CS & IT ENGINEERING

THEORY OF COMPUTATION

Regular Expressions

Lecture -01

By- Venkat sir

Recap of Previous Lecture

Topic

{DFA Construction
min DFA}

?????

NFA

ε-closure()

ε-NFA

Topics to be Covered

Topic MEALY MACHINE & Moore Machine

Topic ?? Regular Expression

Topic ?? Finite Automata ⟺ Regular Expression

Topic ?? Regular Grammar ⟺ Regular Expressions

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

[MCQ]

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

A. $R_1 \cap R_2$ is not regular

B. $R_1 \cup R_2$ is not regular

C. $\Sigma^* - R_1$ is regular

D. R_1^* is not regular

[MCQ]

#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

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

Which one of the following is CORRECT?

A. Only I

B. Only II

C. Both I and II

D. Neither I nor II

Q

Consider the following two statements:

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

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

[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

Q

Consider the following statements:

I. 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.

Which of the above statements is/are TRUE?

[2020: 1 Mark]

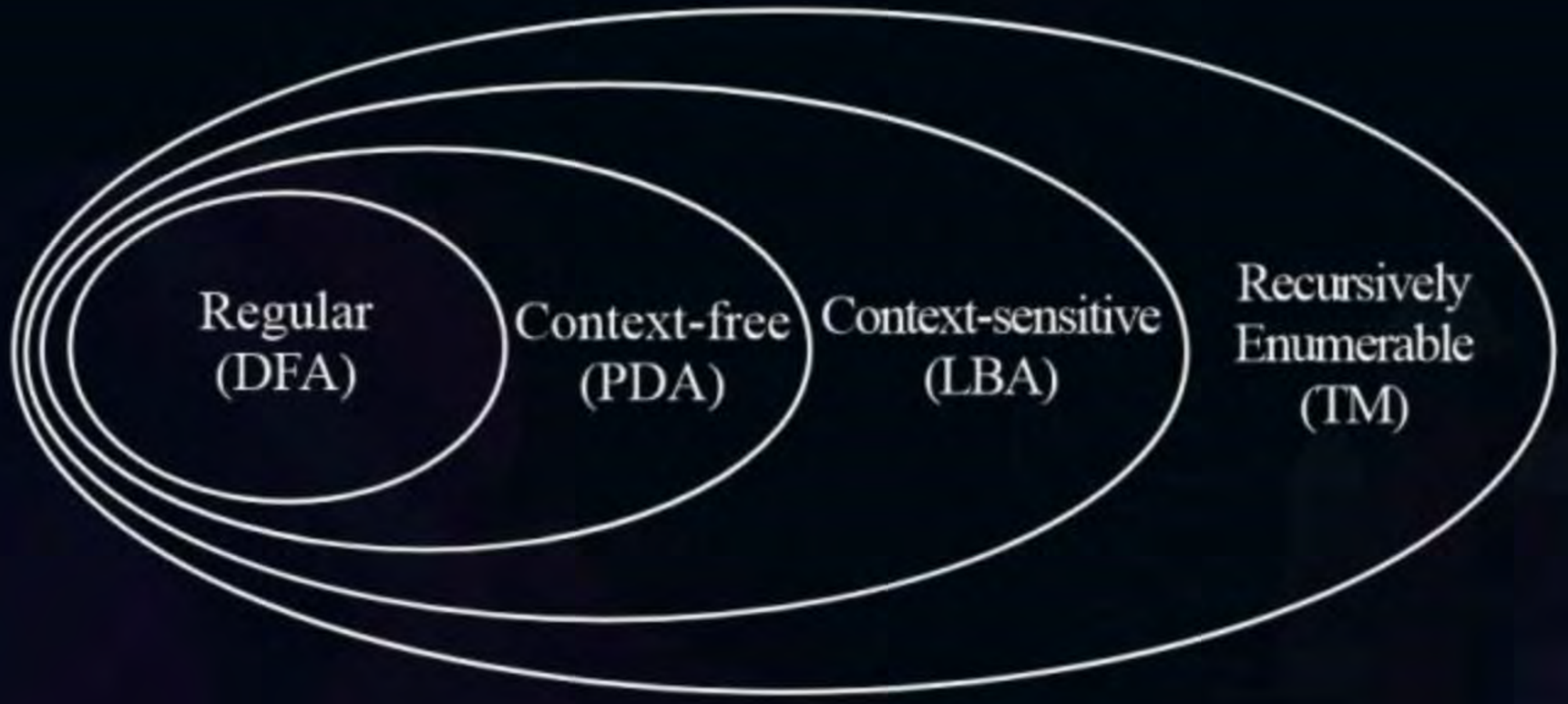
A. Neither I nor II

B. II only

C. I only

D. Both I and II

Topic : Regular expression

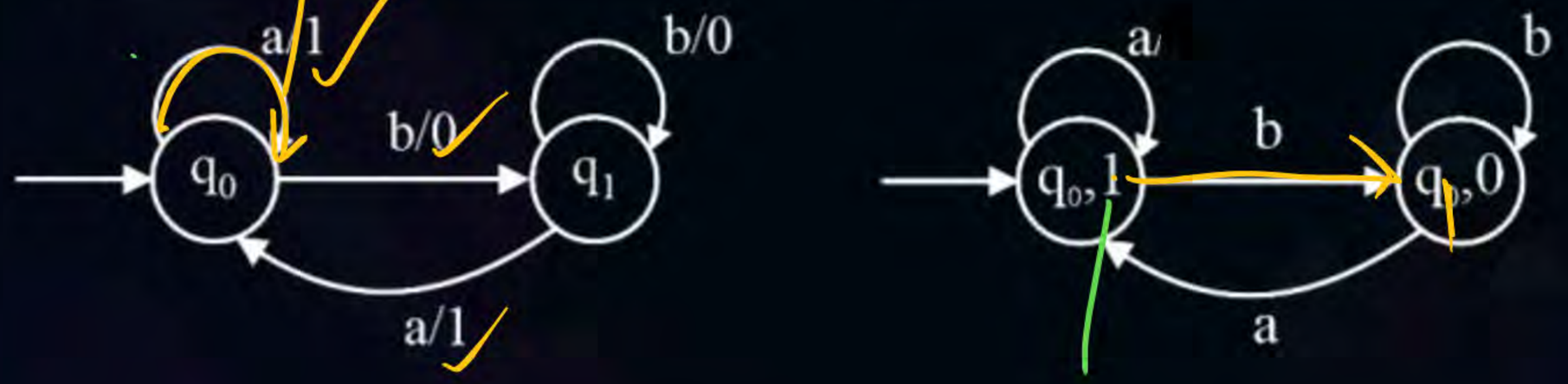


Digital Circuit Design

input abababab → F.A with output → output 01

4 Mealy Machine

5 Moore Machine



- 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 Def : $(Q, \Sigma, q_0, \delta, \Delta, \lambda)$

1 2 3 4 5 6

- Q : finite number of states
- Σ : input alphabet
- q_0 : initial state [only one]
- 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]

aaaaaaaabb

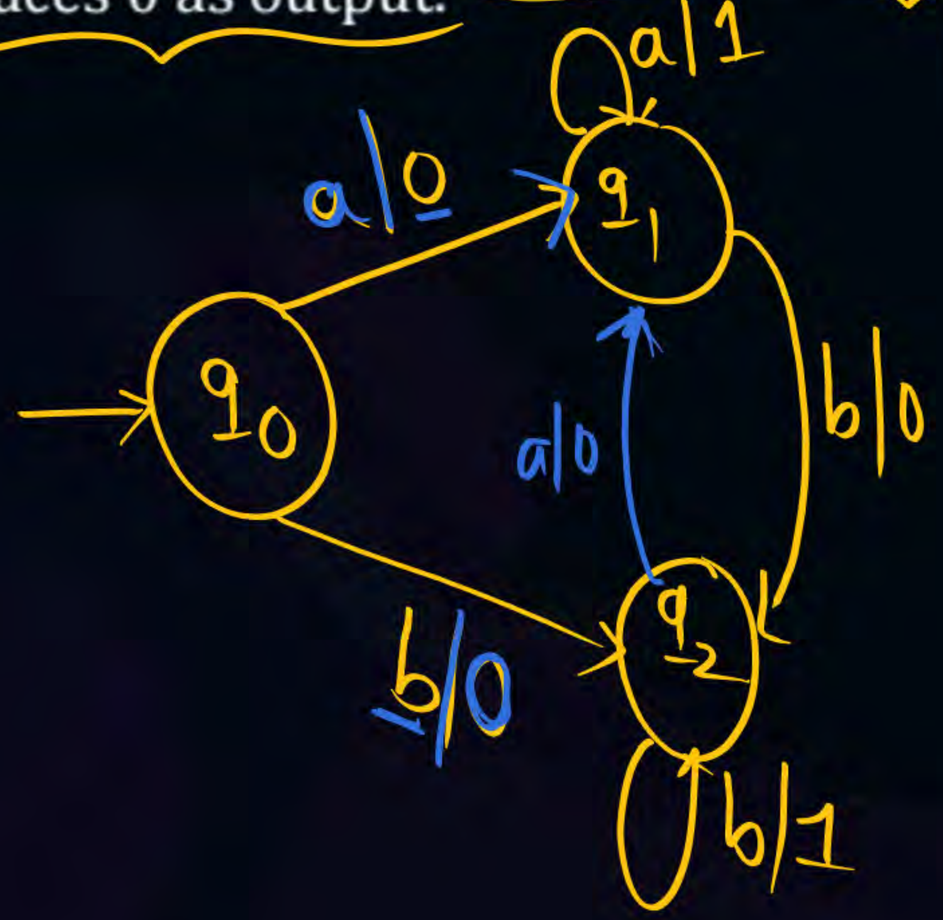
aba

aaaaaab

bbbbbaa

#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.

000



a

abab → Mealy m/c → 0, 1

aa

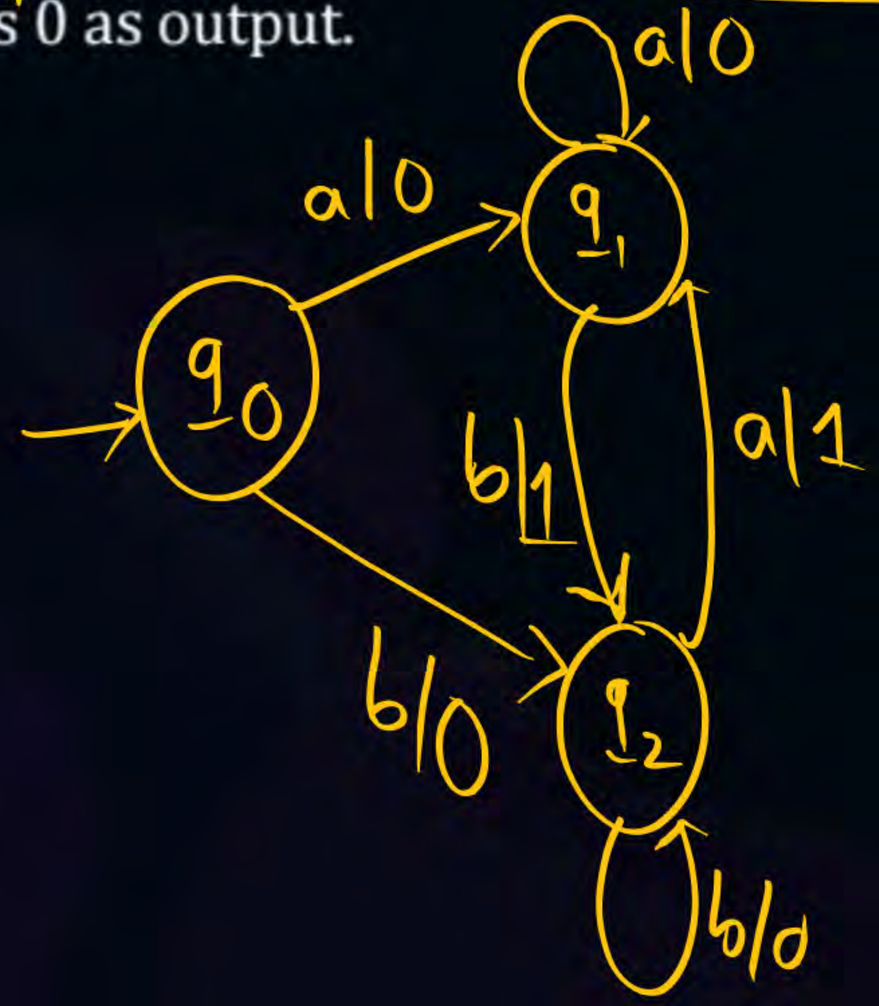
$\Sigma = \{a, b\}$

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

bbbbbbbbbbbbba

[NAT]

#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.



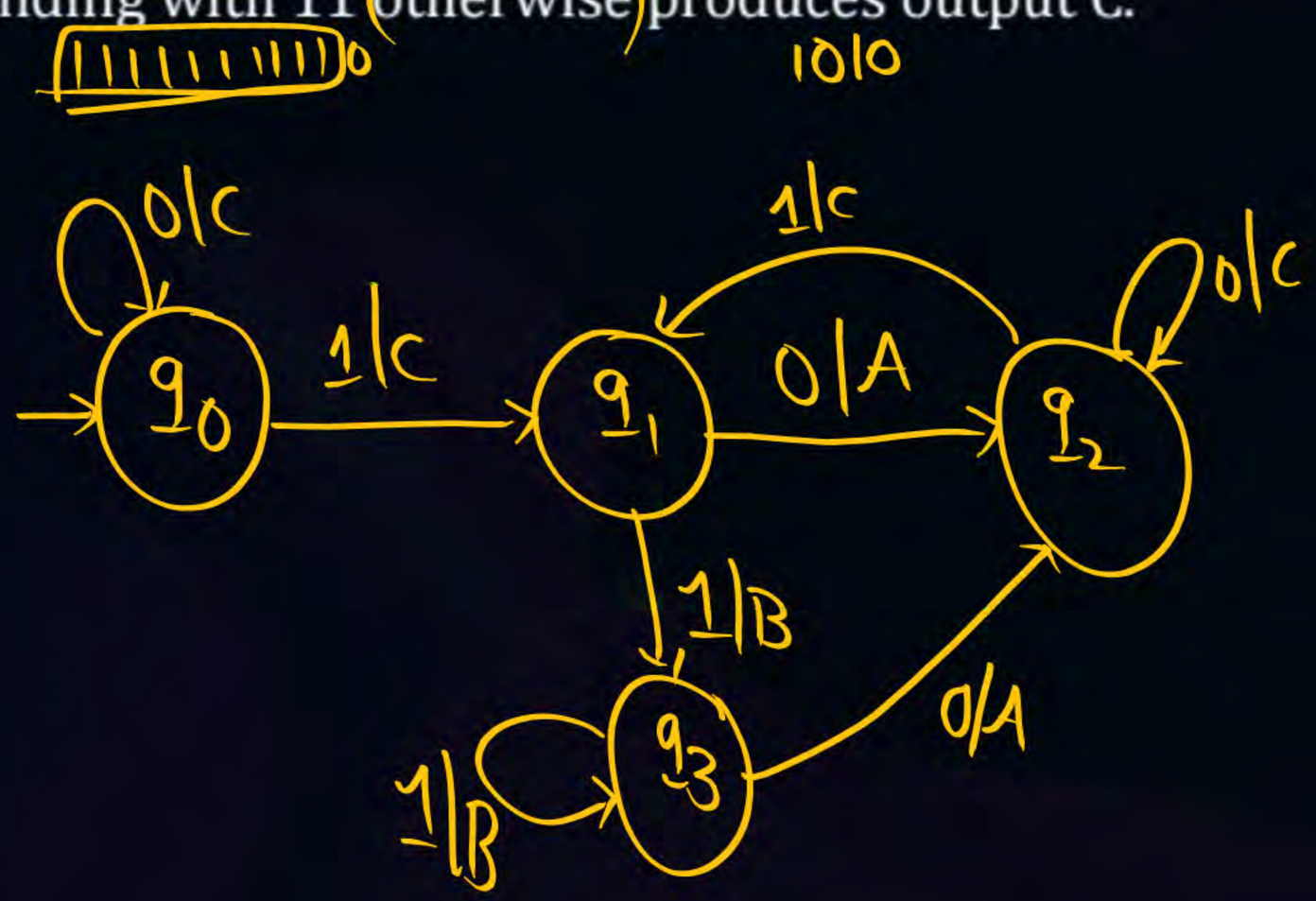
ab, ba } 1

a, b, aa, bb } 0

[NAT]

#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.

