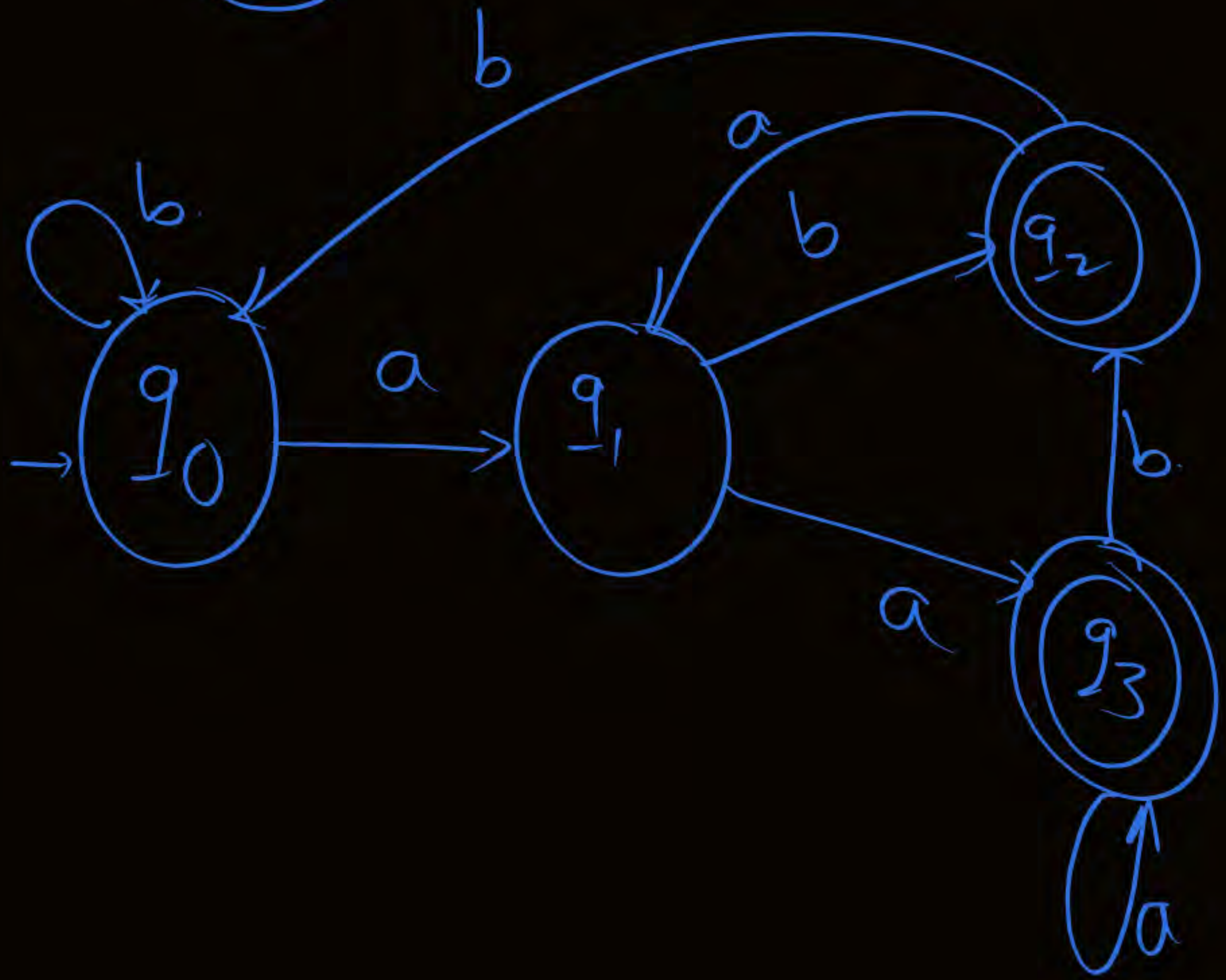
ab

abab

aaaabb

abbaa

ab



$2^{nd} \to 4$

end

(Q) Third input symbol is a from end of the input

(any a's b's) a (a|b)(a|b)

as, bs

aaa
aab
aba
abb

from end

$2 \to 4$

$3 \to 8$

$4 \to 16$

⋮

2^n states

$2^n = 16$

8th input from end is a

$2^8 \to 256$

DFA → 10th from end → $2^{10} \to 1024$ 1 hr

NFA → " " → 11 → $\frac{1}{2}$ min

from Left sid

2nd input is a $\rightarrow$ 4

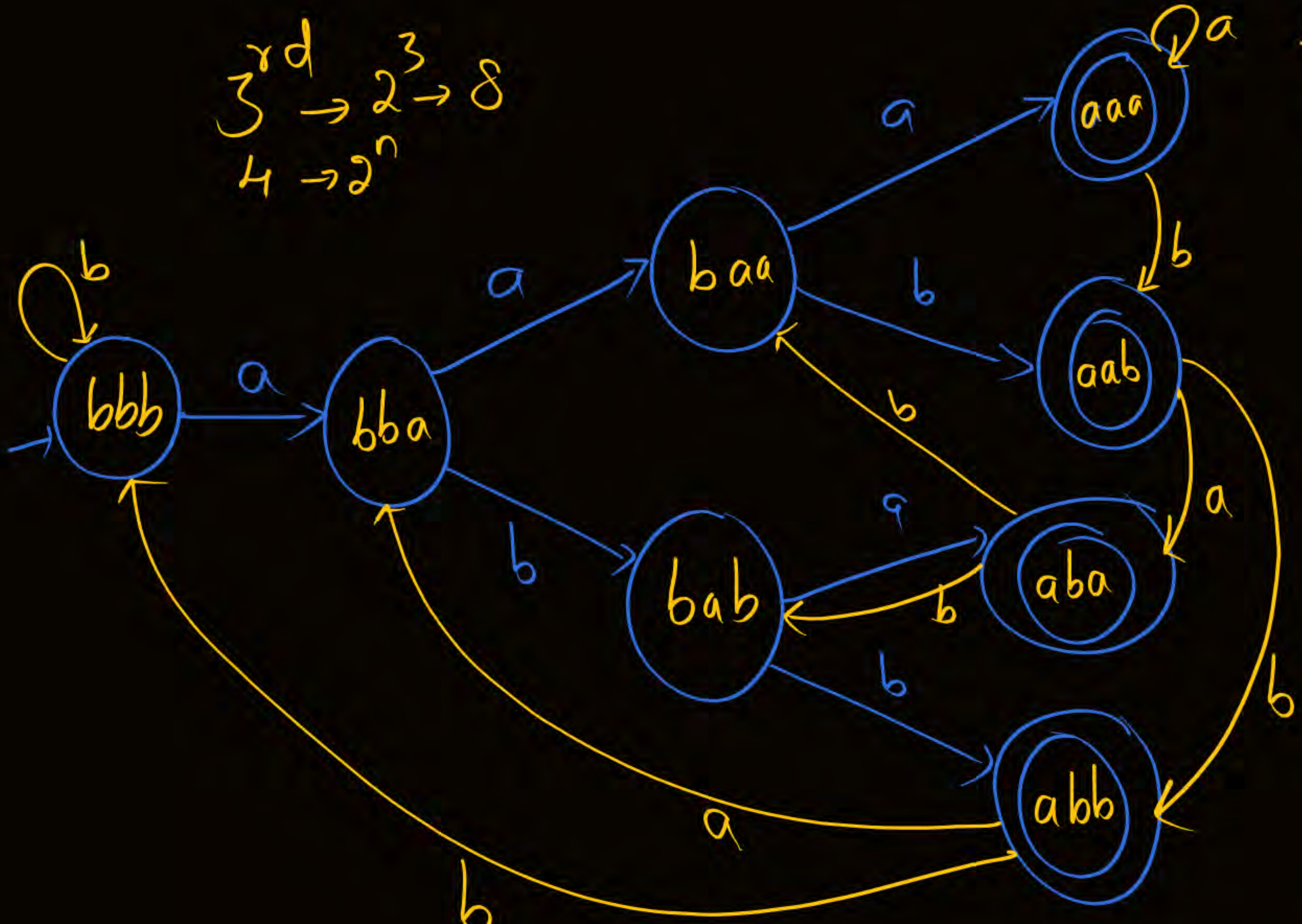
3rd " " " $\rightarrow$ 5

5th " " " $\rightarrow$ 7

nth " " " $\rightarrow$ $n+2$

$3^{rd} \rightarrow 2^3 \rightarrow 8$

$4 \rightarrow 2^n$



aaaa

aaab

aaba

aabb

abaa

abab

abba

abbb

(Q) 5^{th} input symbol is a from end

$$\underline{\quad\quad}\ \ \underline{a}\ \ \underline{a/b}\ \ \underline{a/b}\ \ \underline{a/b}\ \ \underline{a/b}$$

$\uparrow$

Topic : Deterministic Finite Automata

FORMAL DFA :