1010



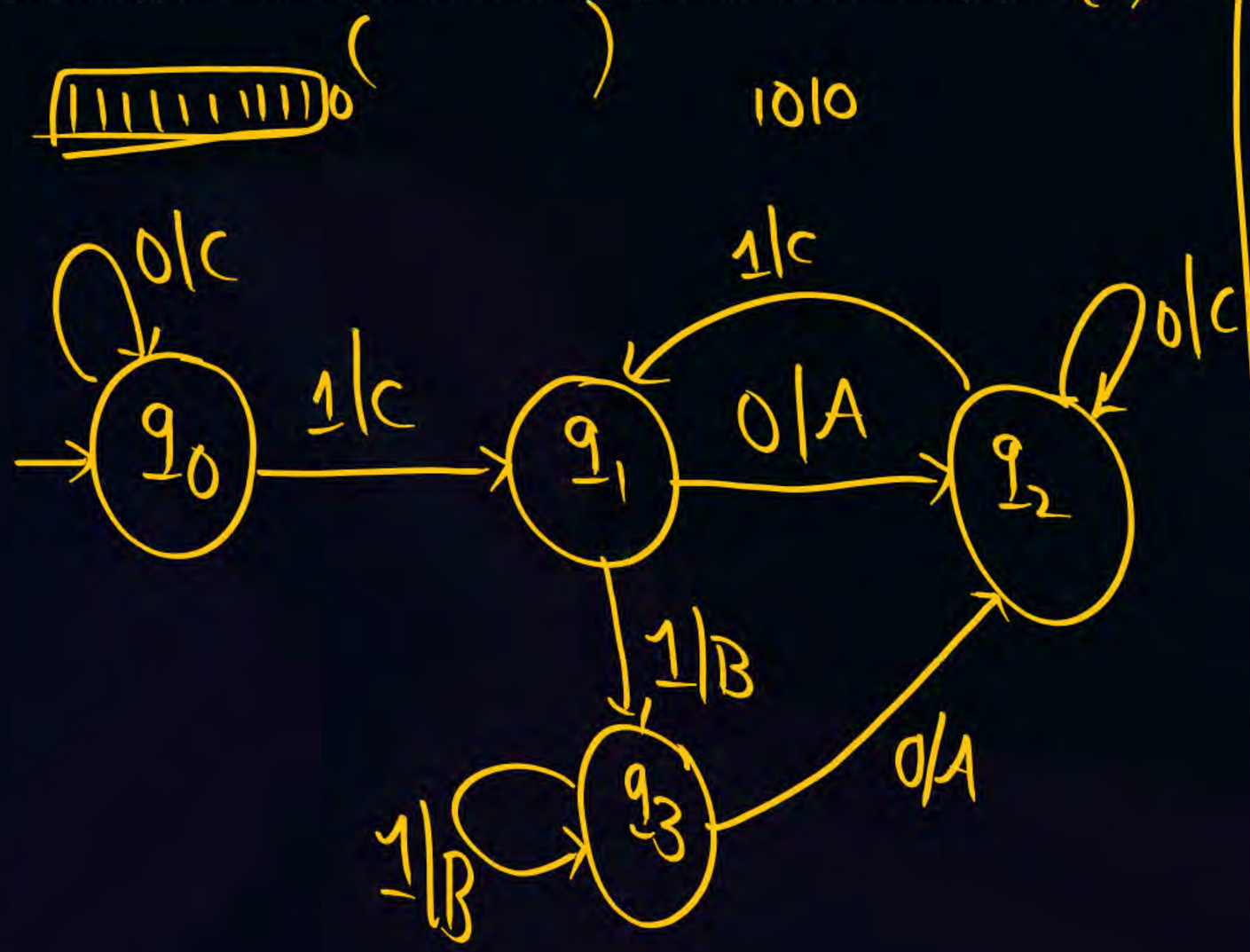
{A, B, C}

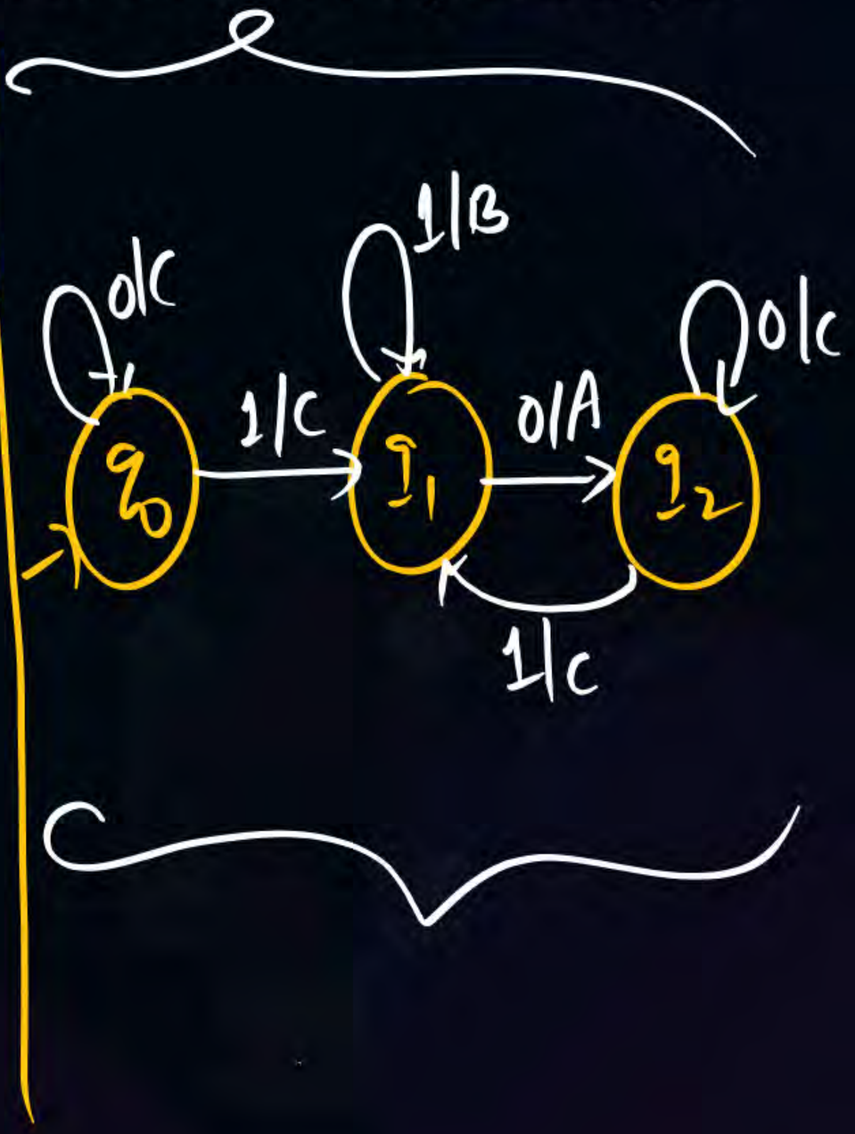
10 } A

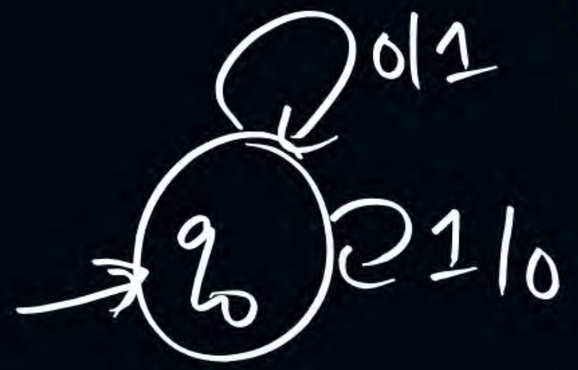
11 } B

01, 00, 0, 1 } C

#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.







What type of output is produced by this m/c ?

1's Complement

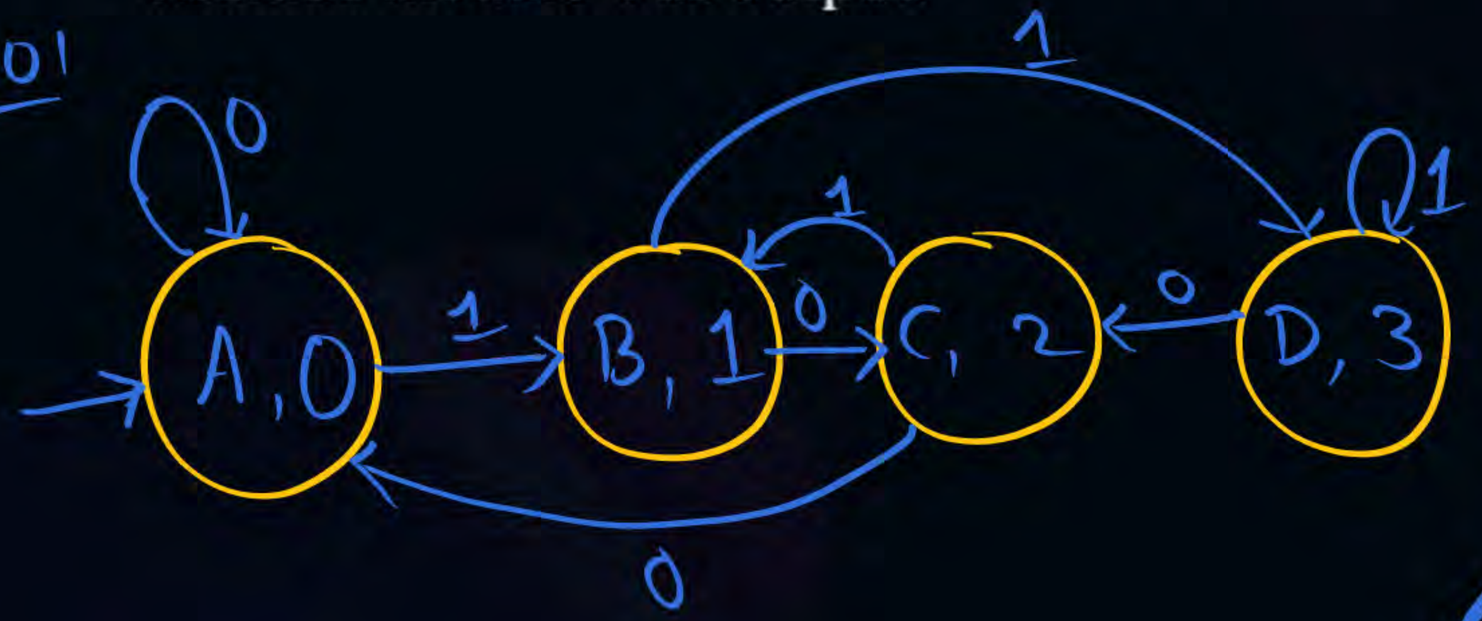
[NAT]

#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~~ numbers as input and produces Residue modulo 4 as output.

0001



1000

8

000
001
010
011
100
101
110

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

$\boxed{\text{moore}}$

6, 7, 8, 9

4) 0, 1, 2, 3, 4, 5,

$\Delta = \boxed{0, 1, 2, 3}$ = output

Reminders

CSIT

Syllabus

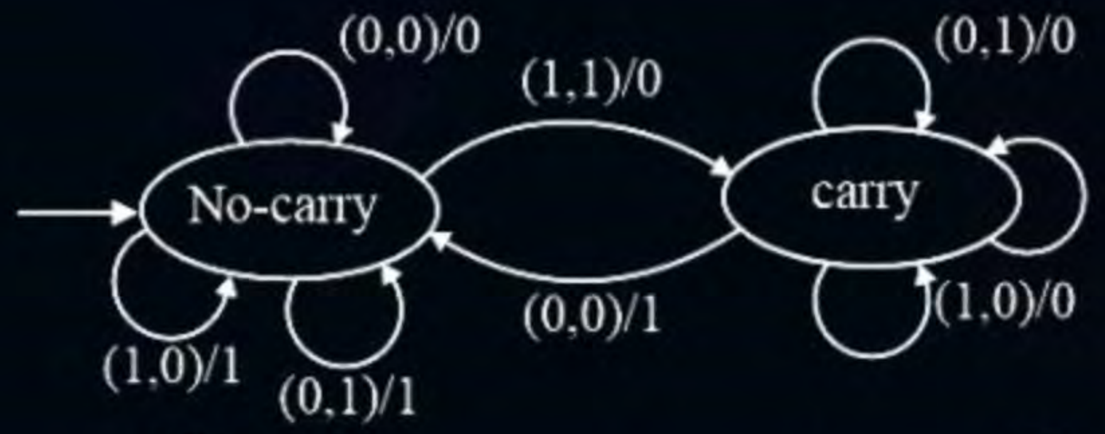
1. C programming
2. Digital Logic

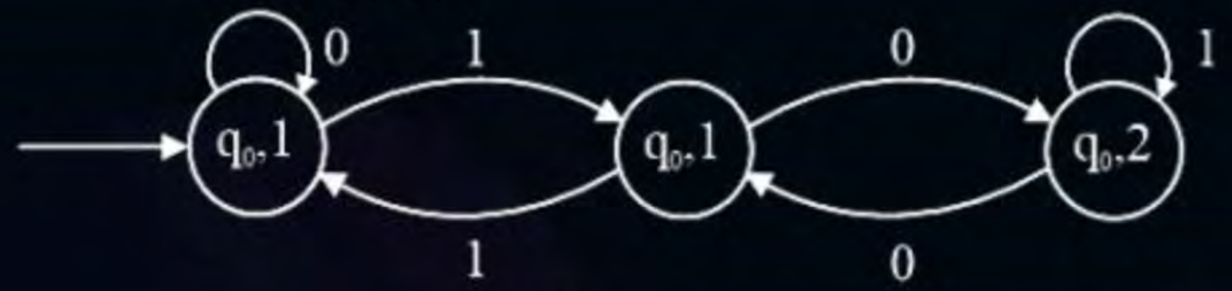
[NAT]

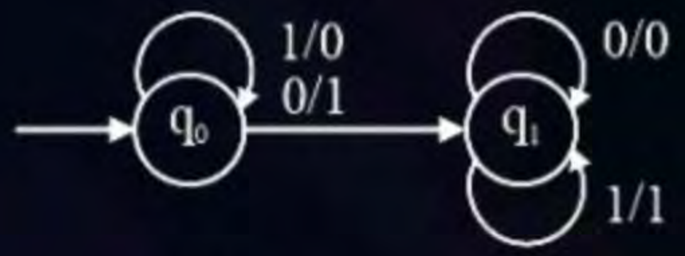
#Q. Construct Moore machine that takes all binary strings as input and produces Residue modulo 5 as output.

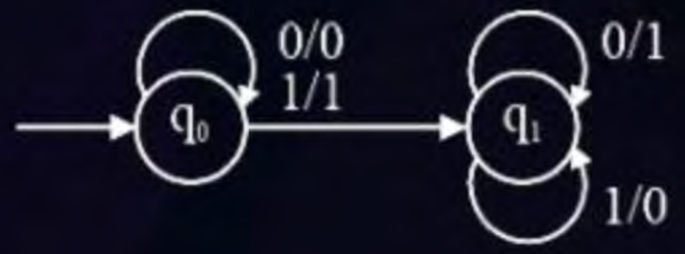
[NAT]

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

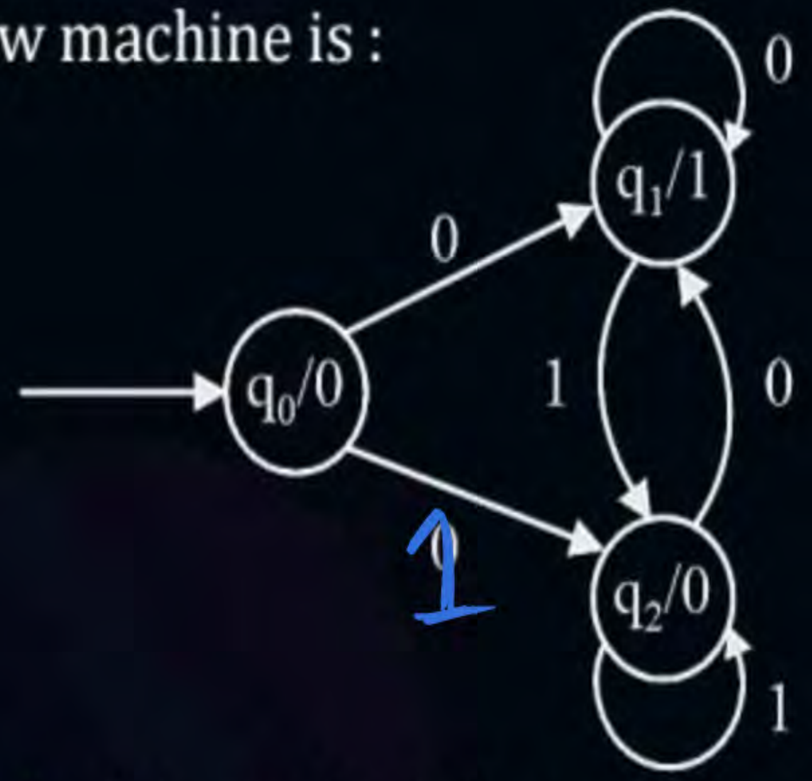

(0,0)/0
(1,1)/0
(0,1)/0
No-carry
carry
(0,0)/1
(1,0)/0
(1,0)/1
(0,1)/1


0
1
0
1
q_0,1
q_0,1
q_0,2
1
0


1/0
0/1
0/0
q_0
q_1
1/1


0/0
1/1
0/1
q_0
q_1
1/0

#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

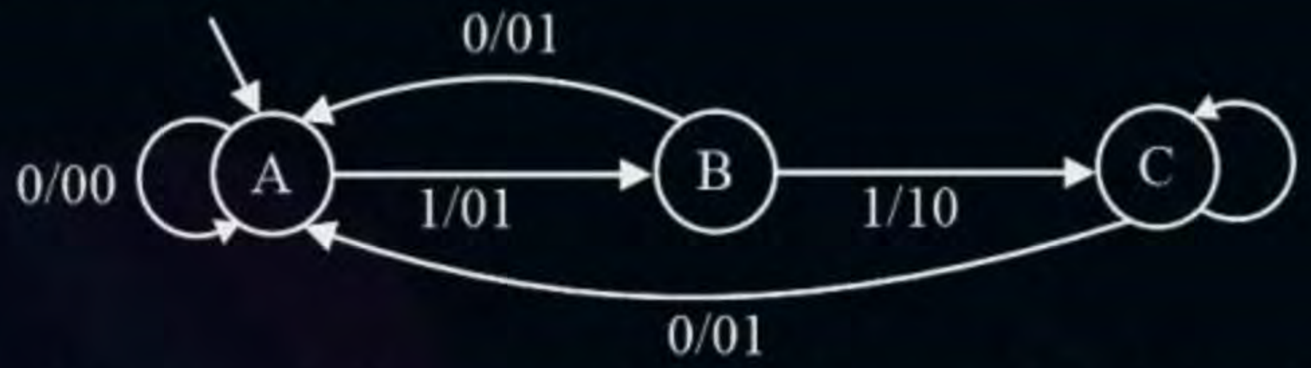
B 10

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.



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

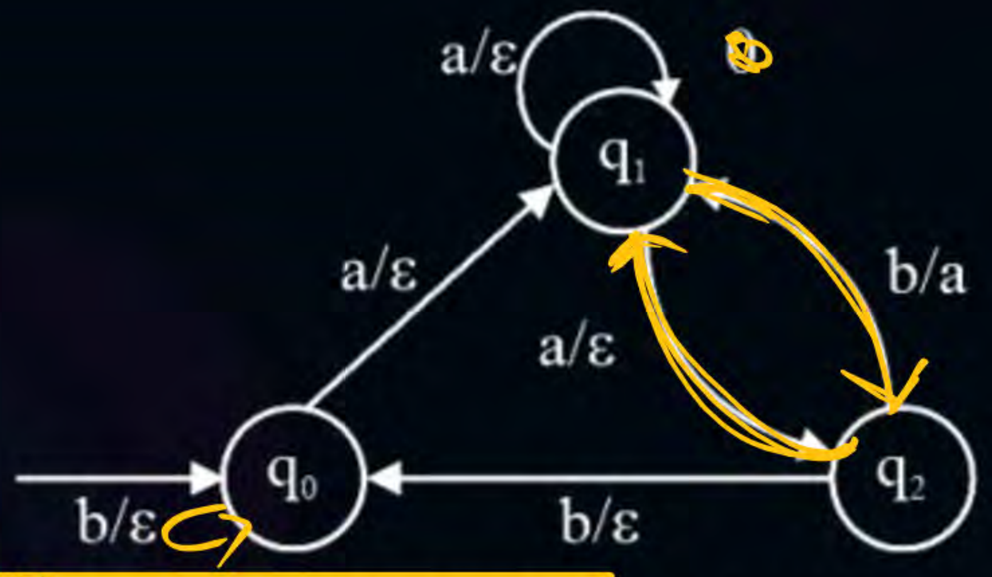
B. Outputs a "01" whenever the input sequence contain "11"

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

D. None of the above

[MCQ]

#Q. Consider the following finite state transducer where the label on an edge x/t denotes if the input is x, follow the arrow and emit t



For the input aabbbaaabbbbabaabb the output is :

A aaaa

B aaaaaaaaa

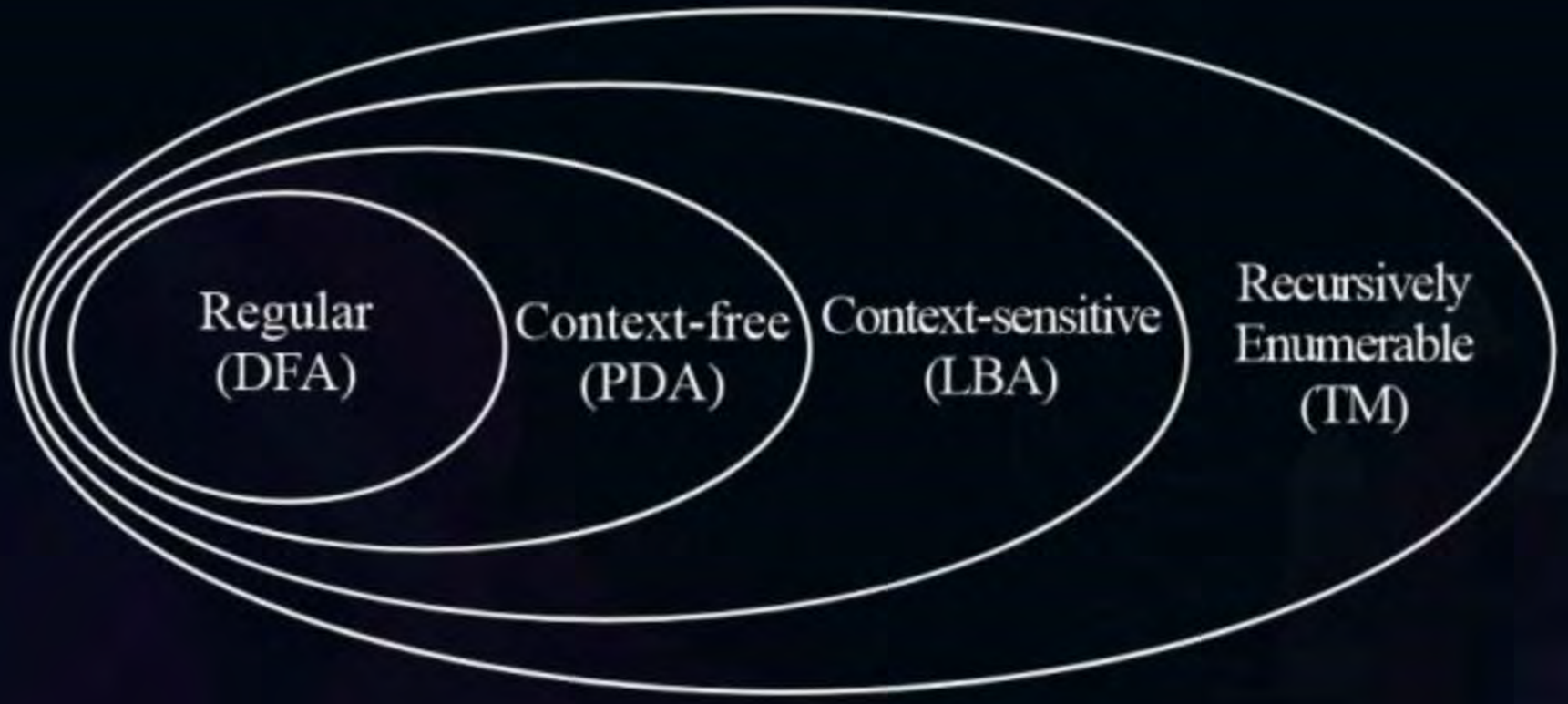
C ab ab ab ab

D abbbabbbbabababb

[NAT]

#Q. Construct an equivalent Moore machine for the following mealy machine.

Topic : Regular expression



Topic : Grammar

- Set of rules used to describe strings of a language is known as grammar.
- Formal definition of grammar is

G=(N, T, P, S)

- **N :-** non terminals or variables
- **T :-** Terminals
- P :- no. of productions
- **S :-** Starting symbol

Topic : Grammar

- For every language grammar exit & every grammar generates one language.
- All grammars is of a from α replacement of $\alpha \rightarrow \beta$, where β is replacement of α

Topic : Derivation

- The process of deriving strings from the given grammar known as derivation.
- The derivation can be either left most derivation or right most derivation

- **Left most derivation:**
  It is the derivation in which left most non terminal is replaced by its R.H.S part at every step.

- **Right most derivation:**
  It is a derivation in which right most non terminal is replaced by its R.H.S part at every step.

Derivation Tree (or)Parse Tree

- Tree representation of the derivation is known as derivation tree.
- All leaf node of the parse tree is known as yield of parse tree .
- while reading yield from left to right sentence of the grammar can be generate.

Sentential form

- Each step in the derivation is one sentential form.
- Hence sentential form is combination of terminals & non terminals (sentence also can be included)
- If the derivation is left most then sentential form is left sentential form.
- If the derivation is right most then sentential is right sentential form
- Every grammar represents only one language but for one language more than one grammar may exist.
- For regular languages there exist a grammar known as regular grammar.

- Context free language there exist a grammar known as context free grammar.
- Context sensitive language there exist a grammar known as context sensitive grammar.
- For recursive enumerable language there exist a grammar known as unrestricted grammar.

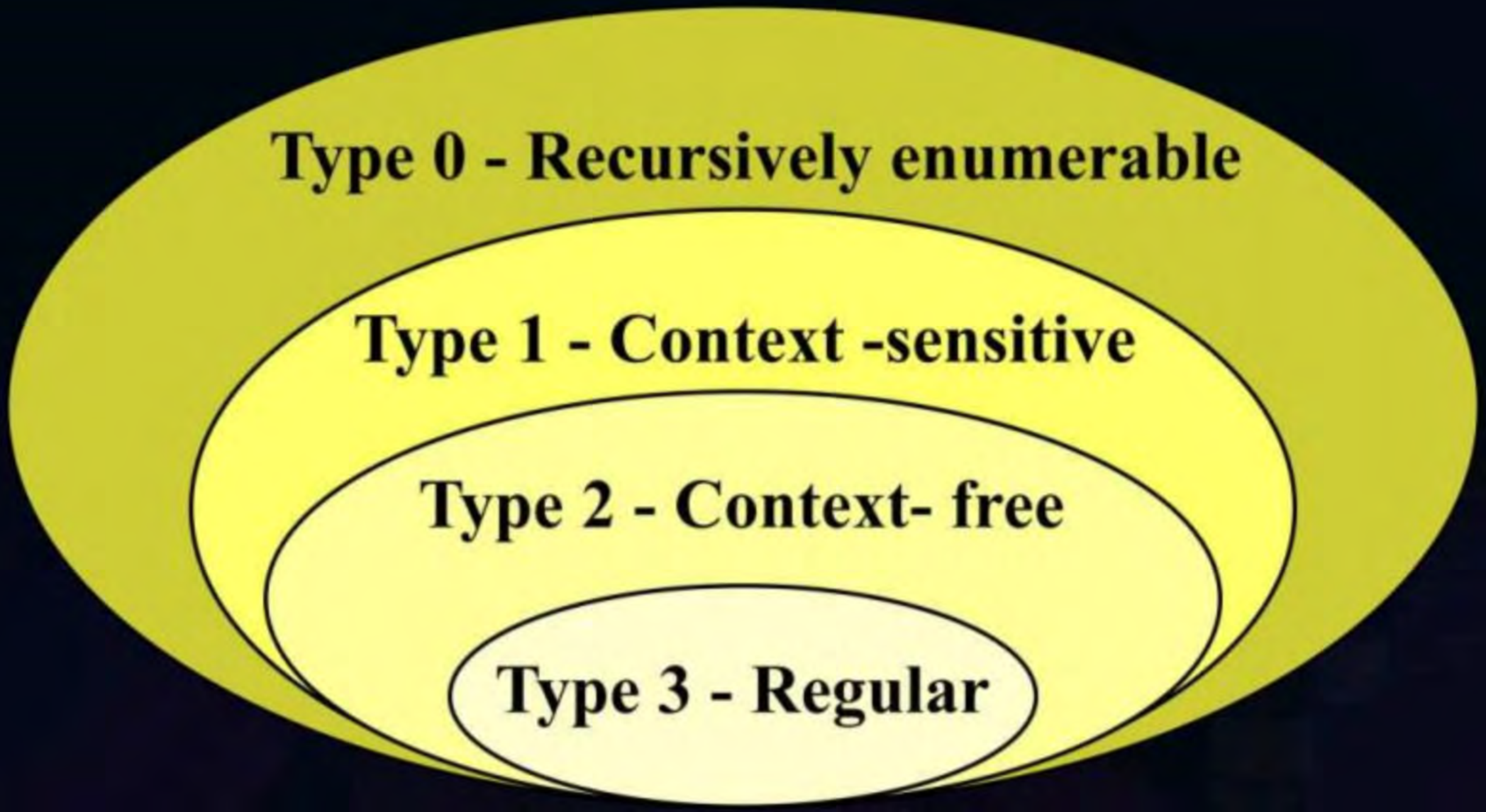
[NAT]

#Q. Identify language generated by following grammar.

[NAT]

#Q. Construct grammar for the following languages.

Type	Language (Grammers)	From of Productions	Accepting Dived
3	Regular	$A \rightarrow aB$, $A \rightarrow A$	Finite Automaton
2	Context –free	$A \rightarrow \alpha$	Pushdown Automaton
1	Context sensitive	$A \rightarrow \beta$ With $\lvert\alpha\rvert \geq \lvert\beta\rvert$	LBA
0	Unrestricted	$\alpha \rightarrow \beta$	Turing machine


Type 0 - Recursively enumerable
Type 1 - Context -sensitive
Type 2 - Context- free
Type 3 - Regular

10000

12 PM

12-2

THANK - YOU

CS & IT ENGINEERING

THEORY OF COMPUTATION

REGULAR EXPRESSION

Lecture No.- 02

By- Venkat sir

Recap of Previous Lecture

Topic MEALY MACHINE & Moore Machine

Topic ?? Regular Expression

Topic ?? Finite Automata ⇔ Regular Expression

Topic ?? Regular Grammar ⇔ Regular Expression

Topics to be Covered

Topic Regular Expressions

Topic ?? Constructions

Topic ??

Topic ??

Topic : Regular Expression

DFA possible

- The simplest way of representing a regular language is known as Regular expression.
- For every regular language regular expression can be constructed.
- To construct regular expression following 3 operators are used.
  - + is known as union operator
  - . is known as concatenation operator
  - * is known as Kleene closure operator

$+ = r_1 \overset{(OR)}{+} r_2 =$ Union

$\bullet = r_1 \overset{(and)}{\bullet} r_2 =$ Concatenation

$* = r_1^* =$ Kleene closure operator

Repeat $= r_1^0 + r_1^1 + r_1^2 + r_1^3 + r_1^4 + \cdots$

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

Dependency

DFA not possible

Regular Expression not possible.

Reg Lang

① $L_1 = \{\epsilon, a, b, aa, ab, bb, ba \ldots\} = \boxed{(a+b)^*}$

② $\Sigma = \{a, b\}$ where last two symbols are same $= (a+b)^*(aa+bb)$

Select * from

all

Reg Expr

DFA = Language = Regular Expression

set

Topic : NOTE

- For one regular language many number of regular expressions can be possible.

- One regular expression can generate only one regular language.

Reg Expre

$L_1 = \{\ \} \rightarrow \phi$

$L_2 = \{\epsilon\} \rightarrow \epsilon$

$L_3 = \{a\} \rightarrow a$

$L_4 = \{a, b\} \rightarrow a + b$

$L_5 = \{ab, ba\} \rightarrow \underline{ab} + \underline{ba}$

$L_6 = \{\epsilon, a, aa, aaa, a^4 \ldots\} = a^*$ (Kleene closure)

$L_7 = \{a, aa, a^3, a^4 \ldots\} = a^+$ (positive closure)

$L_8 = \{\epsilon, a, b, aa, ab, ba, bb \ldots\} = (a+b)^*$

$L_9 = \{a, b, aa, ab, ba, bb \ldots\} = (a+b)^+$

$L_{10} = \{a^n \cdot b^m \mid n, m \geq 0\} = a^* b^* \neq (a+b)^*$

$L_{11} = \{a^n b^m \mid n > m\} =$ not possible

$L_{12} = \{a^n b^m \mid n \neq m\} =$ not possible

$L_{13} = \{a^n b^m \mid n \geq 0, m \geq 1\} = a^* b^+$

$L_{14} = \{a^n b^m \mid n \geq 1, m \geq 0\} = a^+ b^*$

$L_{15} = \{a^n b^m \mid n \geq 1, m \geq 2\} = a^+ b\, b^+$

$b\, b^*$

$L_{16} = \{(a^2)^n \mid n \geq 0\} = \{\epsilon, a^2, a^4, a^6, a^8 \ldots\} = (aa)^*$

$L_{17} = \{(a^3)^n \mid n \geq 0\} = \{\epsilon, a^3, a^6, a^9 \ldots\} = (aaa)^*$

$L_{18} = \{a^{2n} \mid n \geq 0\} = \times$

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

abab

$L_{19} = \{ a^{n!} \mid n \geq 1 \} = \times$ not possible

$L_{20} = \{ a^n b^m \mid (n+m) \text{ is even} \}$ ✓

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

a^4

$\begin{Bmatrix} \underset{a}{\text{even}} + \underset{b}{\text{even}} \\ \text{odd} + \text{odd} \end{Bmatrix}$

$$L_{21} = \{a^n b^m \mid (n+m) \text{ is odd}\}$$

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

$a + b$

even + odd

odd + even

$a^0 b^0$
ϵ

$L_{22} = \{a^n b^m \mid n < m\}$ = not possible

$L_{23} = \{a^n b^m \mid (n > m) \text{ and } (n < m)\} = \{\ \} = \phi$

$L_{24} = \{a^n b^m \mid (n \geq m) \text{ and } (n \leq m)\} =$ not possible

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

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

$$L_{25} = \{a^n b^m \mid (n+m) \text{ is odd}\}$$

$$L_{25} = \{ a^n b^m \mid (n > m) \text{ (or) } (n < m) \} = \text{Depend}$$

$n \neq m$

not possible

$$(a+b)(a+b)$$

$$(a+b)^* = (a+b)^0 + (a+b)^1 + (a+b)^2 + (a+b)^3 + \cdots$$

$$= \epsilon + a + b + aa + ab + ba + bb + \cdots$$

$$\Sigma^* = \{\epsilon, a, b, aa, ab, bb, ba \cdots\} =$$

$$(ab)^* = (ab)^0 + (ab)^1 + (ab)^2 + (ab)^3 + \cdots$$

$$\epsilon + ab + abab + ababab + \cdots$$

#Q. Construct regular expression that generates set of all strings of a's and b's where length of each string is exactly 4.

$$\underset{2}{\underline{a|b}} \times \underset{2}{\underline{a|b}} \times \underset{2}{\underline{a|b}} \times \underset{2}{\underline{a|b}} = 16$$

$$(a+b)\cdot(a+b)\cdot(a+b)\cdot(a+b)$$

#Q. Construct regular expression that generates set of all strings of a's and b's where length of each string is atleast 4

$$(a+b)^4(a+b)^*$$

How many states in min DFA

5 States

How many states in min DFA that accepts

$\{(a+b+\epsilon)^6\} \rightarrow$ length atmost 6 $\}$ 8 states

#Q. Construct regular expression that generates set of all strings of a's and b's where length of each string is atmost 4.

$$\epsilon + (a+b)^1 + (a+b)^2 + (a+b)^3 + (a+b)^4$$

(or)

$$\boxed{(a+b+\epsilon)^4} = \underbrace{(a+b+\epsilon)(a+b+\epsilon)(a+b+\epsilon)(a+b+\epsilon)}_{\text{atmost } 4}$$

#Q. Construct regular expression that generates set of all strings of a's and b's where length of each string is divisible by 4.

$$\underbrace{\left[(a+b)(a+b)(a+b)(a+b)\right]^{4}}$$

$$\left[(a+b)^{4}\right]^{*}$$

$$\overbrace{\{(a+b)[(a+b)(a+b)]^*\}}^{\text{odd}} \quad \}\ \underline{\underline{\text{even}}}$$

DFA states ??

2 states

[MCQ]

#Q. Which of the following regular expressions are equivalent?