DFA is defined as

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

Q : Finite set of states

Σ : Input alphabet

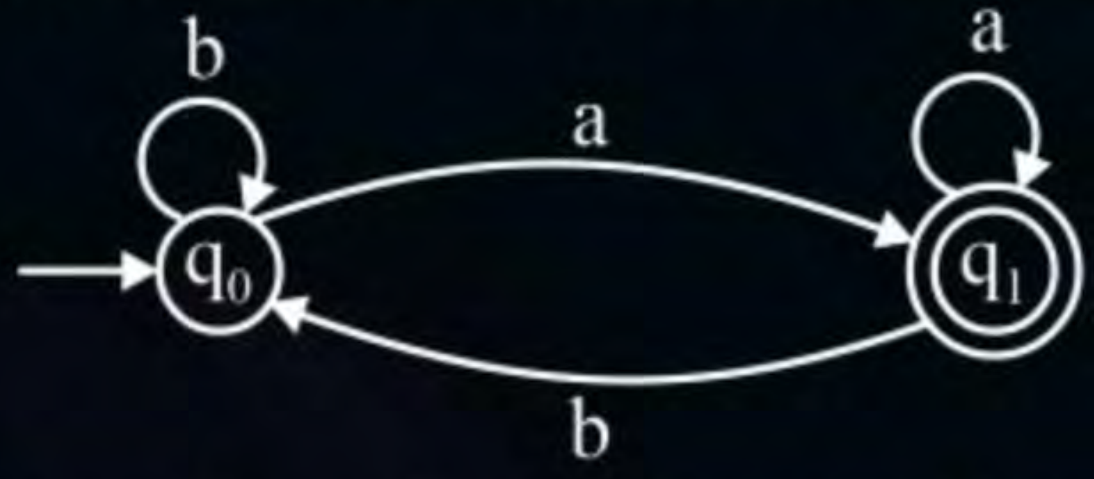
q_0 : Initial state

F : Set of final states

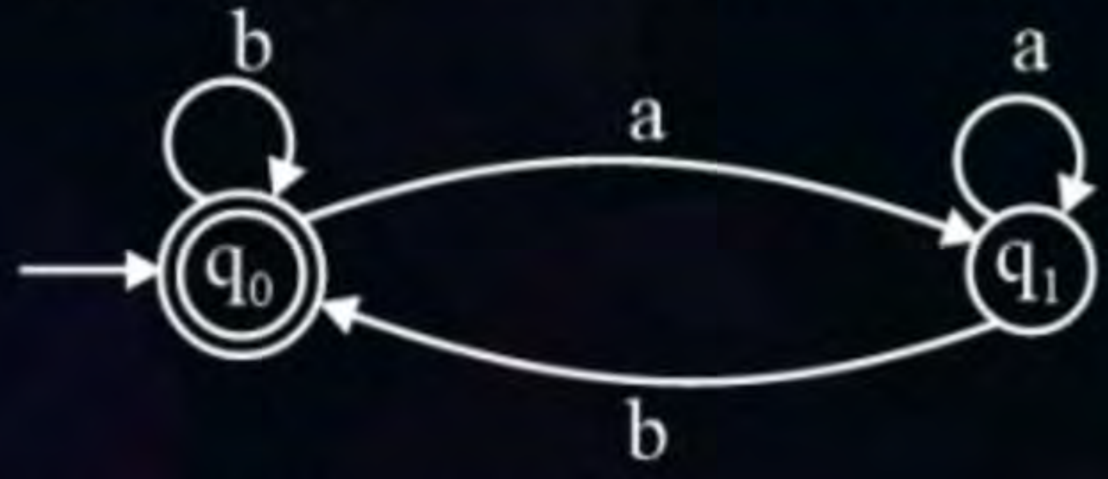
δ : Transition function $Q * \Sigma \rightarrow Q$

Topic : Complement of DFA

By interchanging final and non final states we can convert into complement DFA.



Set of all strings ending with a after complement

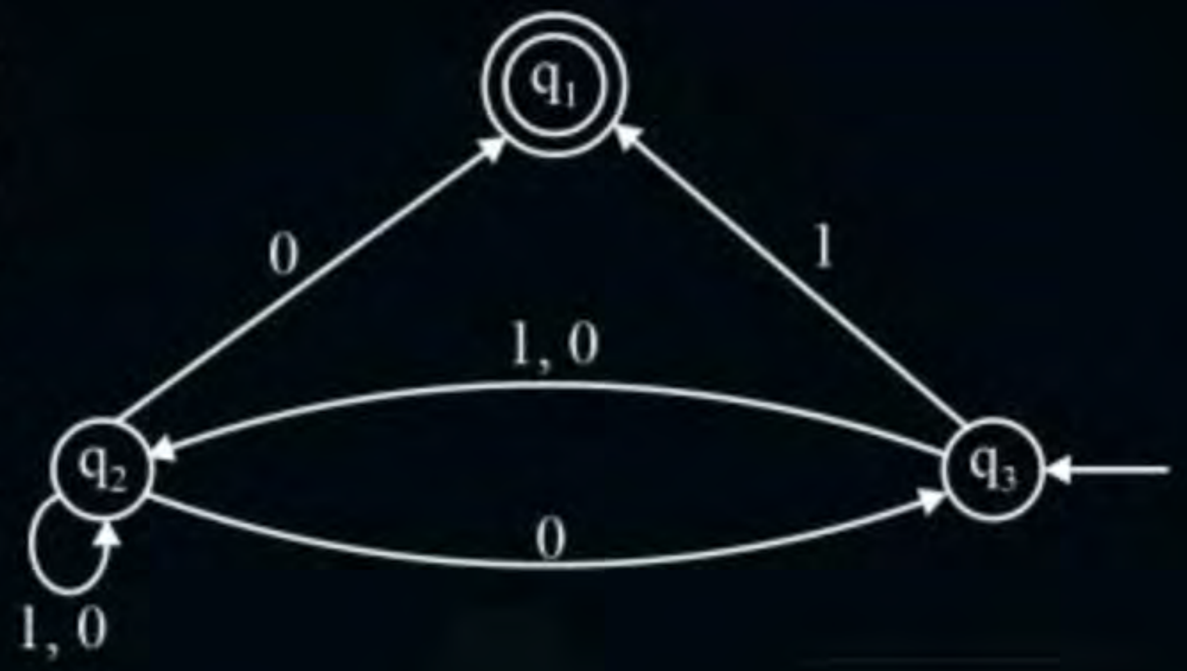


Set of all string .... Ending with

MCQ

#Q. Consider the NFA M shown below.

Let the language accepted by M be L. Let L_1 be the language accepted by the NFA M_1, obtained by changing the accepting state of M to a non-accepting state and by changing the non-accepting state of M to accepting states. Which of the following statements is true?