I. $(00)^*.(\varepsilon+0)$ → {even, odd} = all 0s

II. $(00)^*$ → even 0s

III. 0^* → all 0s

IV. $0(00)^*$ → odd 0s

A (I) And (II)

B (ii) and (iii)

C (i) And (iii)

D (iii) and (iv)

THANK - YOU

CS & IT ENGINEERING

THEORY OF COMPUTATION

REGULAR EXPRESSION

Lecture No.- 03

By- Venkat sir

Recap of Previous Lecture

Topic

- Regular Expressions
- ?????
- Regular Expression Construction
  - $+ \longrightarrow$ Union op
  - $\cdot \longrightarrow$ Concatenation op
  - $* \longrightarrow$ Kleene closure op

Topics to be Covered

Topic Conversion from $\in$-NFA to NFA

Topic ??

Topic ??

Topic ??

Topic : Regular Expression

DFA possible

- The simplest way of representing a regular language is known as Regular expression.
- For every regular language regular expression can be constructed.
- To construct regular expression following 3 operators are used.
- + is known as union operator → OR → $r_1 + r_2 \Rightarrow a+b = \{a,b\}$
- . is known as concatenation operator → (and) → $a \cdot b \Rightarrow \{ab\}$
- * is known as Kleene closure operator → Repeat → $a^* \Rightarrow a^0 + a^1 + a^2 + a^3 + \ldots$

→ positive closure ⇒ $a^+ = a^1 + a^2 + a^3 + \ldots$

Topic : NOTE

- For one regular language many number of regular expressions can be possible.
- One regular expression can generate only one regular language.

#Q. Construct regular expression that generates set of all strings of a's and b's where length of each string is divisible by 4.

#Q. Construct regular expression that generates set of all strings of a's and b's where number of a's are exactly 4.

$b^* a b^* a b^* a b^* a b^*$ ✓

$b^* aaaa b^*$ ✗

How many states present in minimal DFA for given regular expression.

$$a^* \underset{1}{b} a^* \underset{2}{b} a^* \underset{3}{b} a^* \underset{4}{b} a^* \underset{5}{b} a^* \underset{6}{b} a^*$$

b's exactly 6

$\Downarrow$

$n+2$ 8 states

#Q. Construct regular expression that generates set of all strings of a's and b's where number of a's are atleast 3.

$$\underbrace{(a+b)^* a (a+b)^* a (a+b)^* a (a+b)^*}$$

#Q. Construct regular expression that generates set of all strings of a's and b's where number of a's are atmost 3.

$$b^*(a+\epsilon)\,b^*(a+\epsilon)\,b^*\,(a+\epsilon)\,b^*$$

How many states in minimal DFA for the following regular expression

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

atmost 4 b's $\Rightarrow (n+2)$

6 states

#Q. Construct regular expression that generates set of all strings of a's and b's where number of a's are atleast 3.

bbbbbbbbb

#Q. Construct regular expression that generates set of all strings of a's and b's where number of a's are divisible by 3.

$$\left[b^* a b^* a b^* a b^*\right]^* + b^*$$

#Q. Construct regular expression that generates set of all strings of a's and b's where having substring aab.

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

min DFA states ?

$(n+1)$ states

Home Work

#Q. Construct regular expression that generates set of all strings of a's and b's where having substring aba or bab.

Home work

#Q. Construct regular expression that generates set of all strings of a's and b's where 4th input symbol is a from left side.

#Q. Construct regular expression that generates set of all strings of a's and b's where 4th input symbol is b from end. (R.H.S)

$\begin{pmatrix} a \\ b \end{pmatrix}$ b a/b a/b a/b

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

{16 states}

How many states in min DFA?

#Q. Construct regular expression that generates set of all even length palindrome strings over {a}.

$\{\epsilon, a^2, a^4, a^6, a^8 ----\}$

$\Downarrow$

$\boxed{(aa)^*}$

madam

nitin

naman

malayalam

#Q. Construct regular expression that generates set of all even length palindrome strings over {a, b}.

$L = \{\epsilon, aa, bb, abba, baab, aaaa \ldots$
$abaaba, abbbba \ldots\}$

#Q. Construct regular expression that generates set of all odd length palindrome strings over {a, b}.

$L = \{a, b, aaa, aba, bab ---\}$

#Q. Construct regular expression that generates set of all odd length palindrome strings of English language. $\{a, b, c \ldots z\}$

ϕ

NO

① $\{a\} + \{\epsilon\} = a + \epsilon \neq a$

② $a + \phi = a$

R → Reg Expr

$\{\ \} + \{b\}$ $\{\phi = \text{zero}\}$ $\epsilon = 1$

(1) $R + \phi = \phi + R = R$

(2) $R \bullet \phi = \phi \bullet R = \phi$

(3) $R + \epsilon = \epsilon + R \neq R$

$\{\epsilon \mid R\}$

(4) $R \bullet \epsilon = \epsilon \cdot R = R$

(5) $(R^*)^* = (R^*)^+ = (R^+)^* = R^*$

(6) $R \cdot R^* = R^+ = R^* R$

1, 1, 1, 1, 1

$\epsilon \cdot \epsilon \cdot \epsilon \cdot \epsilon \cdot \epsilon$

(7) $\epsilon^* = \epsilon$

(8) $\epsilon^+ = \epsilon$ $\epsilon + \epsilon + \epsilon$

(9) $\phi^* = \epsilon$ $\epsilon + \phi^+ = \epsilon$

(10) $\phi^+ = \phi$ $\phi + \phi + \phi + \phi + \phi$

(11) $(a + b)^* \neq a^* b^* \neq (ab)^*$ ba

(12) $(a + b)^* \neq (a^* b^*)^*$

(13) $a(ba)^* = (ab)^* a$

$\{a, aba, ababa \ldots\} = a, ababa, aba \ldots$

$$R^{+} + \epsilon = R^{*}$$

$$(a^*)^* = a^*$$

$$(a^*)^+ = a^*$$

$$(a^+)^* = a^*$$

$$\underbrace{R\,R^{*}}_{R^{+}} = R.\{\epsilon, R, R^{2}, R_{3}, R_{4} \cdots\}$$

$$\{R, R^{2}, R^{3}, R^{4}, \cdots\}$$

$$\Downarrow$$

$$R^{+}$$

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

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

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

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

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

$a^* b^* a^* b^* a^* b^* a^* b^*$

(15) $a^* + a^* = a^* = a^* a^* = (a^*)^* = (a^*)^+ = (a^+)^*$

(16) $a + b = b + a$

$\{a, b\} =$

(17) $a \cdot b \neq ba$

$a^* b^* \neq (ab)^*$

ababab

[MCQ]

#Q. Identify language accepted by following regular expression

$b^*(a^*.\phi\,.\,b + ab + a\phi^*b^*)(b + \phi)^*$

$b^*((\phi + ab + ab^*)b^*$

A. Exactly one a

B. At least one a

C. At most one a

D. None

[MCQ]

#Q. Which of the following regular expressions are equivalent?

I. $(00)^* (\varepsilon+0)$

II. $(00)^*$

III. 0^*

IV. $0(00)^*$

A. (I) And (II)

B. (ii) and (iii)

C. (i) And (iii)

D. (iii) and (iv)

[MCQ]

#Q. Which of the following pair of regular expressions are not equal

A. $(r^*)^*$ and $(r+)^*$

B. $(r + \varepsilon)^*$ and r^*

C. $(rr + \varepsilon)^*$ and r^*

D. None of the above

THANK - YOU

CS & IT ENGINEERING

THEORY OF COMPUTATION

REGULAR EXPRESSION

Lecture No.- 04

By- Venkat sir

Recap of Previous Lecture

Topic

- Regular Expression
- ?????
- Construction

Properties of Regular expression

Topics to be Covered

Topic Properties of Regular expression

Topic ??

Topic ??

Topic ??

F.A and Regular expression

#Q. How many states are there in minimal DFA that accept following regular expression.

Regular Expressions		DFA states?
① $(a+b)^5$	$\xrightarrow{n+2}$	7
② $(a+b+\epsilon)^3$	$\xrightarrow{n+2}$	5
③ $b^*ab^*ab^*ab^*ab^*$	$\xrightarrow{n+2}$	6
④ $b^*(a+\epsilon)b^*(a+\epsilon)b^*(a+\epsilon)b^*$	$\longrightarrow$	5
⑤ $(a+b)^*a(a+b)^*a(a+b)^*a(a+b)^*$	$\xrightarrow{n+1}$	4

#Q. How many states are there in minimal DFA that accept following regular expression.

Reg Expr

DFA

(6) $[(a+b)(a+b)(a+b)]^*$ $\xrightarrow{\text{div by } n}$ 3

(7) $(b^*ab^*ab^*ab^*)^*+b^*$ $\longrightarrow$ 3

(8) $(a+b)^*abab(a+b)^*$ $\xrightarrow{\text{Substring} \to n+1}$ 5

(9) $(a+b)^*a(a+b)(a+b)$ $\xrightarrow{2^n}$ 8

(10) $(a+b)(a+b)(a+b)b(a+b)^*$ $\xrightarrow{\text{last} \to (n+2)}$ 6

#Q. Construct regular expression that generates set of all strings of a's and b's where each string starting and ending with different symbol.

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

each string

#Q. Construct regular expression that generates set of all strings of a's and b's where having substring aba or bab.

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

#Q. Construct regular expression that generates set of all strings of a's and b's where 4th input symbol is a from left side.

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

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

(1) $R + \phi = \phi + R = R$

(2) $R \cdot \phi = \phi \cdot R = \phi$

(3) $R + \in = \in + R \neq R$

(4) $R \cdot \in = \in \cdot R = R$

(5) $(R^*)^* = (R^*)^+ = (R)^* = R^*$

(6) $R \cdot R^* = R^+ = R^* R$

$R\{\in, R, R^2, R^3\}$

(7) $\in^* = \in$

(8) $\in^+ = \in$

(9) $\phi^* = \in$

(10) $\phi^+ = \phi$

(11) $(a + b)^* \neq a^* b^*$

(12) $(a + b)^* \neq a^* b^*$

(13) $a(ab)^* = (ab)^* a$

ba

(14) $(a + b)^* = (a + b^{**})^*$

$= (a^* + b^{*})^*$

$= (a^{*} + b^{*})^*$

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

(15) $a^* + a^* = a^* = a^* a^*$

(16) $a + b = b + a$

(17) $a \cdot b \neq ba$

[MCQ]

#Q. Identify language accepted by following regular expression

$b^*(a^*.\phi\ .\ b + ab + a\phi^*b^*)(b + \phi)^*$

ϵ b

(1) $b^*(\phi + ab + ab^*)b^*$

A Exactly one a $b^*(ab + ab^*)b^*$

B At least one a $b^*(a(b + b^*))b^*$

C At most one a $b^*a(b^*)b^*$

D None

b^*ab^*

(Q) The regular expression $a^*b^*(a^*b^*)^*$ is equal to

(a) $(abab)^*$

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

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

(d) $(ab)^*$

$$\underbrace{\underbrace{a^*b^*}\,\underbrace{(a^*b^*)^*}}_{\boxed{a^*b^*}\,(a+b)^*}$$

$$\boxed{a^*b^*}\,(a+b)^* \Rightarrow (a+b)^*$$

{Complete language}

(Q) $R = \underbrace{\epsilon + 1^*(01)^* \left(\underset{a}{1}^* \underset{b}{(01)^*}\right)^*}$ } which of the following is equal to given regular expression

$1^*(01)^*(1+01)^*$

$\Downarrow$

$(1+01)^*$

(a) $(10+01)^*$

(b) $(1\,01)^*$

(c) $1^*(01)^*$

(d) $(1+01)^*$

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

$\Downarrow$

$\boxed{(a+b)^*}$

(1) Consider following 3 regular expressions

① $(a+ba)^*(b+\epsilon)$

② $\boxed{(a^*(ba)^*)^*}(b+\epsilon)$

ϵ

equal

$\boxed{(a+ba)^*(b+\epsilon)}$

③ $(a+ba)(a+ba)^*(b+\epsilon)$

$(a+ba)^+(b+\epsilon)$

$R \cdot R^* = R^+$

$a\,a^* = a^+$

(1) which of the following regular expression generates set of all strings not having 100 as sub string ($\Sigma = \{0,1\}$)

$\{\epsilon, 0, 1, 0, 11, 00, 11, 00, 111 \ldots\}$

(a) $0^*(1^*0)^* = 1^*0 \quad 0$

(b) $0^*1^*01^*$

(c) 0^*1010^*

100

(d) $0^*(10+1)^*$

Q. which of the following regular expressions are not equal?

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

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

(c) $(ab)^*$ and a^*b^*

(d) none

(Q)

ending with 11

$r_1 = 11(0+1)^*$

$r_2 = (0+1)^* 11$

$r_3 = 11(0+1)^* 11 + 111 + 11$

Which of the following is true?

(a) $r_1 = r_2 = r_3$

$r_1 \cup r_2$

$\{110, 011, 1100 \ldots\}$

(b) $r_1 \cup r_2 \neq r_3$

(c) $r_1 \cap r_2 = r_3$

{Common strings are starting with 11 and ending with 11}

(d) none

[MCQ]

#Q. Which of the following regular expressions are equivalent?

I. $(00)^* (\varepsilon+0)$

II. $(00)^*$

III. 0^*

IV. $0(00)^*$

A. (I) And (II)

B. (ii) and (iii)

C. (i) And (iii)

D. (iii) and (iv)

[MCQ]

#Q. Which of the following pair of regular expressions are not equal

A. $(r^*)^*$ and $(r+)^*$

B. $(r + \varepsilon)^*$ and r^*

C. $(rr + \varepsilon)^*$ and r^*

D. None of the above

THANK - YOU

CS & IT ENGINEERING

THEORY OF COMPUTATION

REGULAR EXPRESSION

Lecture No.- 05

By- Venkat sir

Recap of Previous Lecture

Topic ?????

Regular Expressions

properties

Equal

Simplification of R.E

Topics to be Covered

Topic Regular Expression

= and =

Topic ?? Finite Automata

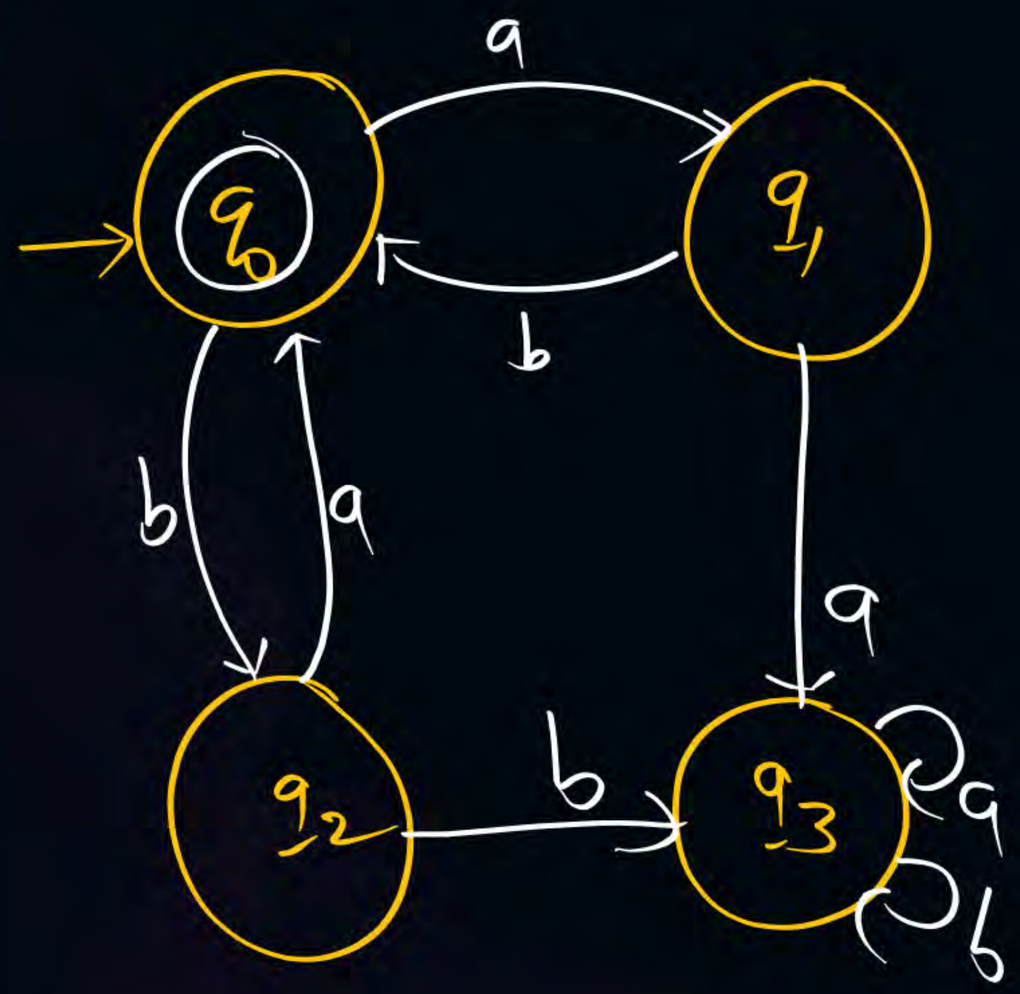
Topic ??

Topic ?? Regular grammar and Reg Expr.