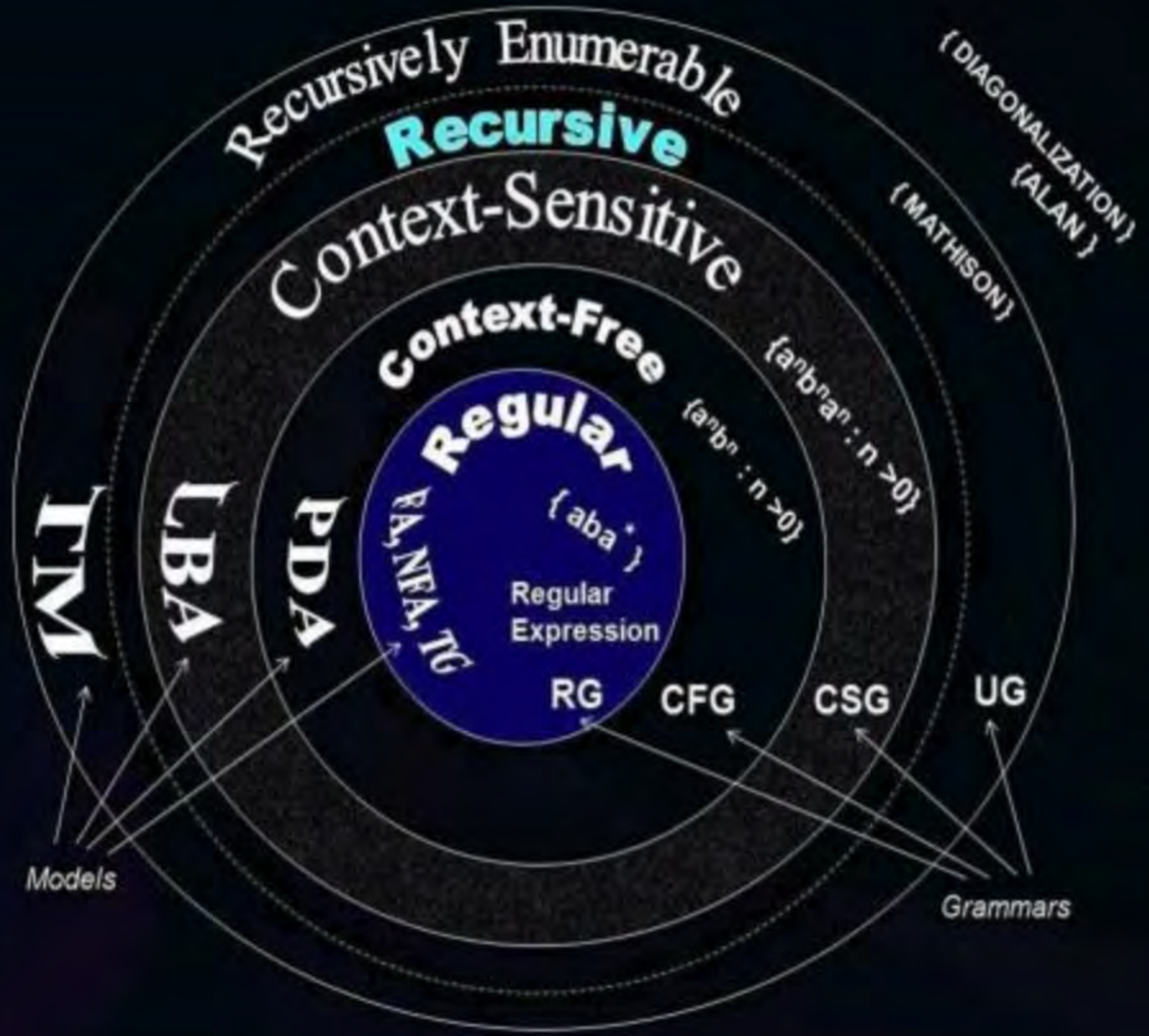
A. $L_1 = \{0, 1\}^* - L$

B. $L_1 = \{0, 1\}^*$

C. $L_1 \subseteq L$

D. $L_1 = L$

Topic : Theory of Computation



Topic : Expressive Power

Number of languages accepted by particular automata is knowns as expressive power.

(TM > LBA > PDA > FA)

1. Expressive power of NFA and DFA same. Hence every NFA is converted into DFA.
2. Expressive power of NPDA is more than DPDA. Hence conversion not possible
3. Expressive power of DTM and NTM is same.

MCQ

#Q. Let D_f, D_p are number of languages accepted by DFA and DPDA respectively. Let N_f, N_P are number of languages accepted NFA and NPDA respectively. Which of the following is true.

A. $N_f = D_f$
$N_p = D_p$

B. $N_f \supset D_f$
$N_p \supset D_p$

C. $N_f = D_f$
$N_p \subset D_p$

D. None

MCQ

#Q. In which of the cases stated below the following statement is false?
"Every nondeterministic machine M_1 there exists an equivalent deterministic machine M_2 recognizing the same language"

A. M_1 is non deterministic FA

B. M_1 is non deterministic turing machine

C. M_1 Is non deterministic PDA

D. None

Topic : DFA Construction

Construct minimal state DFA that accerpts all strings os 0’s and 1’s where each string ending with 00.

Topic : Minimization of DFA

→ For a given regular language even though many DFA exist but minimal state DFA is unique.

Ex: Complete Language: Σ^*

Topic : Minimization Algorithm

1. State equivalence algorithm
2. Table filling algorithm

Equivalent States:

Two states q_0, q_1 are said to be equivalent both $\delta\ (q_0, x)$ and $\delta(q_2,x)$, $\forall\ x \in \Sigma^*$ should result either final state or non final state.

$\delta\ (q_1, x)$

F NF

$\delta\ (q_2, x)$

Topic : Procedure of minimization

1. Elimination inaccessible states.

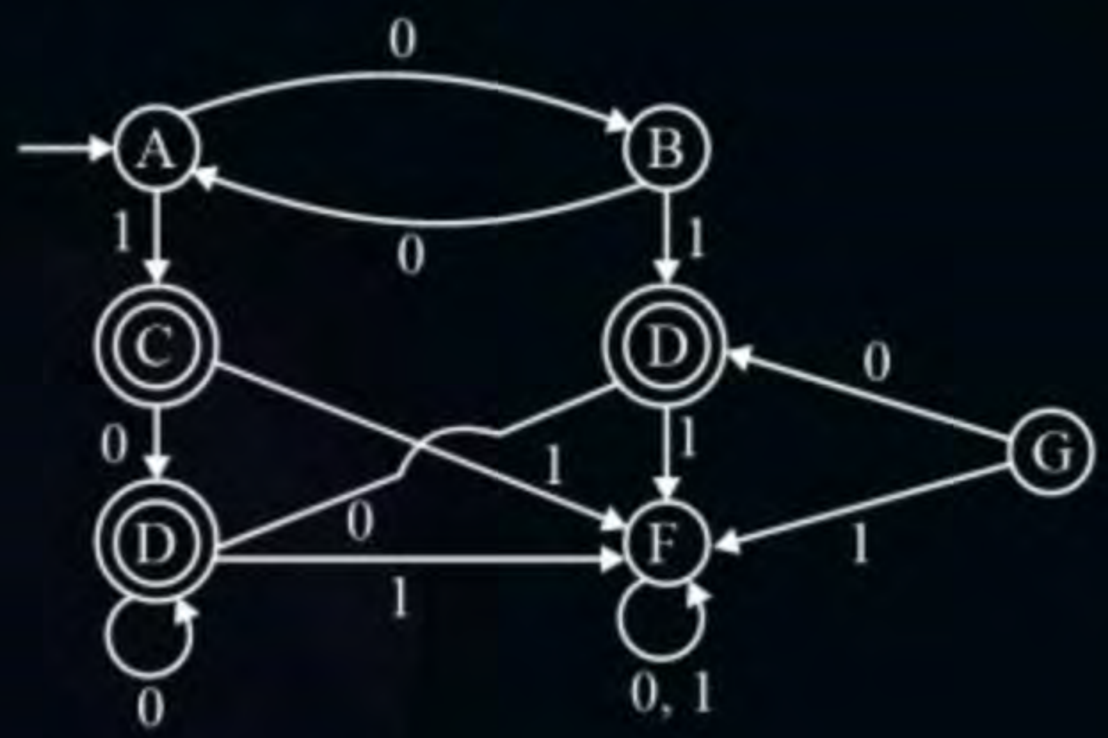
<u>inaccessible state:</u>

Any State which is not reachable from dead state is inaccessible state.

2. Apply algorithm steps
3. Merge single group into one state
4. Construct new minimized DFA

Topic : Procedure of minimization

1. Reduce states of following DFA



Step-1: Elimination inaccessible state.

Note: Dead state is different from inaccessible state.

Topic : Procedure of minimization

Step:2

State	0	1
A	B	C
B	A	D
F	F	F
C	E	F
D	E	F
E	E	F

Algorithm:

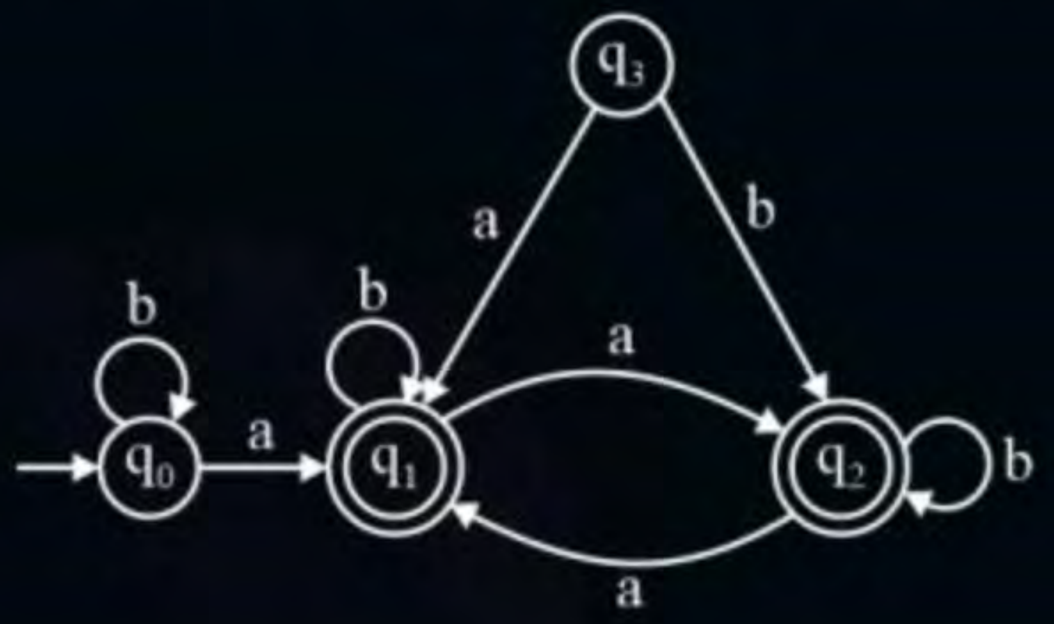
1. {A, B, F} {C, D, E}
2.
3.

Topic : Procedure of minimization

Minimized DFA

Topic : Procedure of minimization

Consider the following Finite State Automation



Topic : Procedure of minimization

Step 1: Eliminate q_3

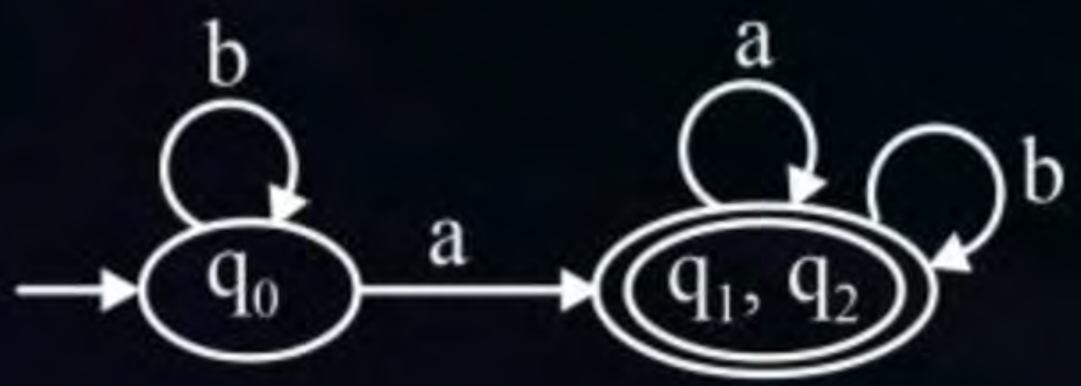
Step 2:

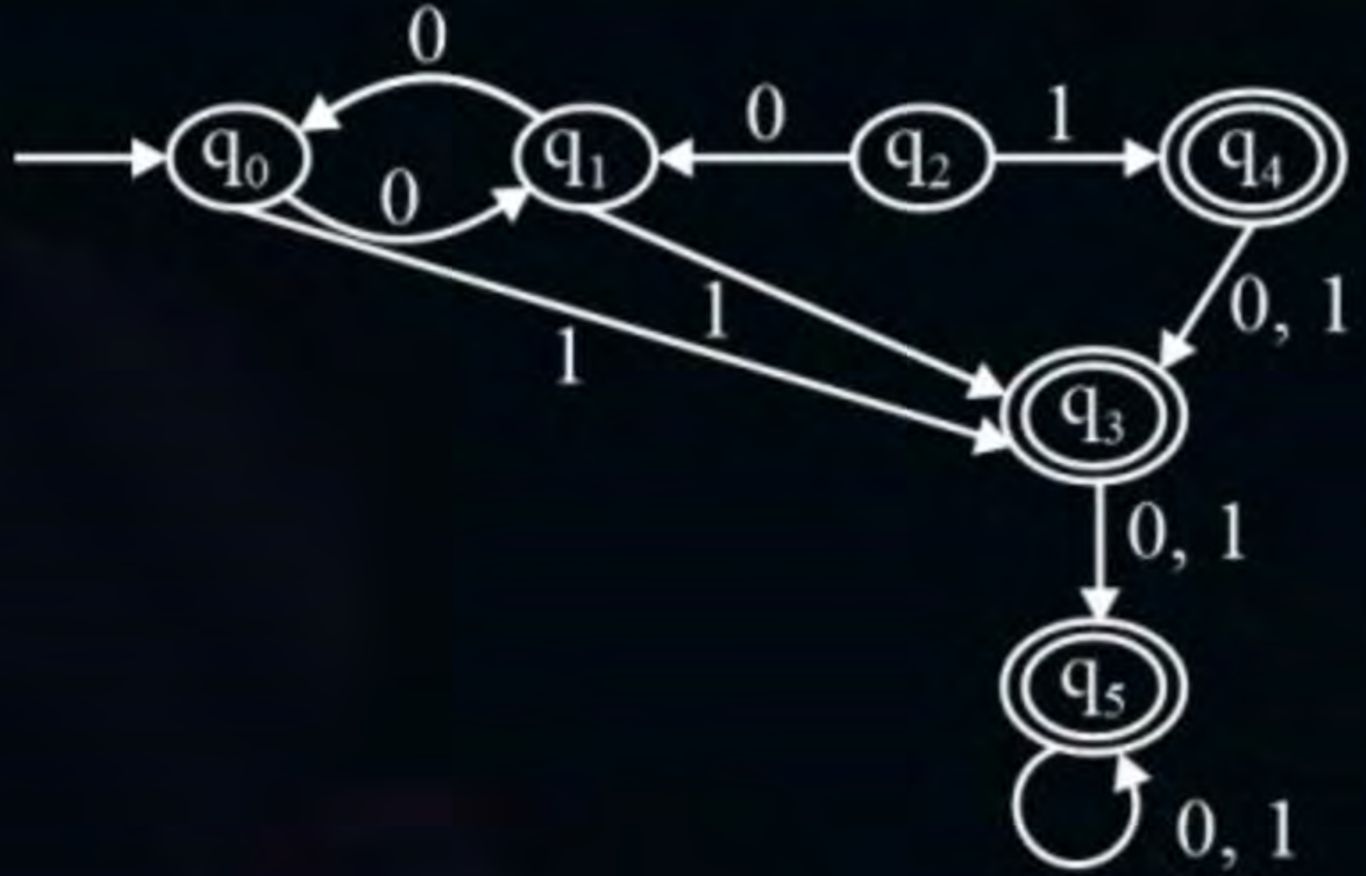
	a	b
q_0	q_1	q_0
q_1	q_2	q_1
q_2	q_1	q_2

Algorithm step

1. $\{q_0\}\ \{q_1, q_2\}$
2. $\{q_0\}\ \{q_1, q_2\}$

Minimum DFA





Minimize given DFA

Topic : Procedure

Step 1: Eliminate

Step 2:

	a	b
q_1	q_1	q_3
q_2	q_0	q_3
q_3	q_5	q_5
q_5	q_5	q_5

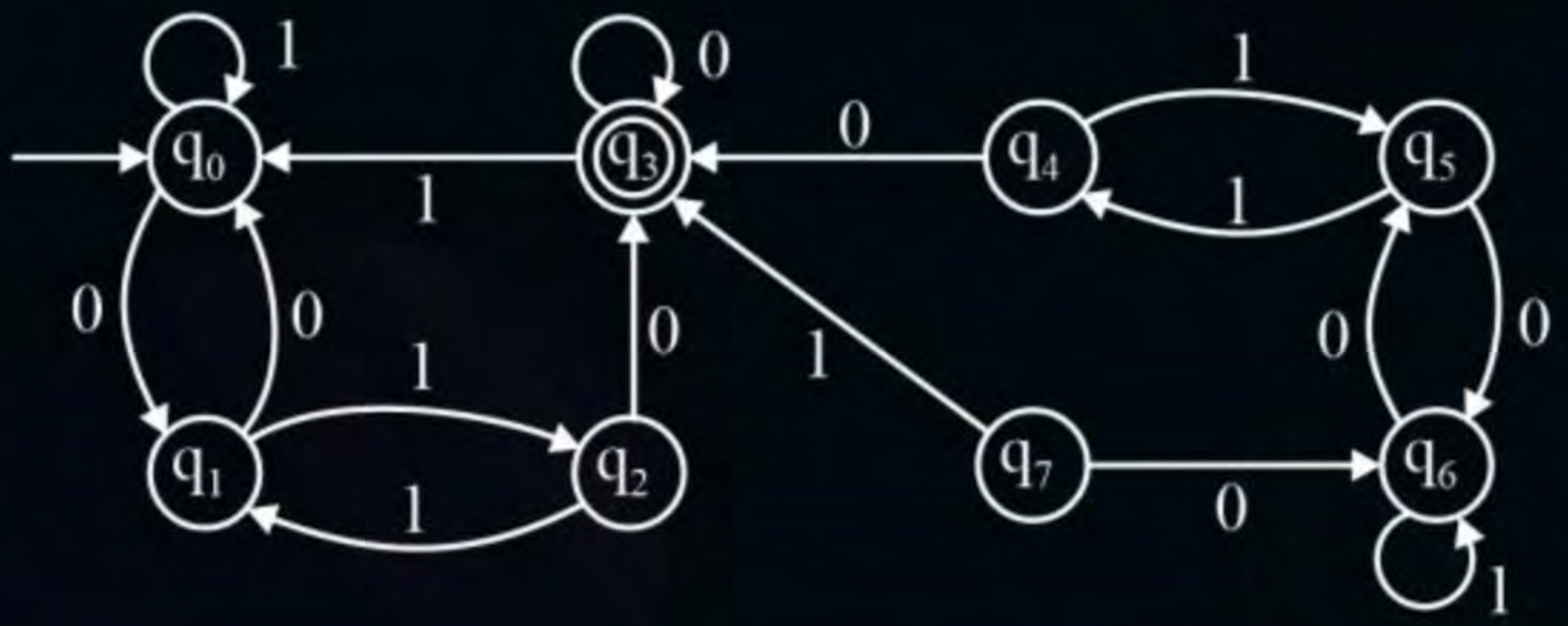
Algorithm

1. $\{q_0, q_1\}$ $\{q_3, q_5\}$

2.