① Arden's method

② State elimination method

FINITE AUTOMATA TO REGULAR EXPRESSION

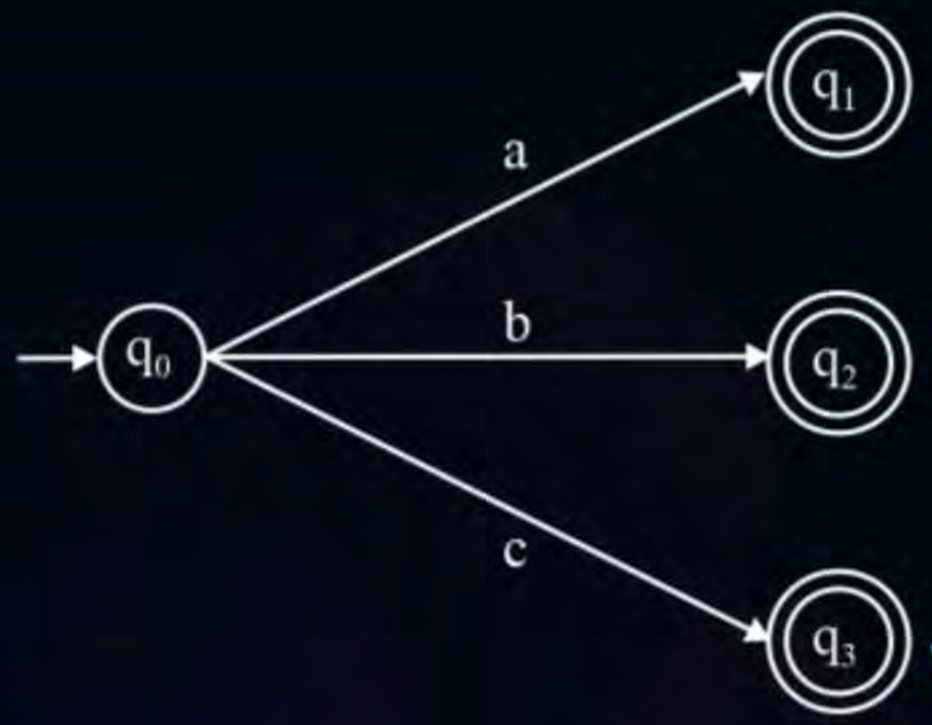


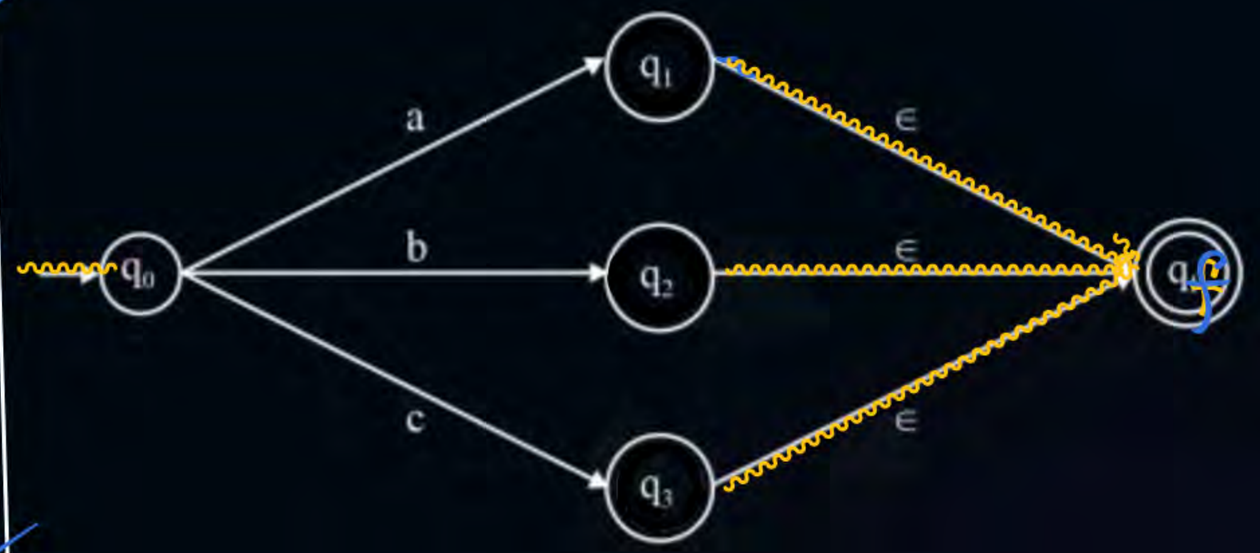
= Regular Expression

Target

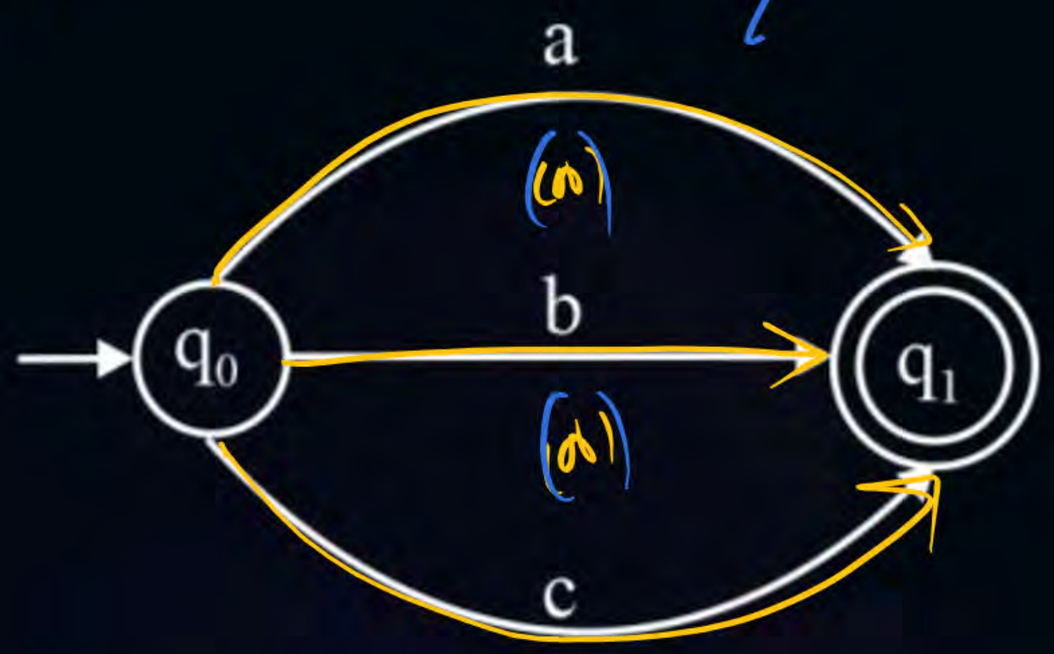
State elimination method

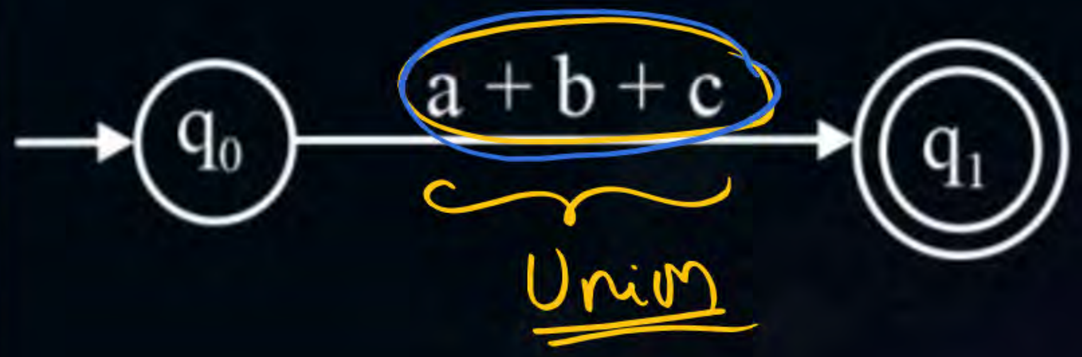
1.





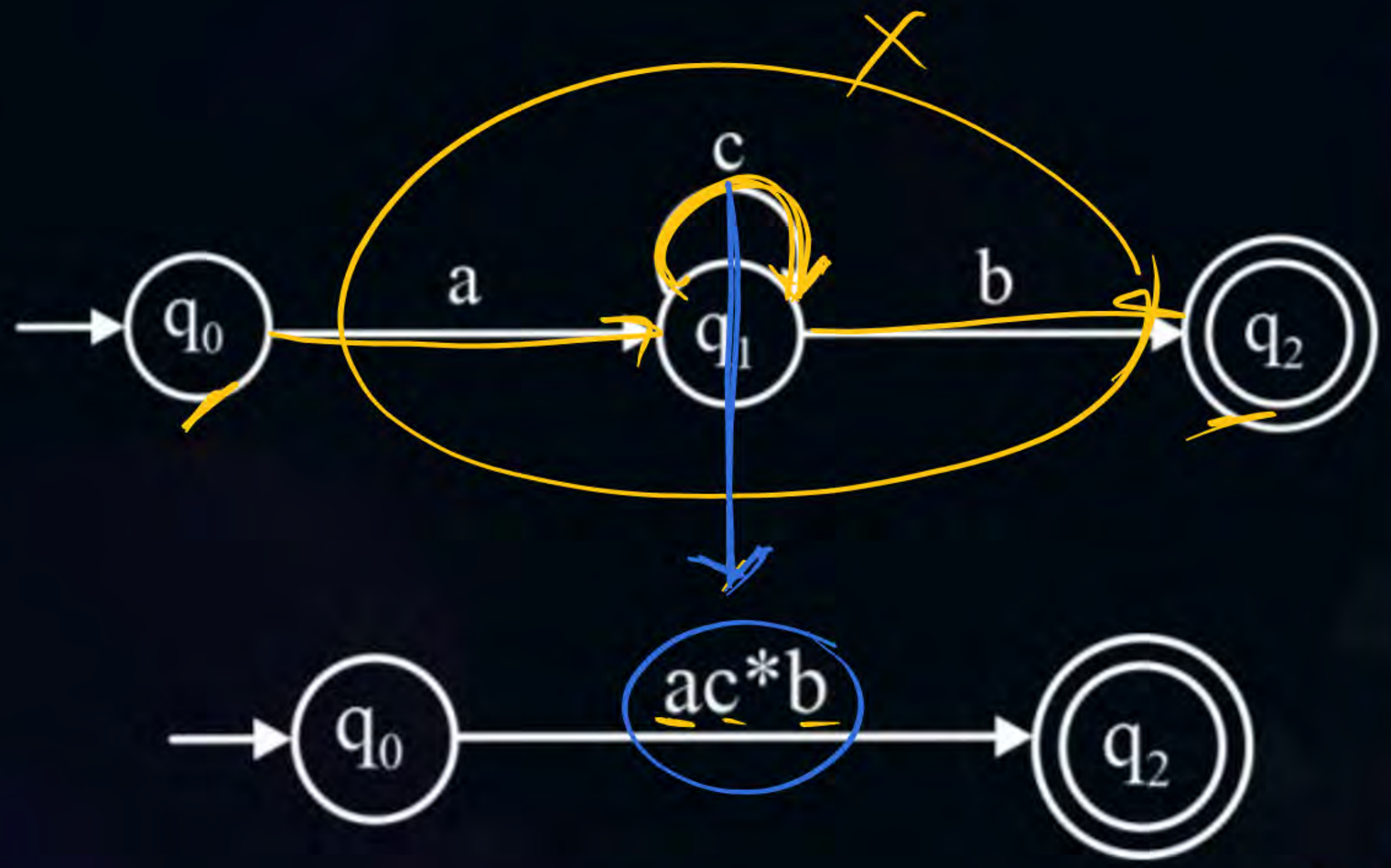
2.





3.

4.

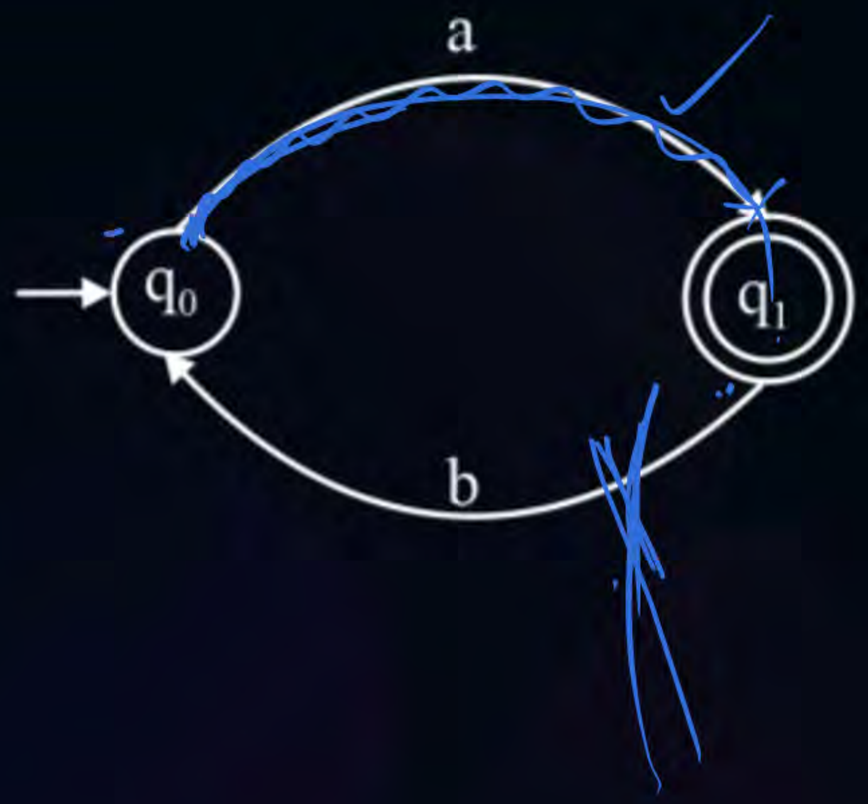


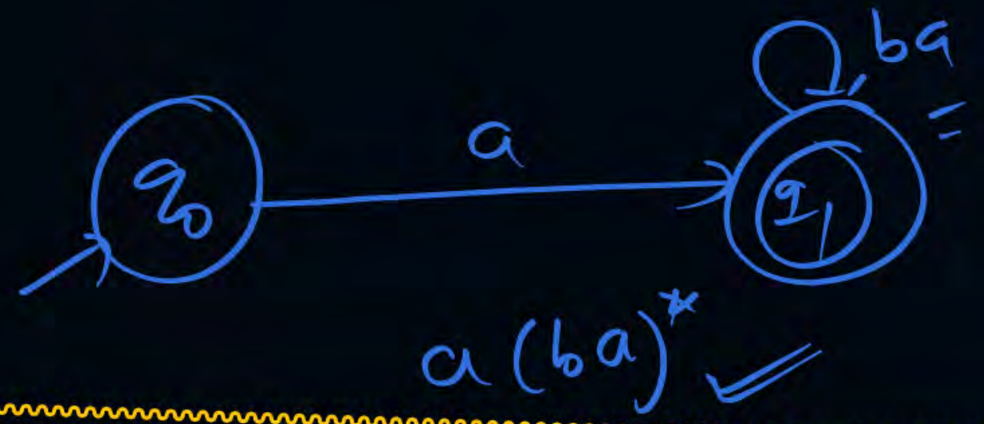
Selfloop => *

a^*b^*

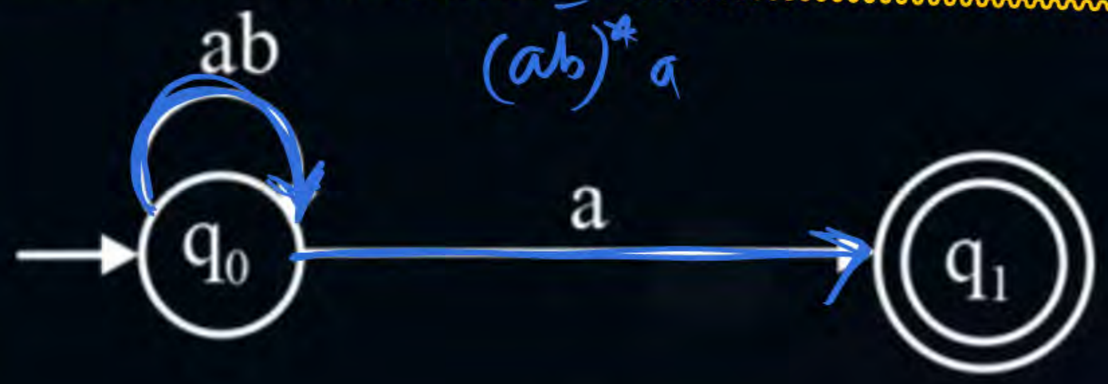
$(a+b)^*$

5. Reverse edge



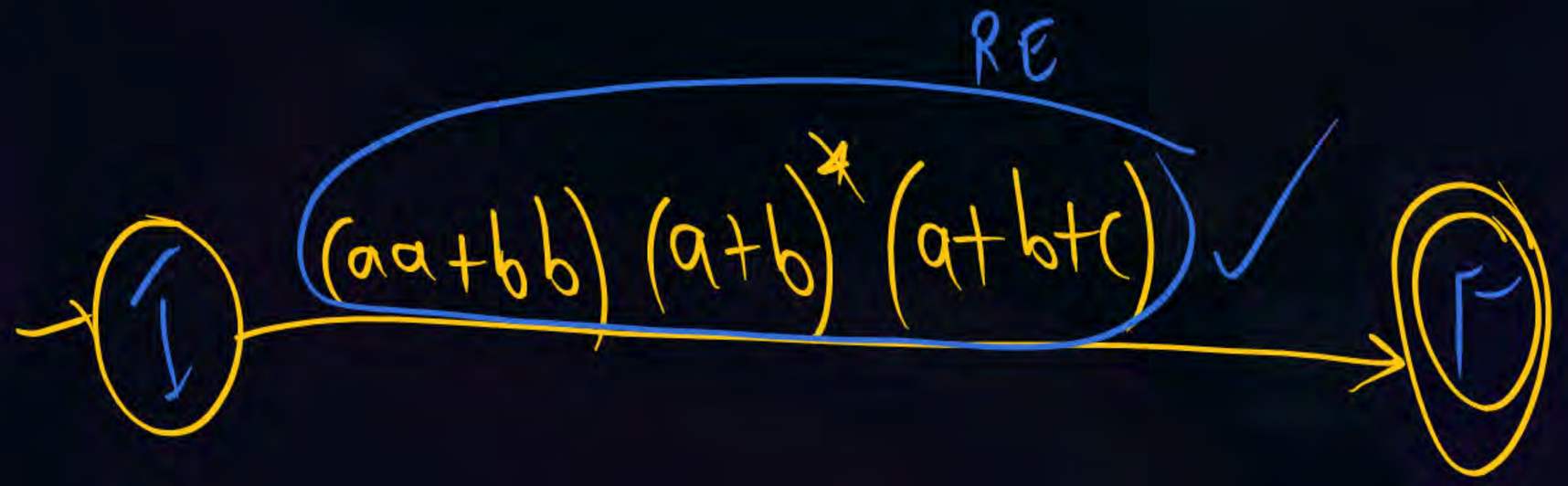


$a(ba)^*$



$(ab)^*a$

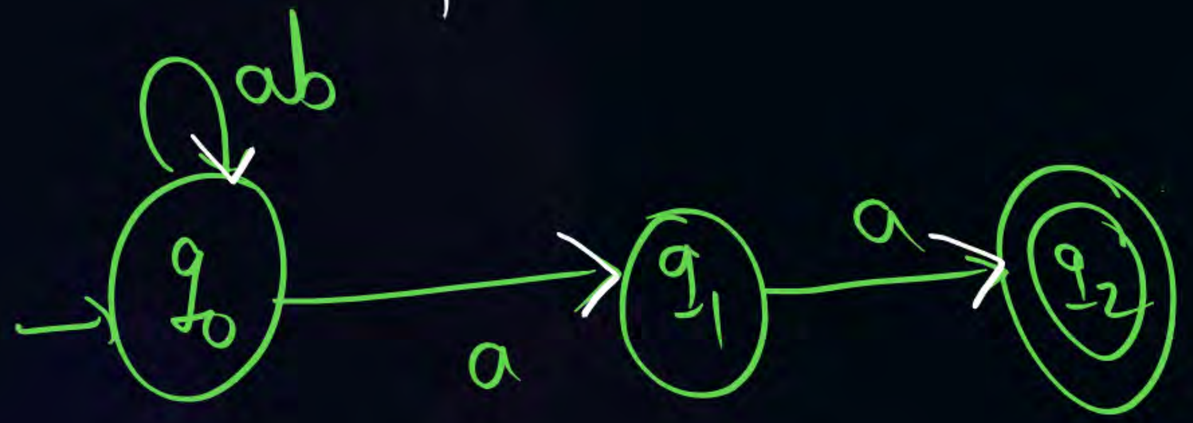
#Q. Construct Regular Expression for the following Finite Automata.