Minimum DFA

Topic : Procedure of minimization



How many inaccessible states present in given DFA

[MSQ]

#Q. Consider the 5-state DFA M accepting the language $L(M) \subset$ subset $(0 + 1)^*$ shown below. For any string $w \in (0 + 1)^*$ let $n_0(w)$ be the number of 0's in w and $n_1(w)$ be the number of 1's in w.

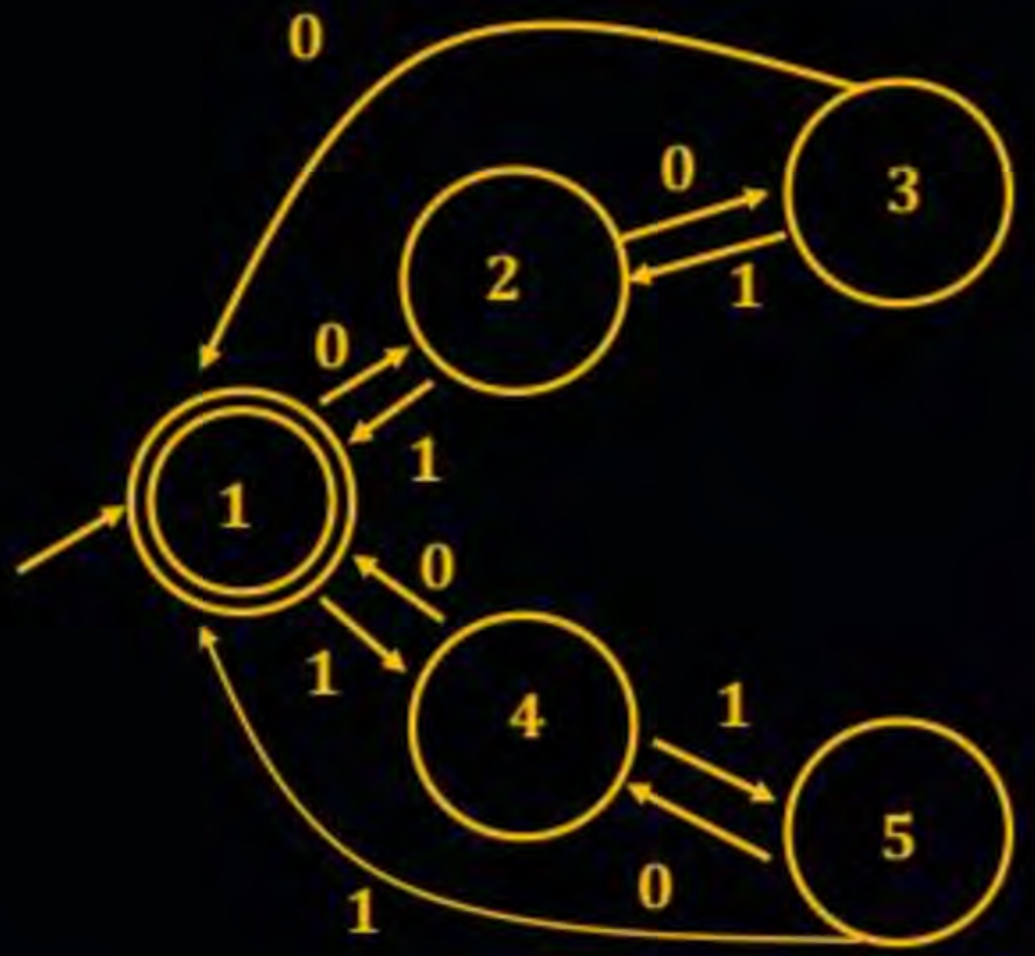
Which of the following statements is/are FALSE? **[GATE-CS-shift-I-24: 2M]**

(A) States 2 and 4 are distinguishable in M

(B) States 2 and 5 are distinguishable in M

(C) Any string w with $n_0(w) = n_1(w)$ is in L(M)

(D) States 3 and 4 are distinguishable in M



Topic : DFA

#Q. Construct the minimal DFA that accept all binary no divisible by

Topic : DFA

Construct the minimal DFA that accept all strings of a's and b's where

1. Each string ending with b.
2. Each string start with a and end with b.
3. Each string starting and ending with different symbol.
4. Each string starting and ending with same symbol.

Topic : DFA

Construct the minimal DFA that accept all string a's and b's where

1. Length of string exactly 4.
2. Number of a's length of string atleast 4.
3. Length of string atmost 4.
4. Length of string divisible by 4.
5. Number of a's exactly 5.
6. Number of b's exactly 2.
7. Number of a's divisible by 3.
8. Number of b's not divisible by 4.
9. Length of the string even.

Topic : DFA

#Q. Length of string divisible by 4.

Topic : DFA

NOTE:

→ Minimal DFA that accept exactly N length string requires (N + 2) states includes dead state.

→ Minimal DFA that accept atleast N length string requires (N + 1) states.

→ Minimal DFA that accept atmost N length string requires (N + 2) states includes dead states.

→ The minimal DFA that accept length of the string divisible by N then requires N states.

Construct a minimal DFA that accept all string a's and b's. where number of a's divisible by 2 and number of b's divisible by 3.

Topic : DFA

How many number of state are there with minimum DFA for the following state.

a) Number of a's divisible by 2 and number of b's not divisible by 3.

b) Number of a's divisible by 2 and number of b's atleast 3.

c) Number of a's atleast 2 and number of b's atleast 3.

d) Number of a's exactly 2 and number b's atleast 2.

e) Number of b's atmost 3 and number b's exactly 3.

f) Number of a's not divisible by 2 or number of b's exactly 3.

NOTE:

Number of States of DFA on length conditions

(i) Then in the given condition on length if one number divide other number then number of states of minimal DFA for "and" automata is LCM of given condition.

(ii) Number of states of minimal DFA for "OR" automata is GCD of given condition.

(iii) In the given length condition one number not divide other number then

→ If GCD of given condition is 1 then number of states of 'and' automata OR automata is multiplication of given condition.

Topic : DFA

(iv) The given condition on length one number not divides other and GCD of given condition is not equal to 1 then number of states of 'and' automata, number of states of 'OR' automata is LCM of given condition.

Find the number of stage of minimal DFA for the following matrix.

(Length of the string divisible by 3 or divisible by 6)

GCD (3, 6) = 3

Length of the string di is by 4 and di by 6

(v) LCM (4, 6) = 12

(vi) Number of a's divisible by 4 AND number of b's divisible by 6.

$4 \times 6 = 24$

Topic : DFA

Find the number of stage of minimal DFA for the following matrix.

Length of the string divisible by 3 or divisible by 6

GCD (3, 6) = 3

Length of the string di is by 4 and di by 6

LCM (4, 6) = 12

Number of a's divisible by 4 AND number of b's divisible by 6.

$4 \times 6 = 24$

Topic : DFA

1. Length of the string divisible by 2 and divisible by 1
2. Length of string divisible by 2 OR divisible by 4.
3. Length of string divisible by 3 divisible by 4
4. Length of string divisible by 3 OR divisibly by 4
5. Length of string divisible by 6 OR divisibly by 8
6. Number of a's divisible by 6 and number of divisible by 8.

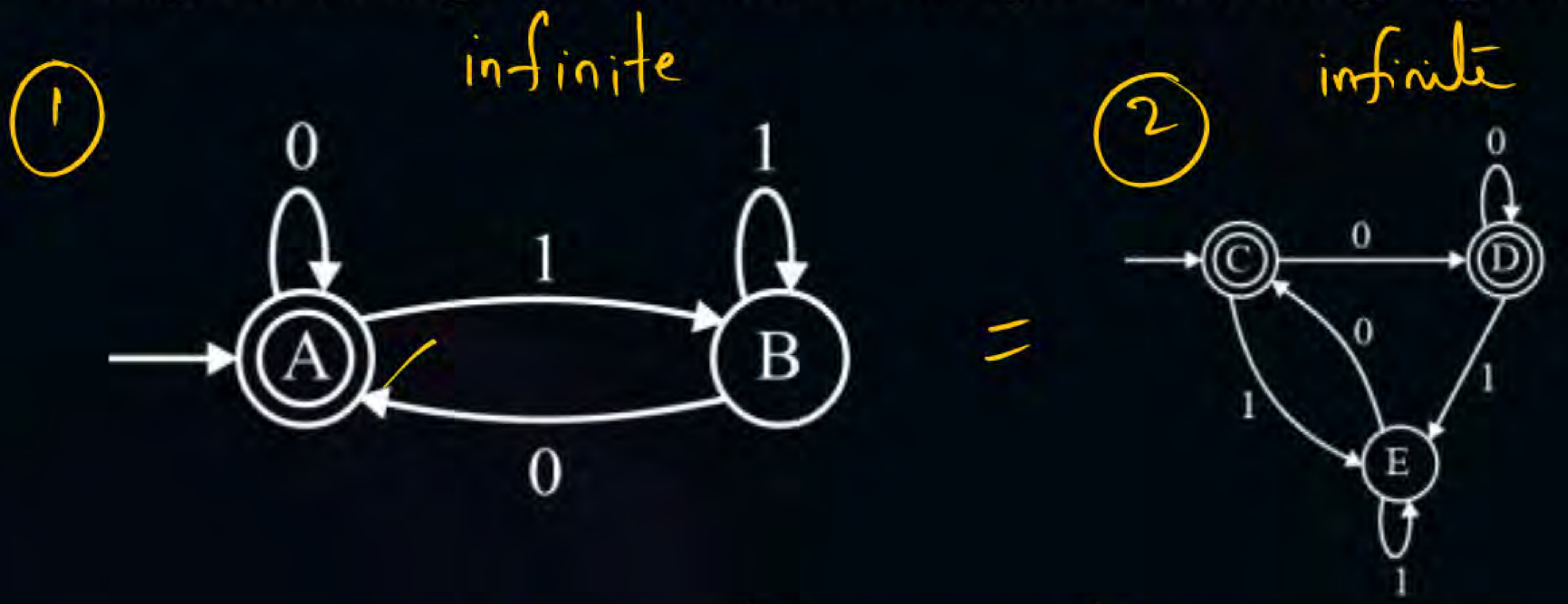
Topic : Decision Properties of Finite Automata