[NAT]

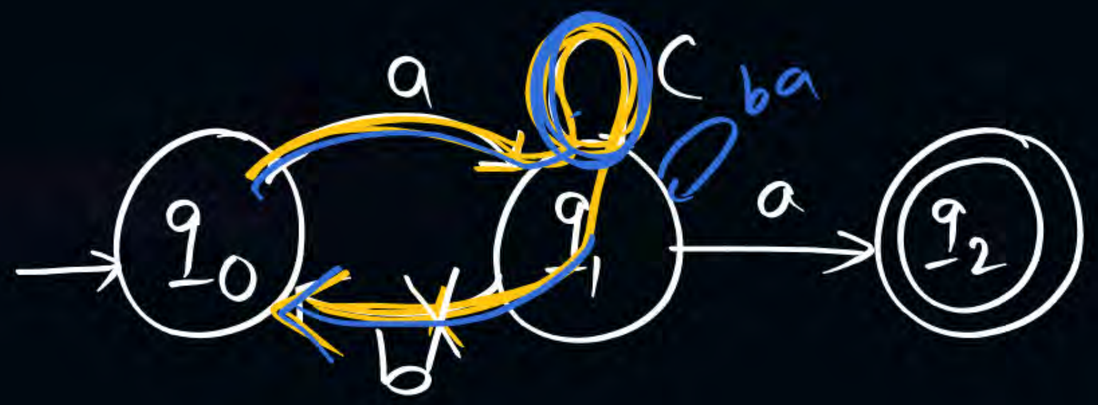
#Q. Construct Regular Expression for the following Finite Automata.

2



$(ab)^* aa$ ✓

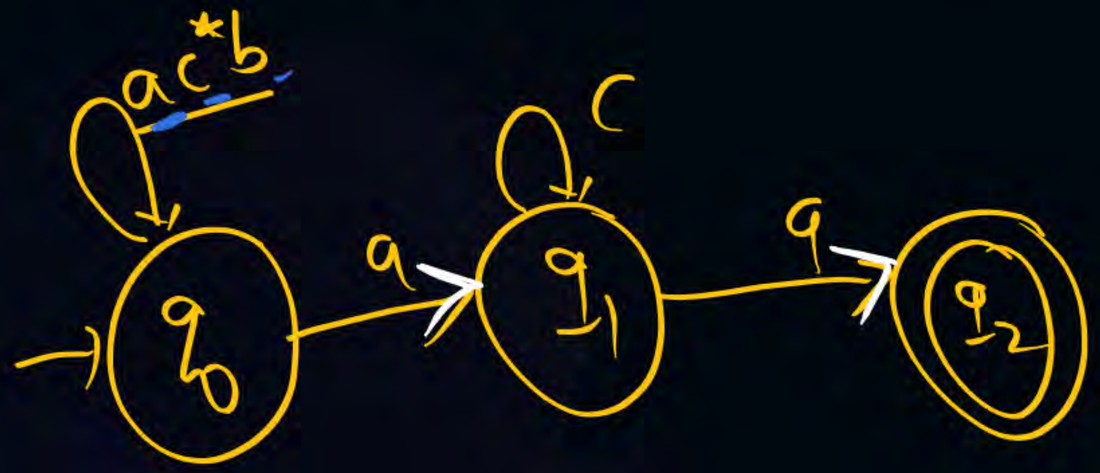
(3)



accbaa ✓

$(ab)^* a c^* a$

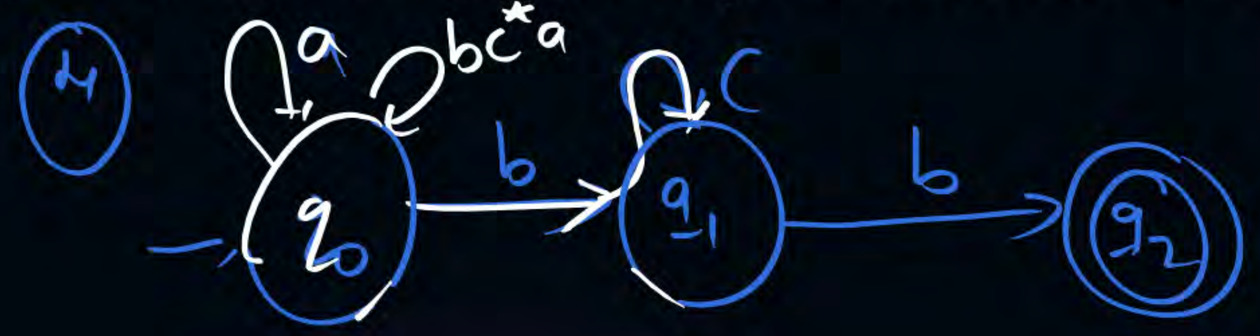
$= a(ba+c)^* a$



$(ac^*b)^* a c^* a$ ✓

[NAT]

#Q. Construct Regular Expression for the following Finite Automata.

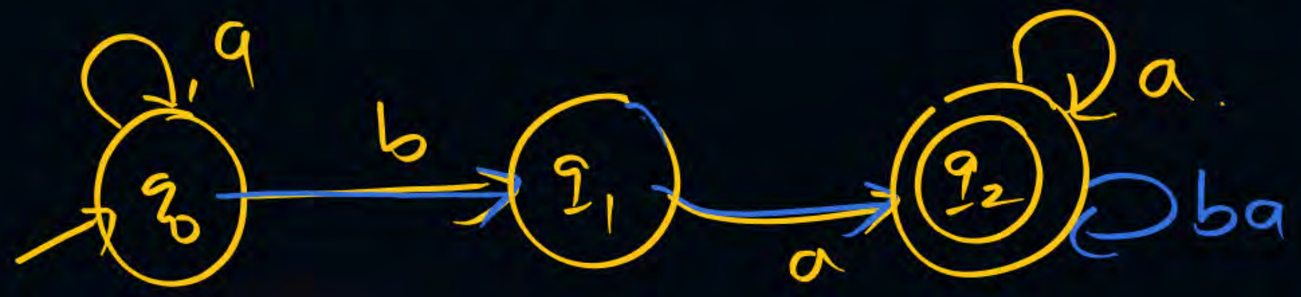


$$\boxed{(a + bc^*a)^* bc^*b}$$

[NAT]

#Q. Construct Regular Expression for the following Finite Automata.

5

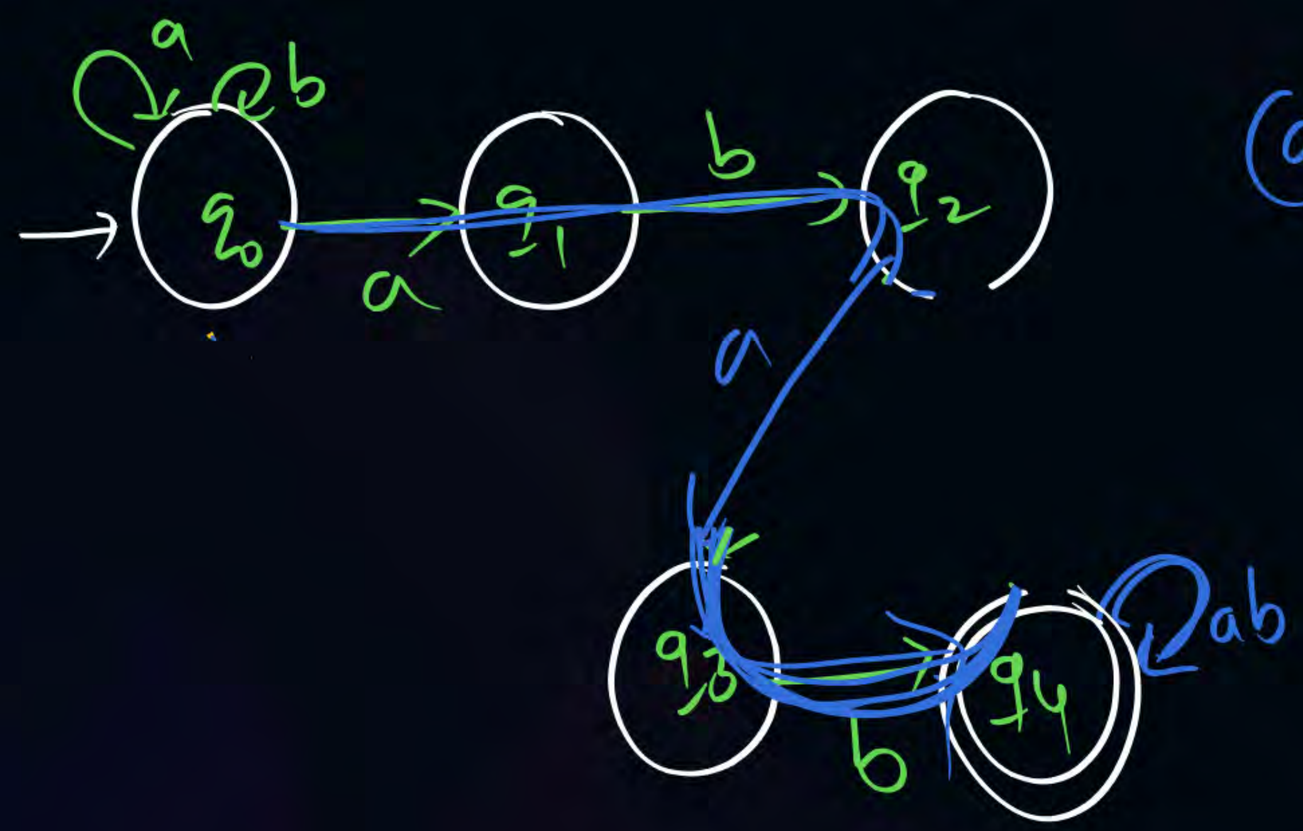


$$\{a^* ba(a+ba)^*\}$$

[NAT]

#Q. Construct Regular Expression for the following Finite Automata.

6



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

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

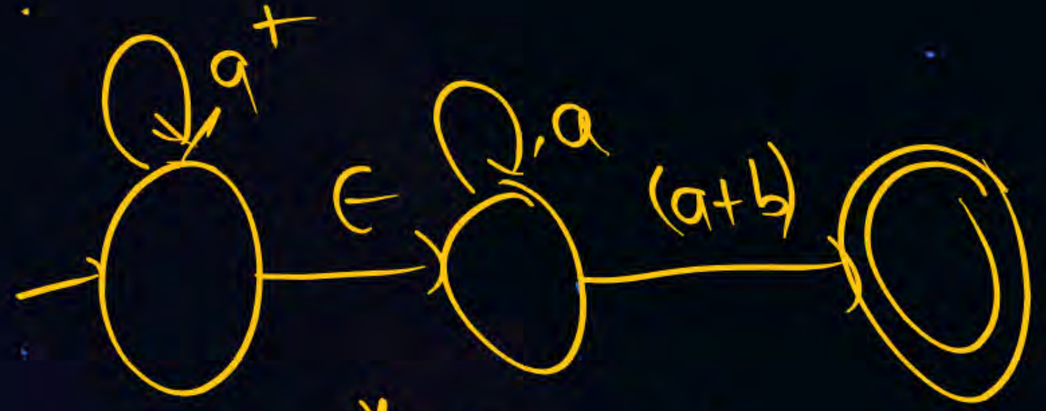
[NAT]

#Q. Construct Regular Expression for the following Finite Automata.

7

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

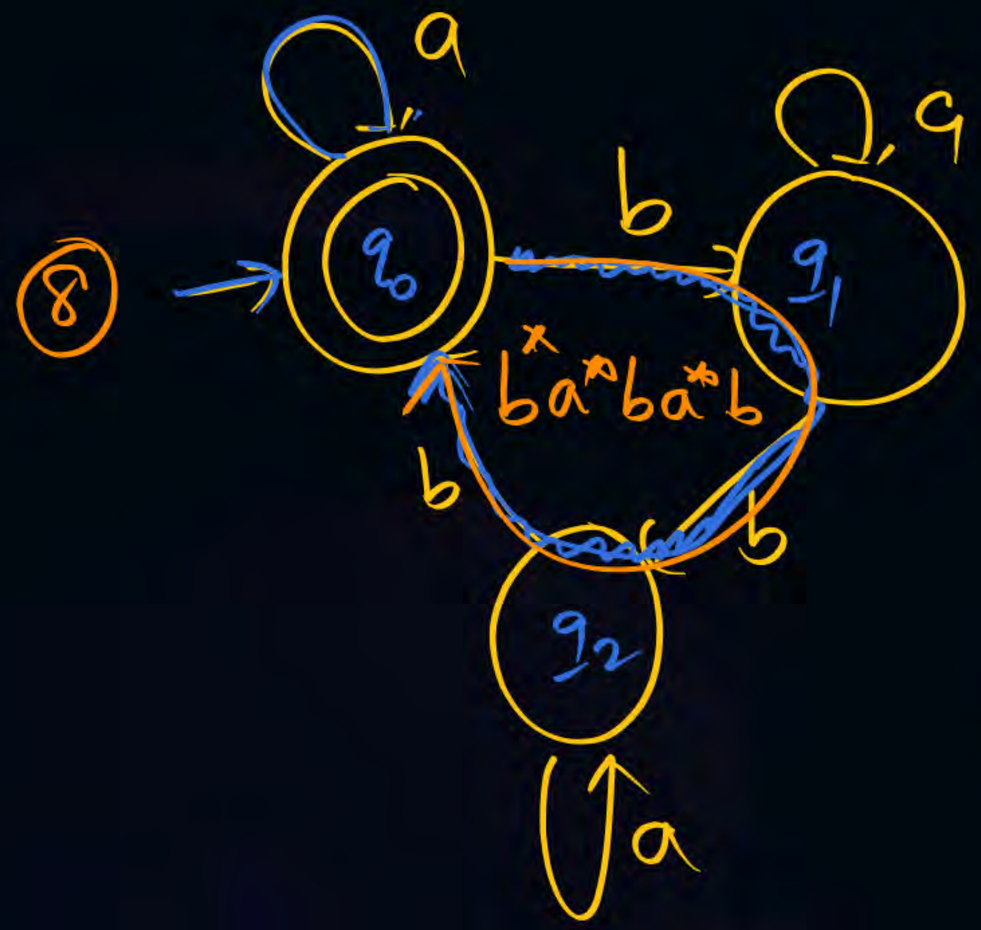
$\{a^*a^* = a^*\}$



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

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

a^*

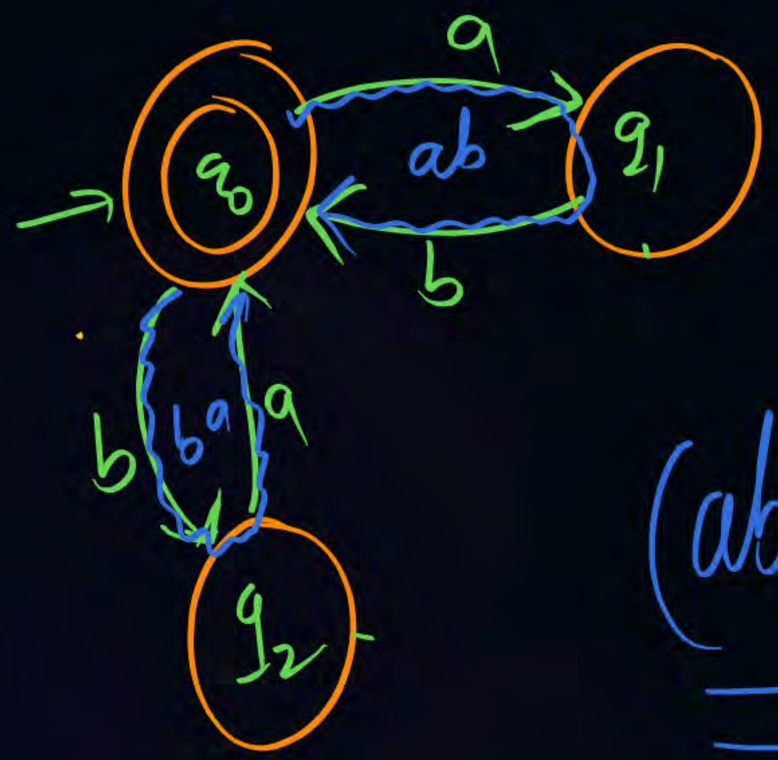


$$(a+ba^*ba^*b)^*$$

[NAT]

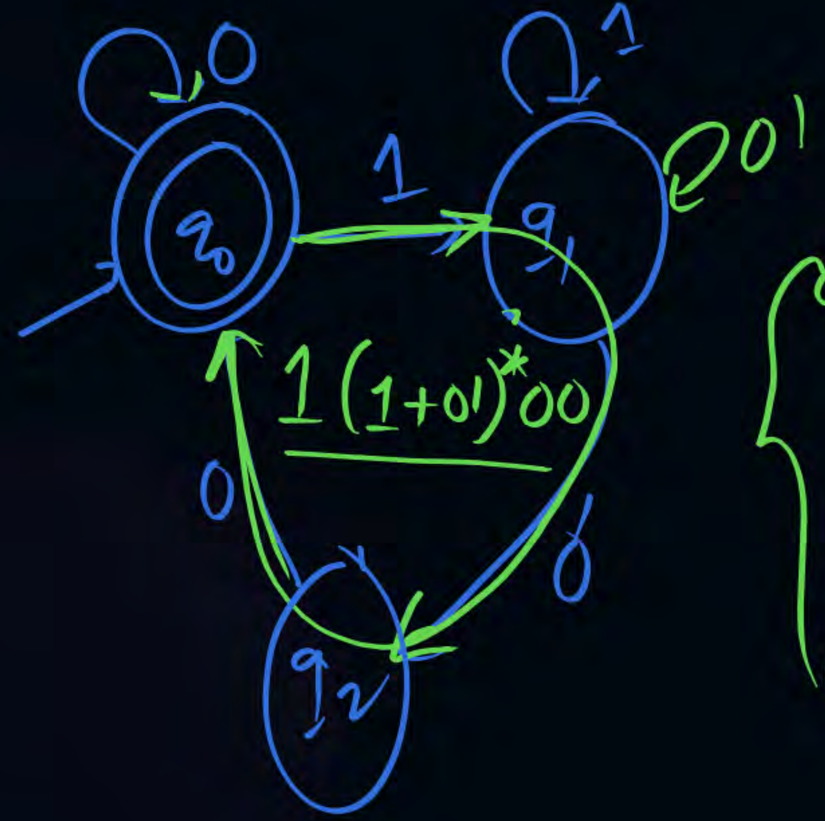
#Q. Construct Regular Expression for the following Finite Automata.

9



$$(ab+ba)^*$$

10

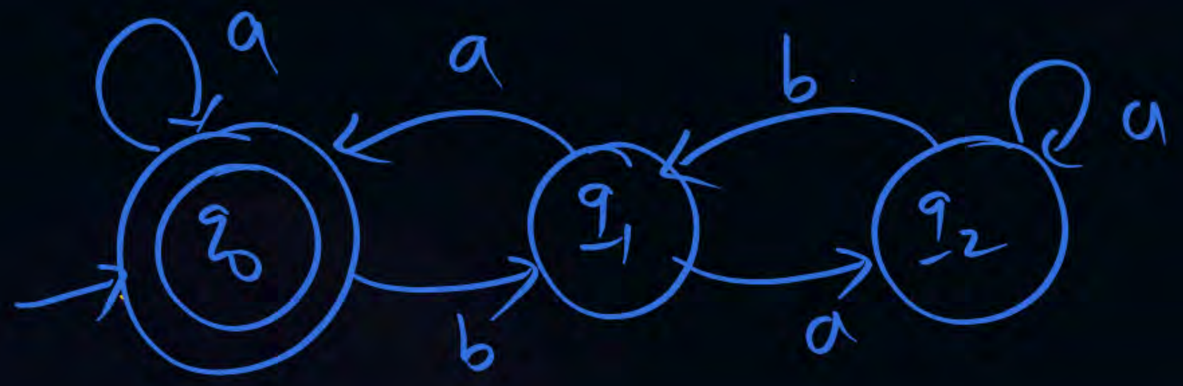


$$\left\{\left[0+1(1+01)^*00\right]^*\right\}$$

[NAT]

Home Work

#Q. Construct Regular Expression for the following Finite Automata.



THANK - YOU

CS & IT ENGINEERING

THEORY OF COMPUTATION

REGULAR EXPRESSION

Lecture No.- 06

By- Venkat sir

Recap of Previous Lecture

Topic

Fintite Automata → Regular expression

?????

Topics to be Covered

Topic Conversion from $\in$-NFA to NFA

Topic ??

Topic ??

Topic ??

DFA, NFA, ε-NFA ① State elimination method

② Arden's method

DFA, NFA

× ε-NFA

FINITE AUTOMATA TO REGULAR EXPRESSION

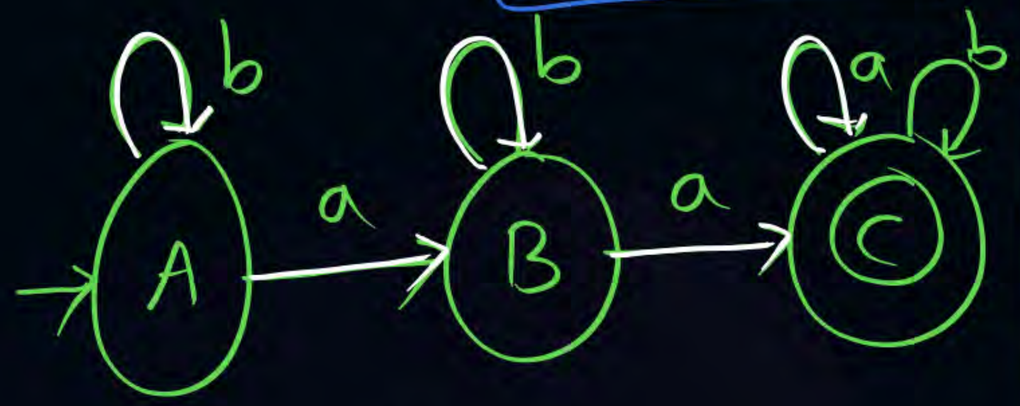
Arden's method

If P and Q are any two regular expressions then the equation $\boxed{R = RP + Q}$ has solution

solution $\Longrightarrow$ $\boxed{R = QP^*}$

① Construct regular expression using Arden's method for the following Finite Automata

$\boxed{b^* a b^* a (a+b)^*}$



① $A = Ab + \epsilon$

② $B = Aa + Bb$

③ $C = Ba + Ca + Cb$

① $A = Ab + \epsilon$

$A = \epsilon \cdot b^* = b^*$

② $B = Aa + Bb$

$B = Bb + Aa$

$B = Bb + b^* a$

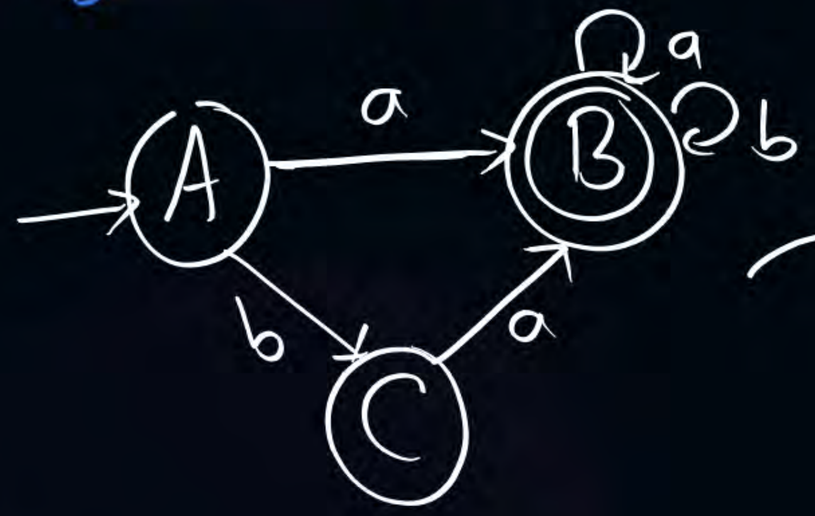
$B = b^* a b^*$

③ $C = Ca + Cb + Ba$

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

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

② Construct regular expression using Arden's method for the given Finite Automata



① $A = \epsilon$

② $B = Aa + Ca + Ba + Bb$

③ $C = Ab = b$

②

$B = Ba + Bb + a + ba$

$B = B(a+b) + (a+ba)$

$B = (a+ba)(a+b)^*$

Equivalence classes

① How many equivalence classes possible for the regular expression $a^* b^*$

Number of equivalence classes = No. of States of min DFA

(a) 1

(b) 2

(c) 3

(d) 4

$$\underbrace{\left[(a+b)(a+b)\right]^*}$$ = min DFA States?

$$\text{length}\left\{\frac{\text{Div by 2}}{\text{even}}\right\} = \text{2 States}$$

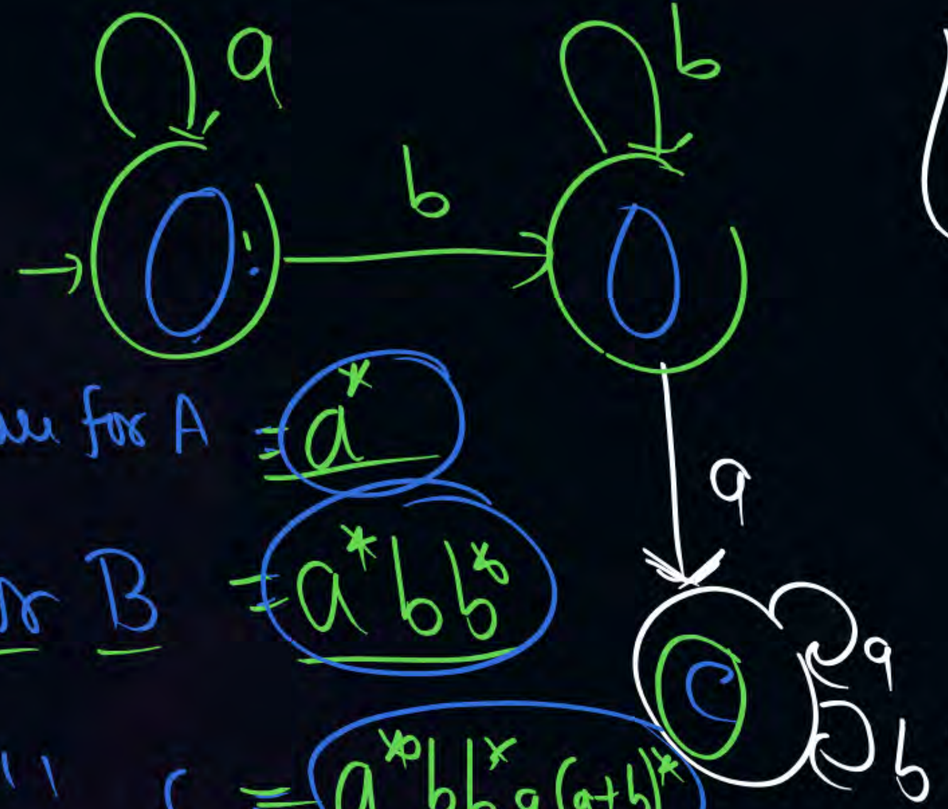
How many equivalence classes possible?

2 equivalence

ba

a^*b^* $\{\epsilon, aaaa, bbbb, ab \ldots\}$

$\{a^n b^m \mid n, m \geq 0\}$



3 states

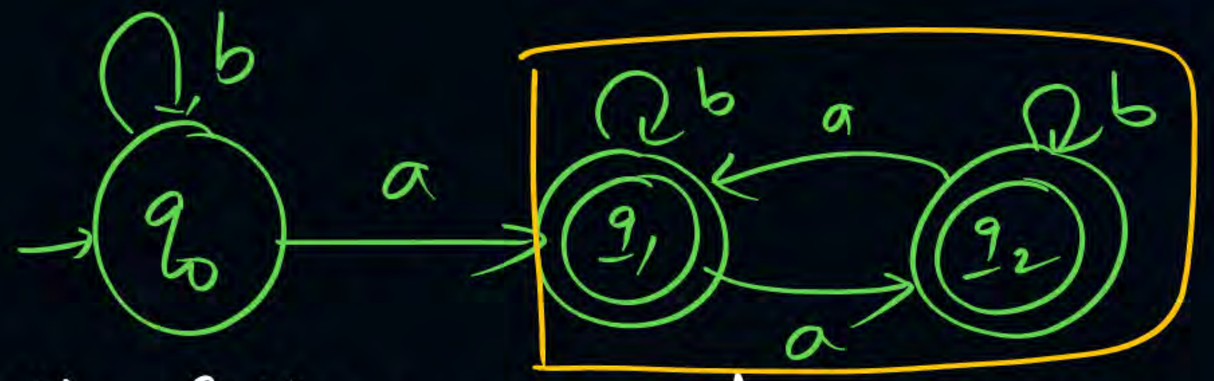
Equivalence class for A $= a^*$

Equivalence for B $= a^*bb^*$

" " C $= a^*bb^*a(a+b)^*$

Equivalence class

$a^* + a^*bb^*$



Which of the following regular expression is equal to given DFA?

(a) $(a+b)^+$ ✗

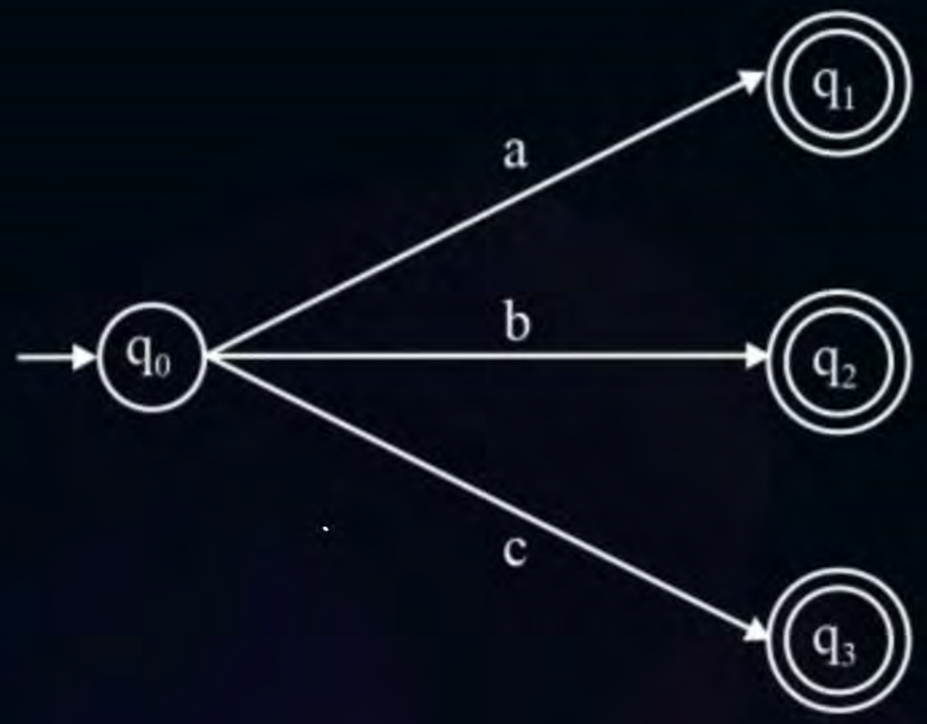
(b) $b^* a b^* a$ ✗

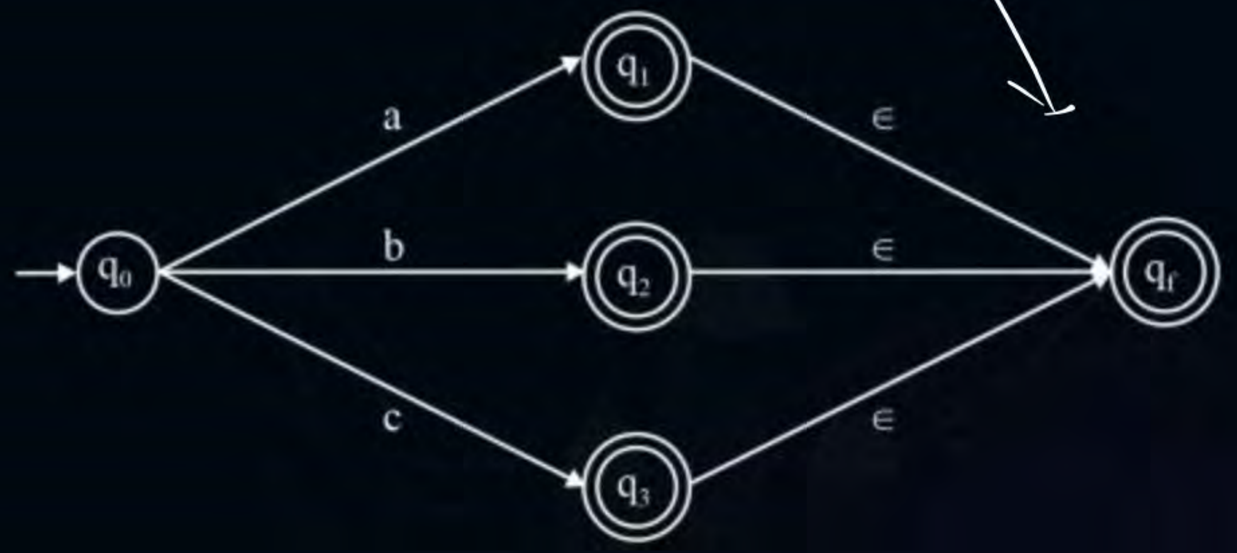
(c) $b^* a (a+b)^*$

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

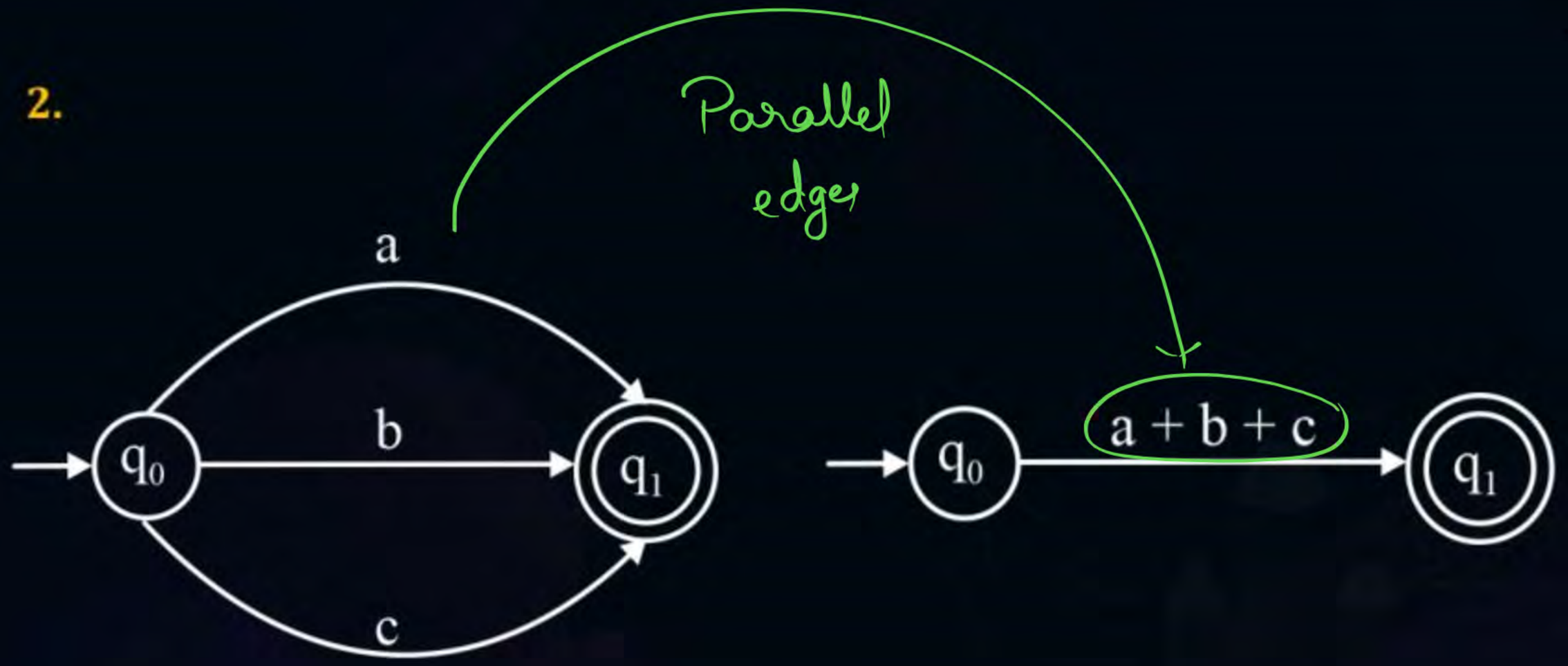
$b^* a (a+b)^*$

1.