Decidable problems

1. Equivalence Problem
2. Finiteness Problem → Loop (or) cycle exist → infinite.
   otherwise → finite
3. Emptiness Problem → { }

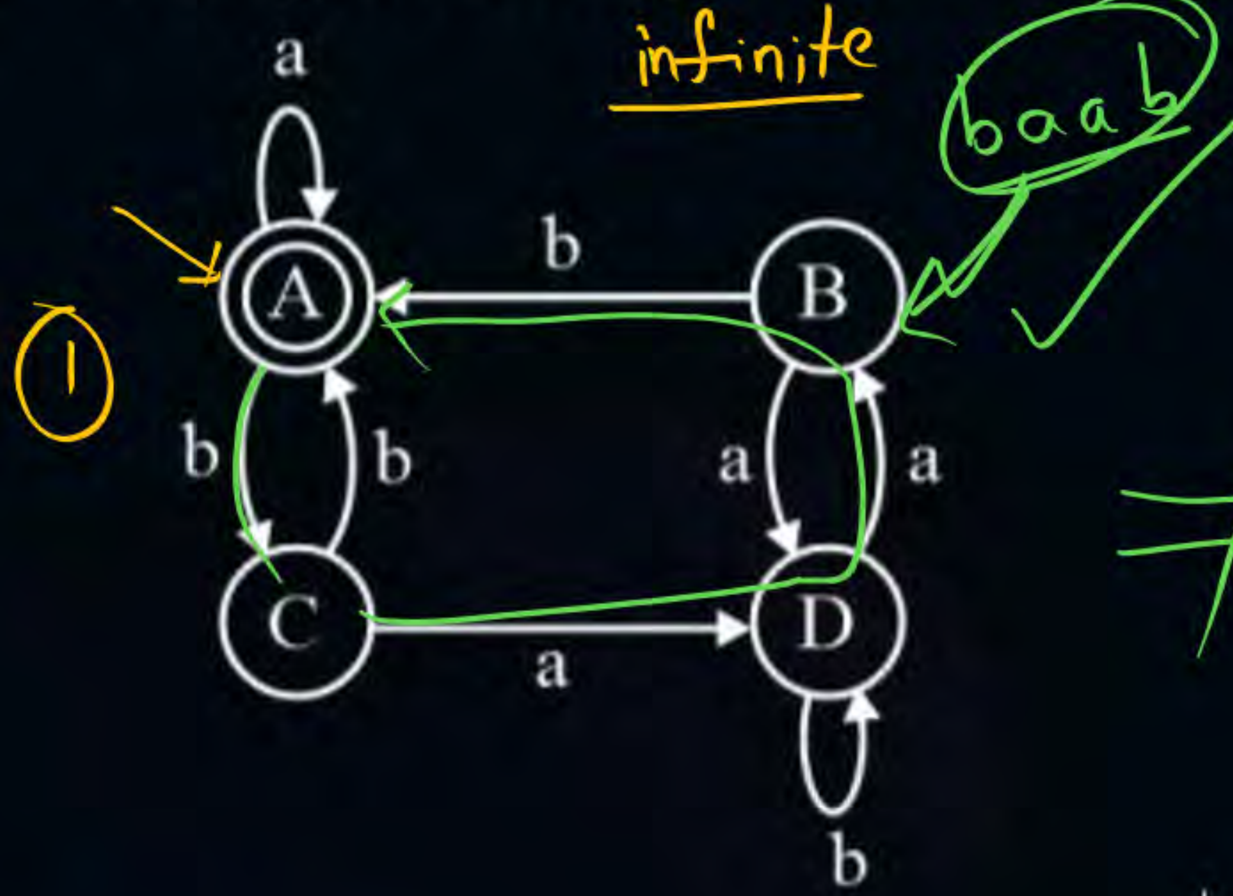
× {ε} → finite language

#Q. Verify following two automata accepts same language or not

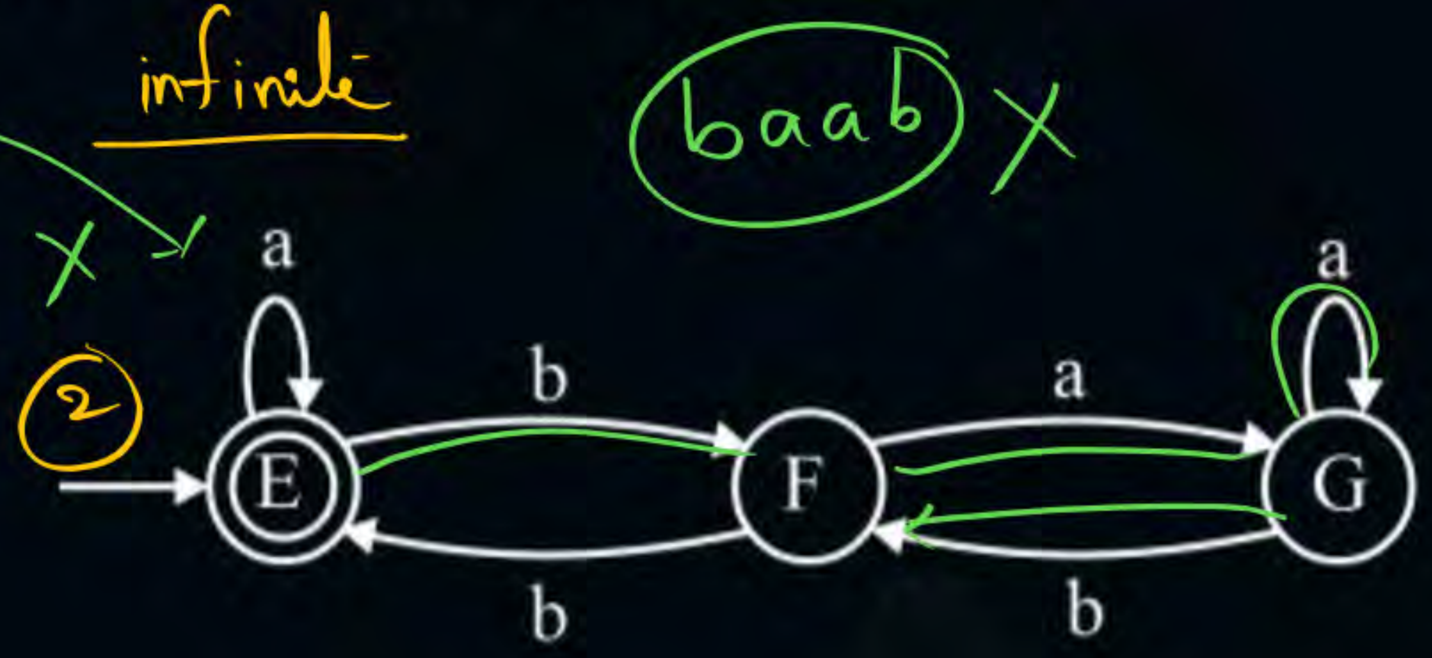


	0	1
(A, C)	A D	B E
(A D	A D	(B, E)
(B, E)	A C	B E

#Q. Verify following two automata accepts same language or not



$\neq$



	a	b
(A, E)	(A E) ✓	(C F) ✓
(C, F)	(D G) ✓	(A E) ✓
(D, G)	(B G)	(D F)
(B G)	(D G)	(A, F)

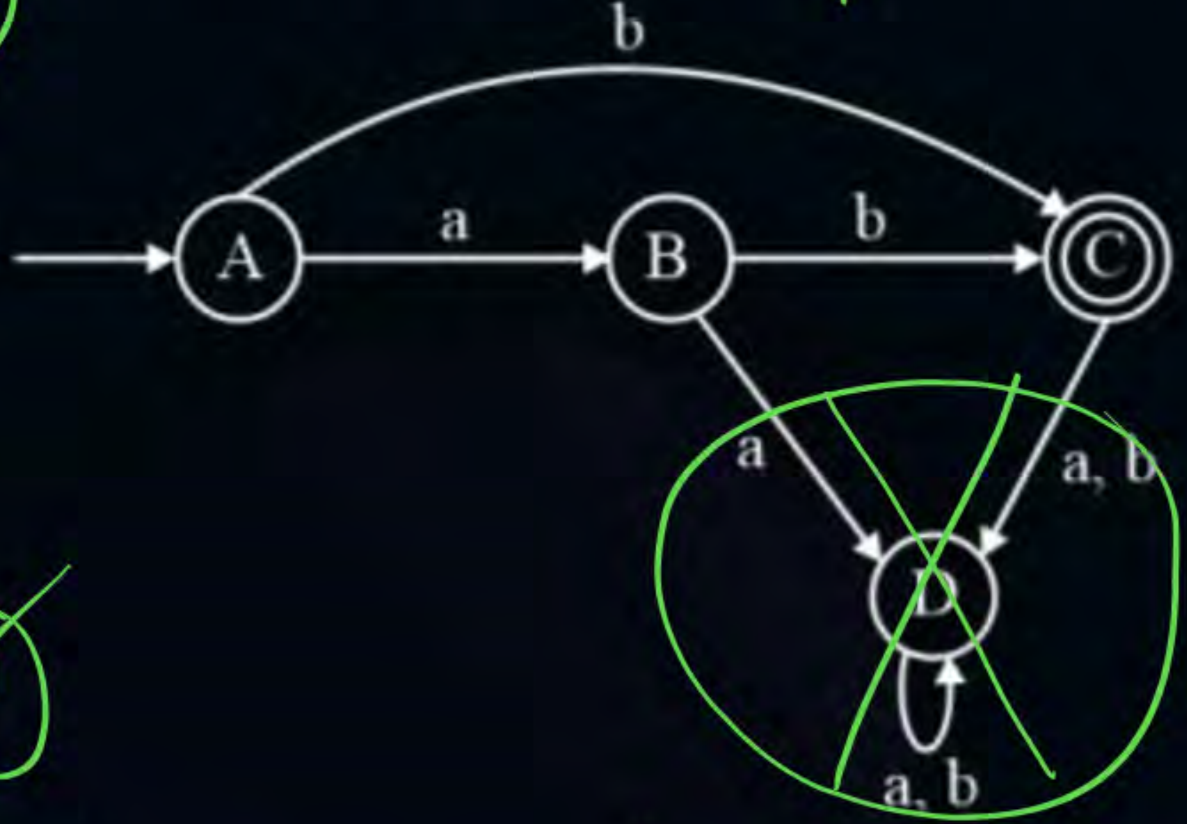
Loop (or) Cycle exist (or) not ?

#Q. Which of the following automata accepts infinite Language

(a) finite

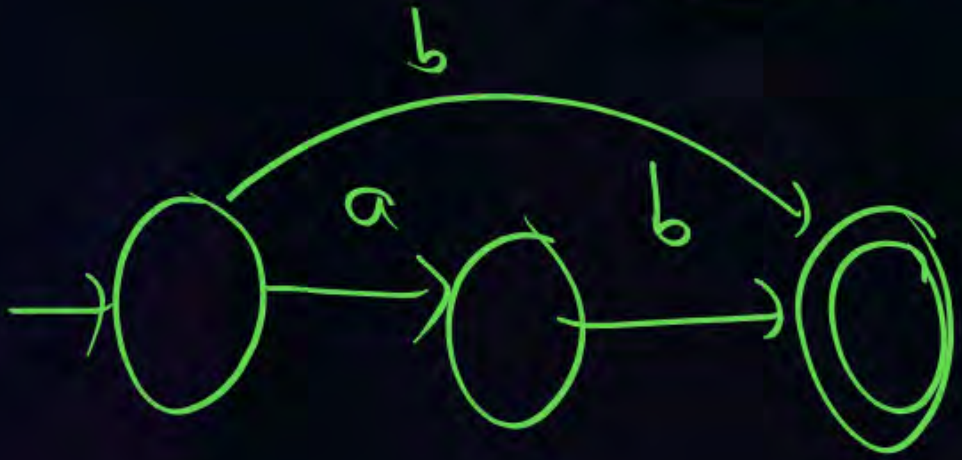
(b) infinite

(1)

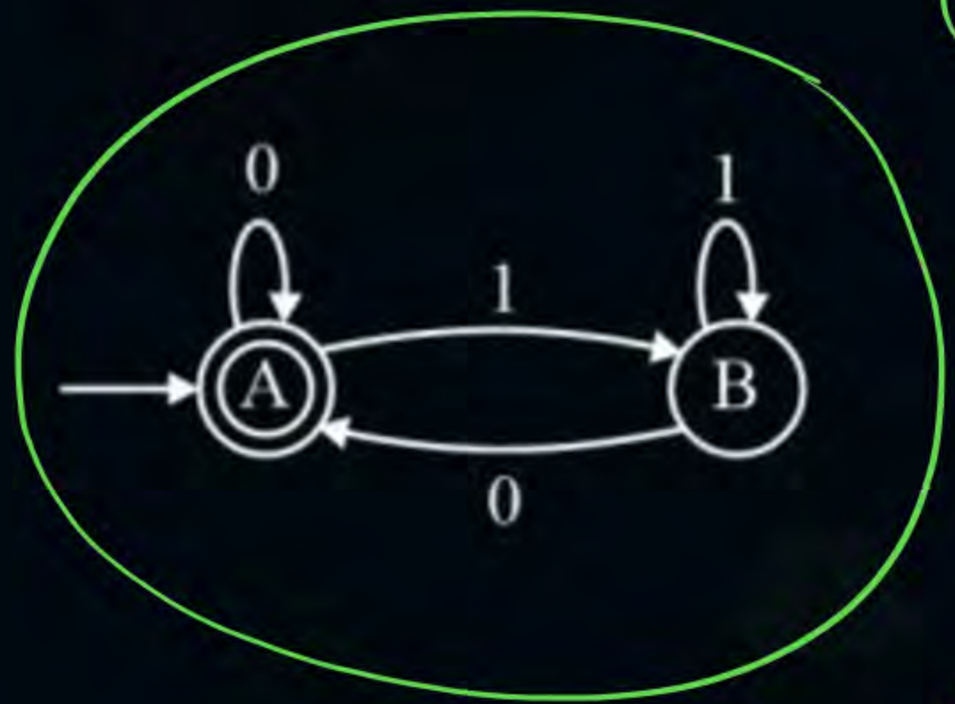


(1)

(2)



{b, ab}



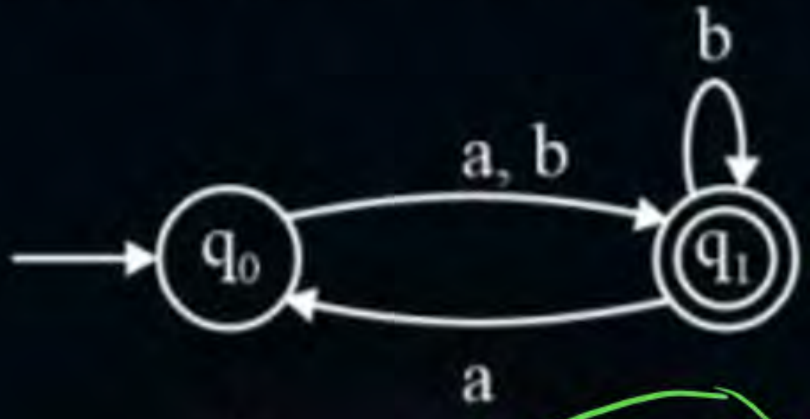
infinite

1. Eliminate unreachable states

2. Eliminate states from which final state is not reachable.

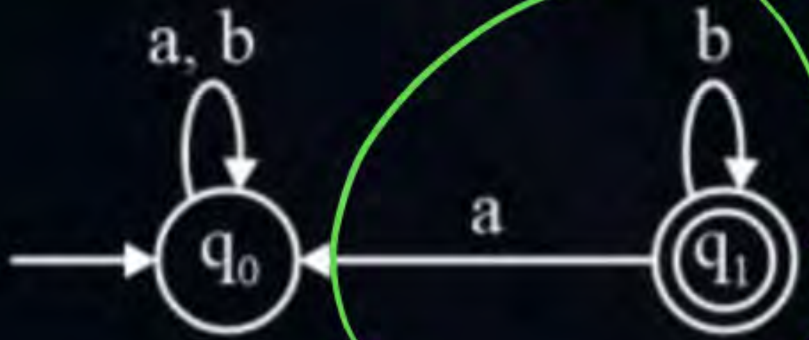
#Q. Which of the following automata accepts Empty Language.

1.



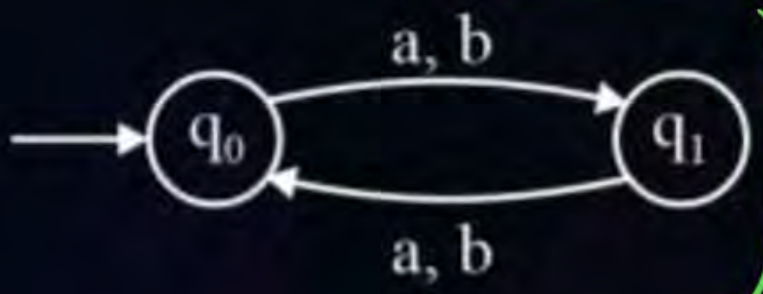
→ non empty

2.



→ Empty

3.



} Empty

(1) Eliminate unreachable states

(2) if no final state then empty language

true → S_1 : In any DFA all states are final then it accepts Complete Language

false → S_2 : In any DFA all states are non final then it accepts finite Language.

Empty Lang

{ab, ba}

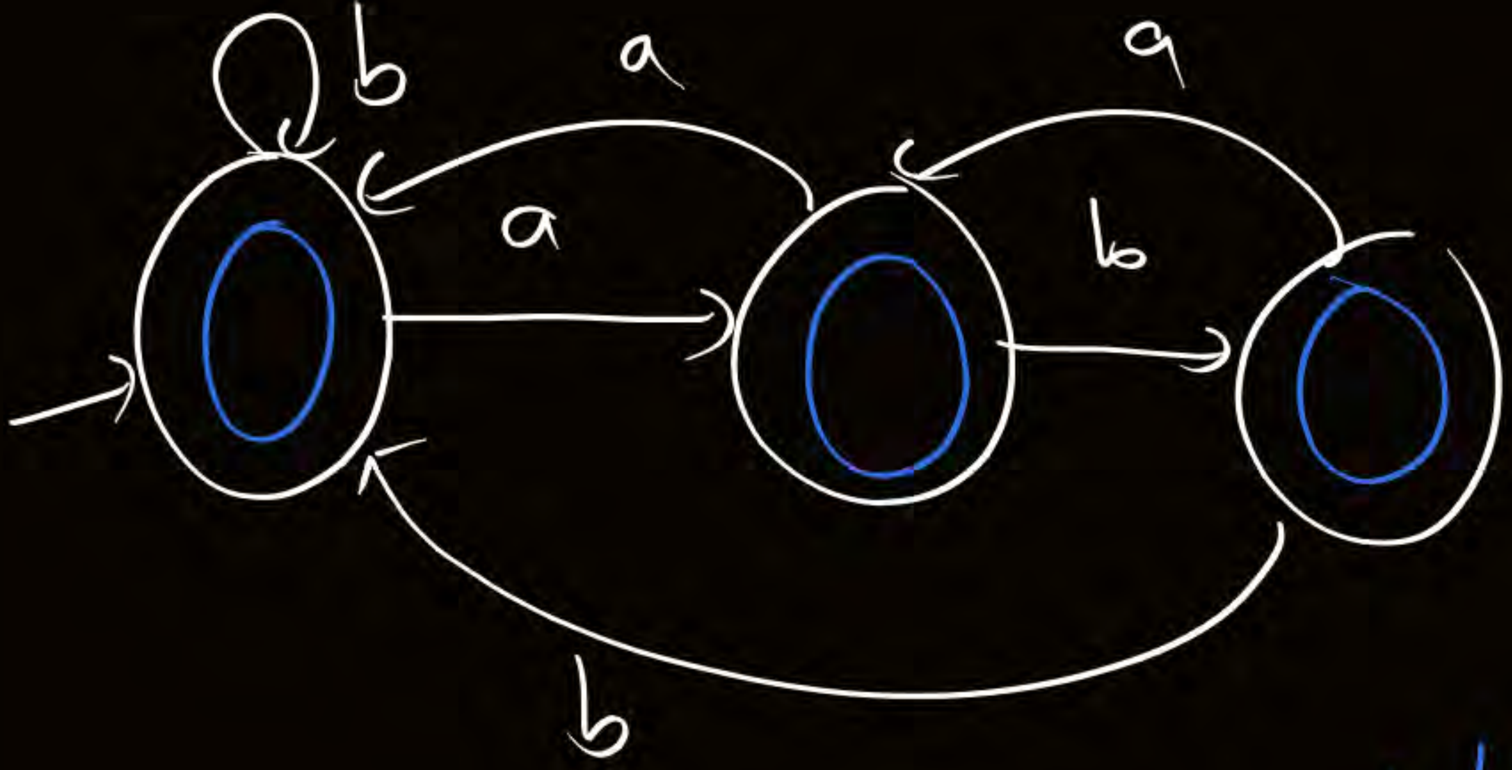
which of the above is true?

ⓐ S_1 only

ⓑ S_2 only

ⓒ both S_1, S_2

ⓓ none.



$\{\}$

Empty

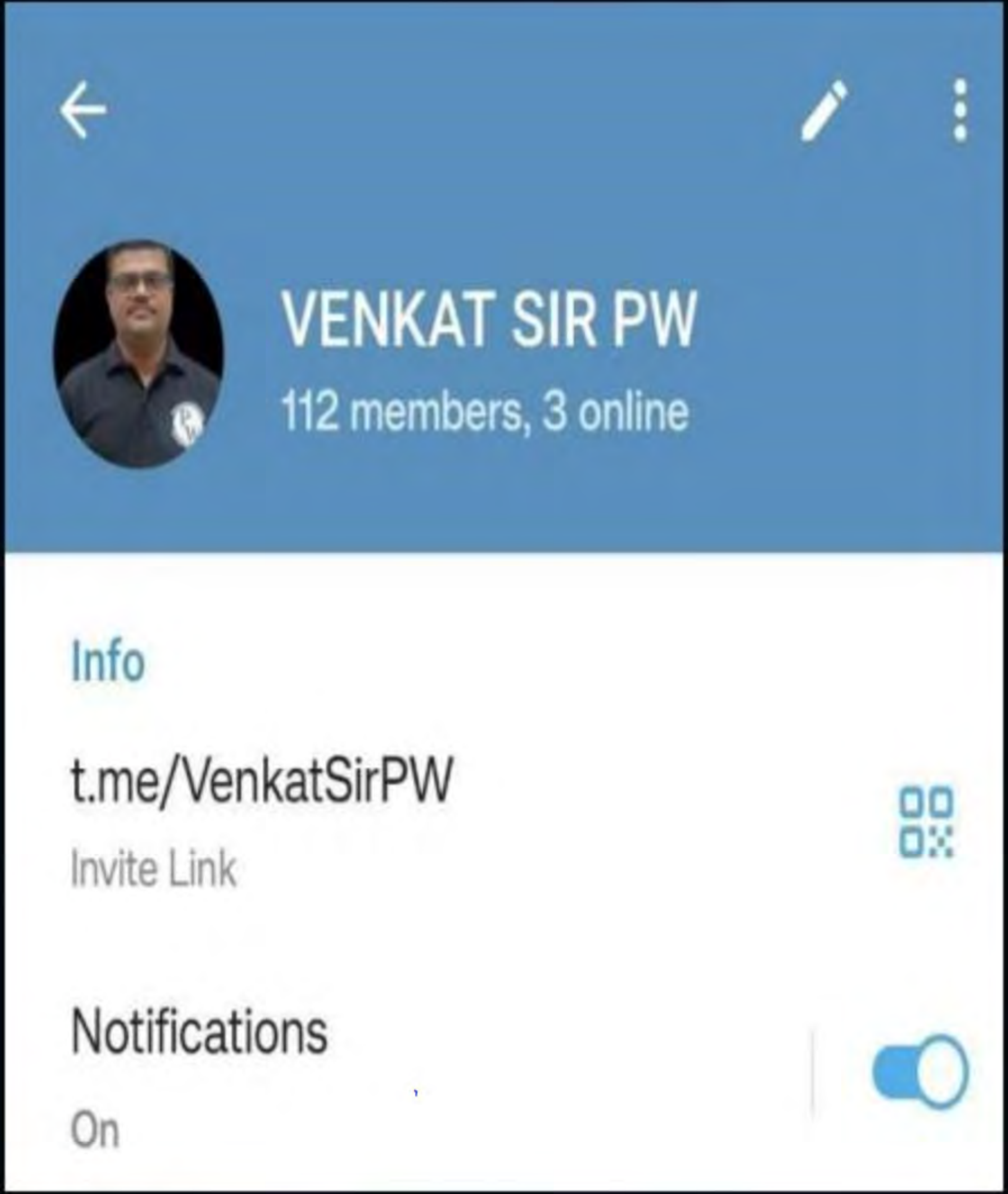
$\Sigma^* - \{\}$

Complete Lang Σ^*

1. Unreachable states

2. Eliminate any state from which final state is non reachable.

3. Loop (or) Cycle exist } infinite


VENKAT SIR PW
112 members, 3 online
Info
t.me/VenkatSirPW
Invite Link
Notifications
On


@VENKATSIRPW

THANK - YOU