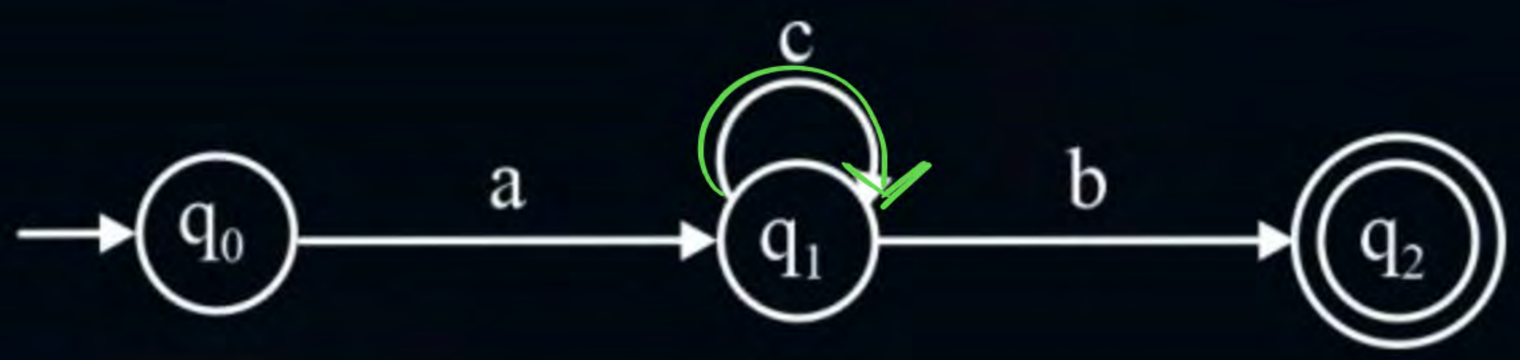


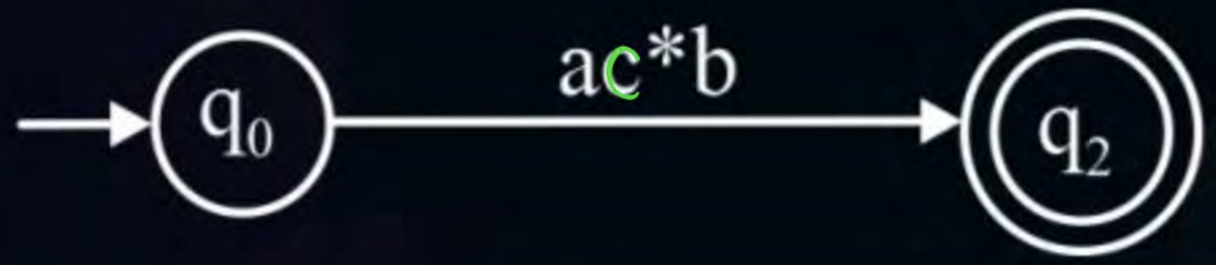


2.

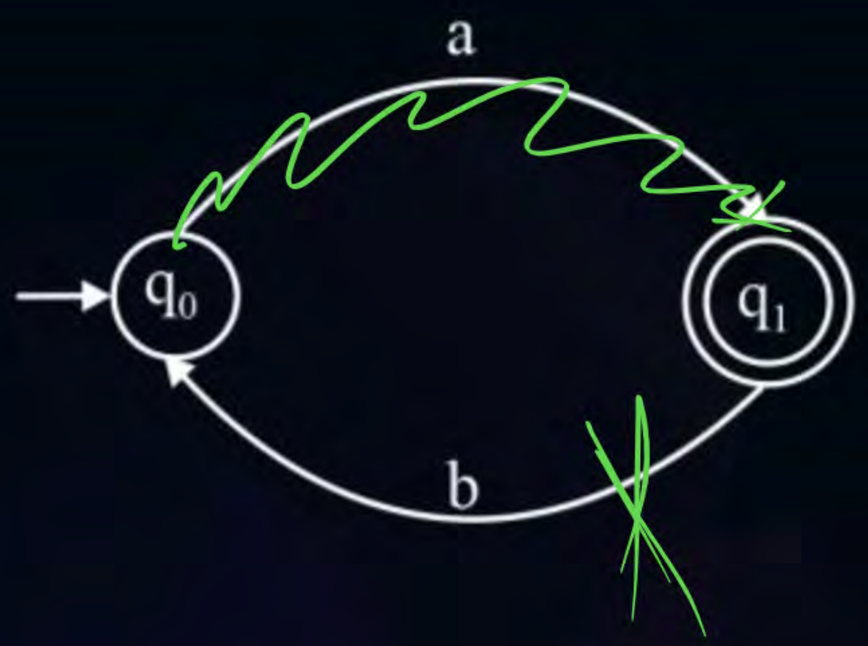


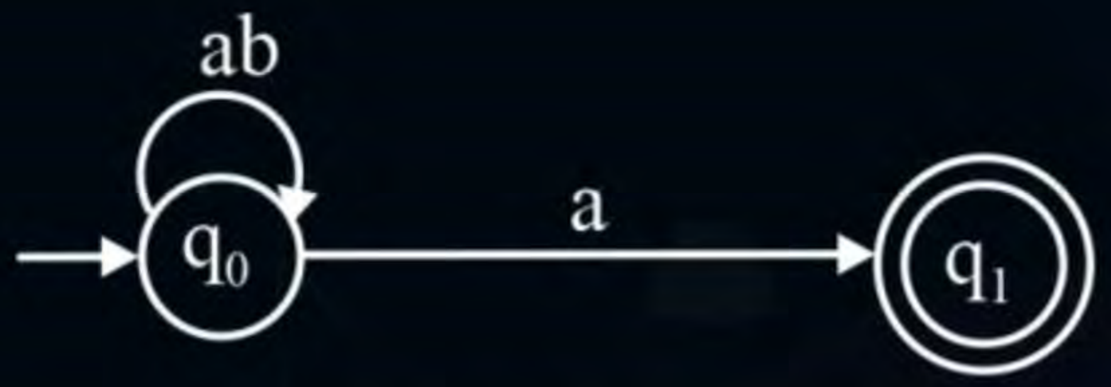
4.





5.

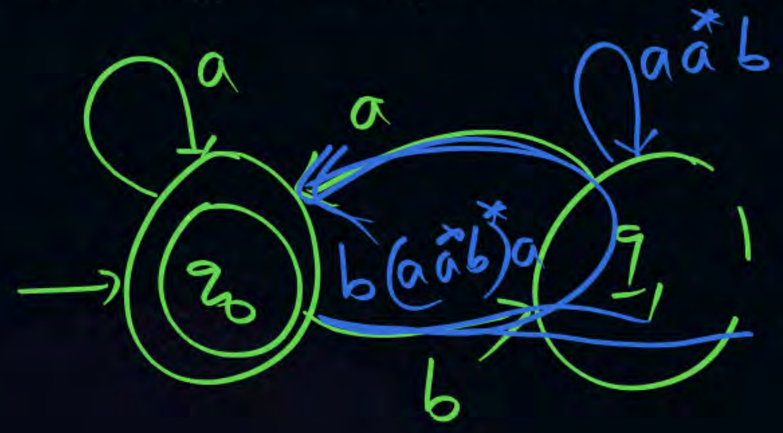




[NAT]

{ Home Work }

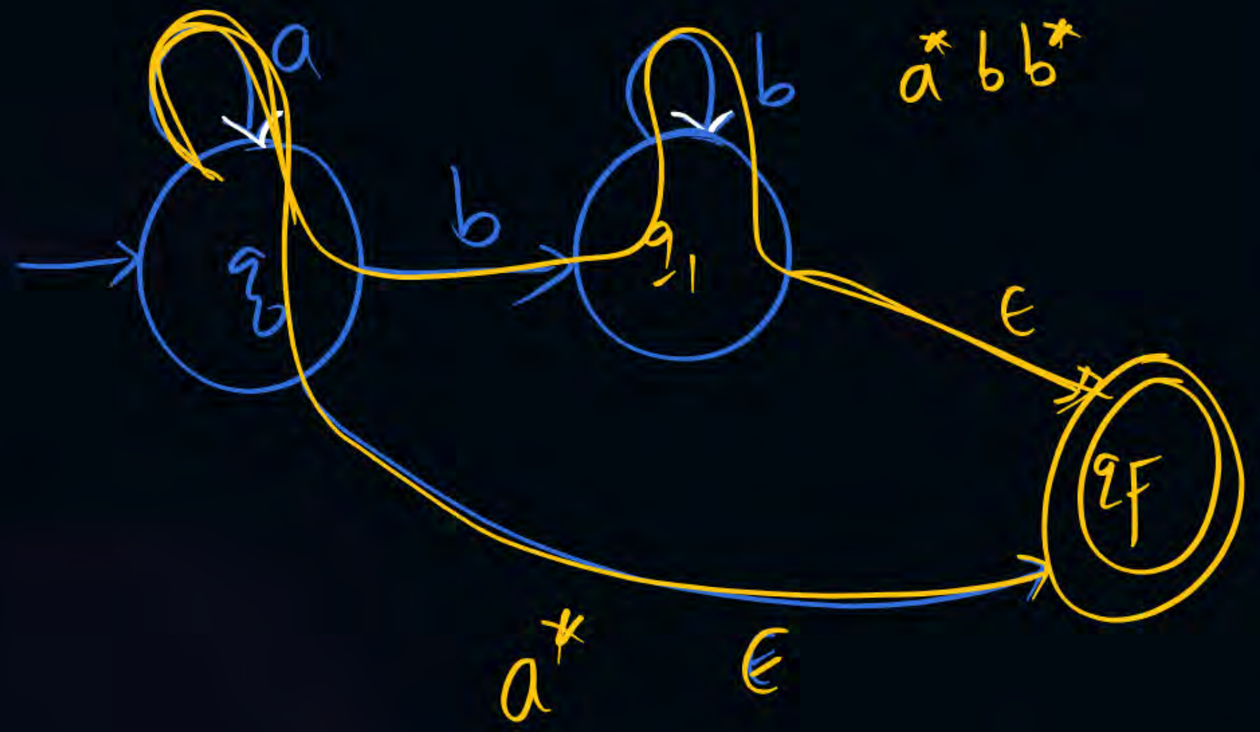
#Q. Construct Regular Expression for the following Finite Automata.



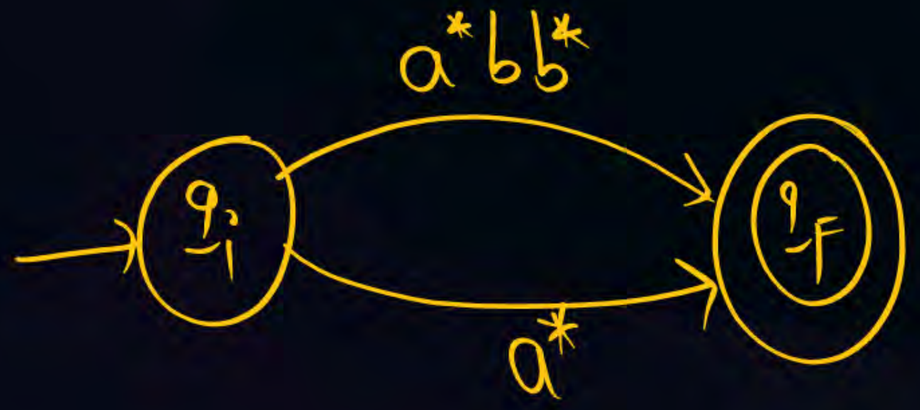
$$\left[a + b(\underbrace{a^+a^*b}_{})^*a\right]^*$$

$$\overline{\left[a + b(a^+b)^*a\right]^*}$$

①



a^*bb^*



$= a^*b^+ + a^*$

$= a^*(b^+ + \epsilon)$

$= a^*b^*$

Construct regular expression ?

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

$a^*(\epsilon + bb^*)$

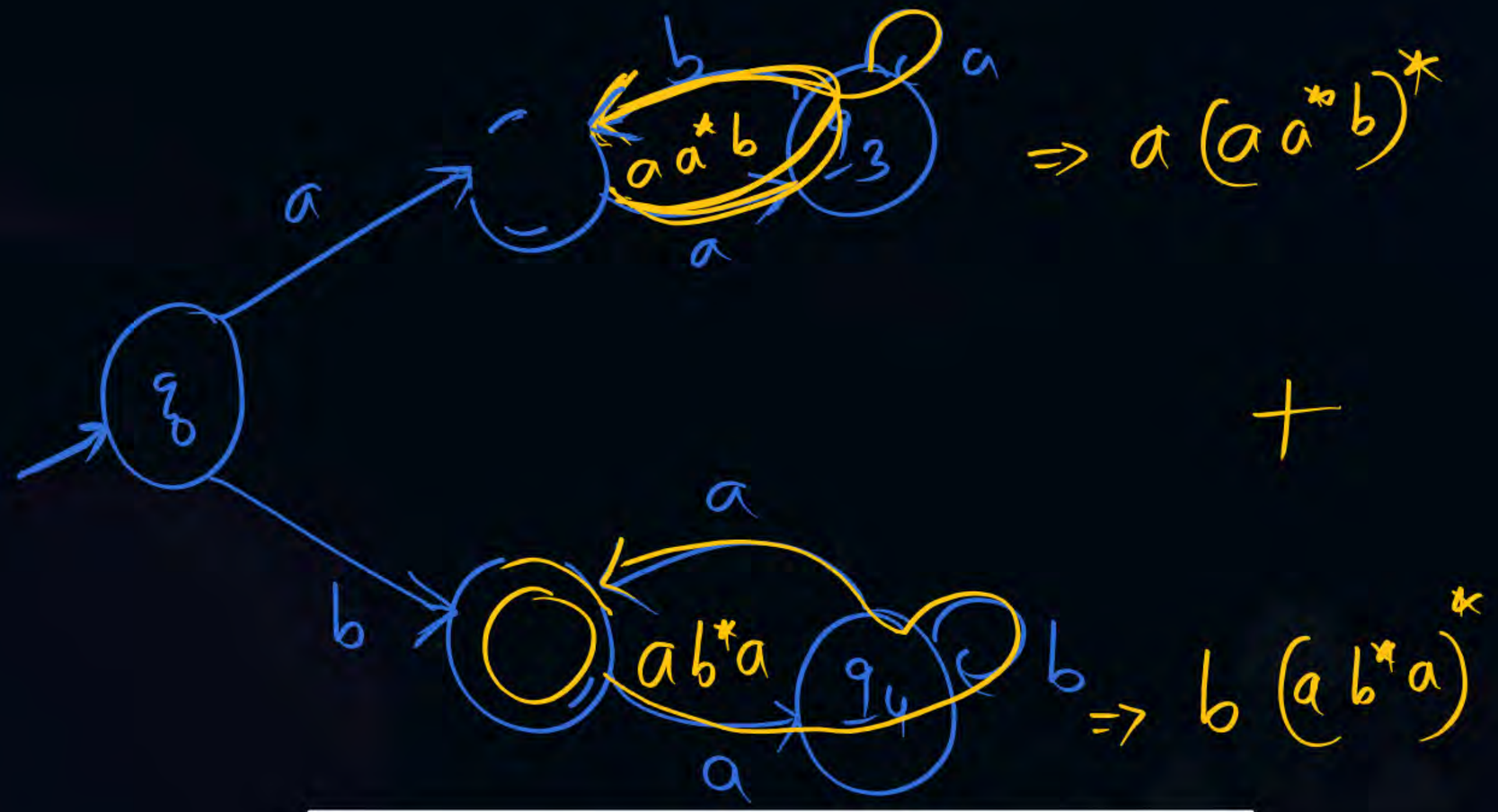
$a^*(\epsilon + b^+)$

$a^* b^*$

× $(a+b)^*$

× $a^* b^+$

b



$\Rightarrow a(aa^*b)^*$

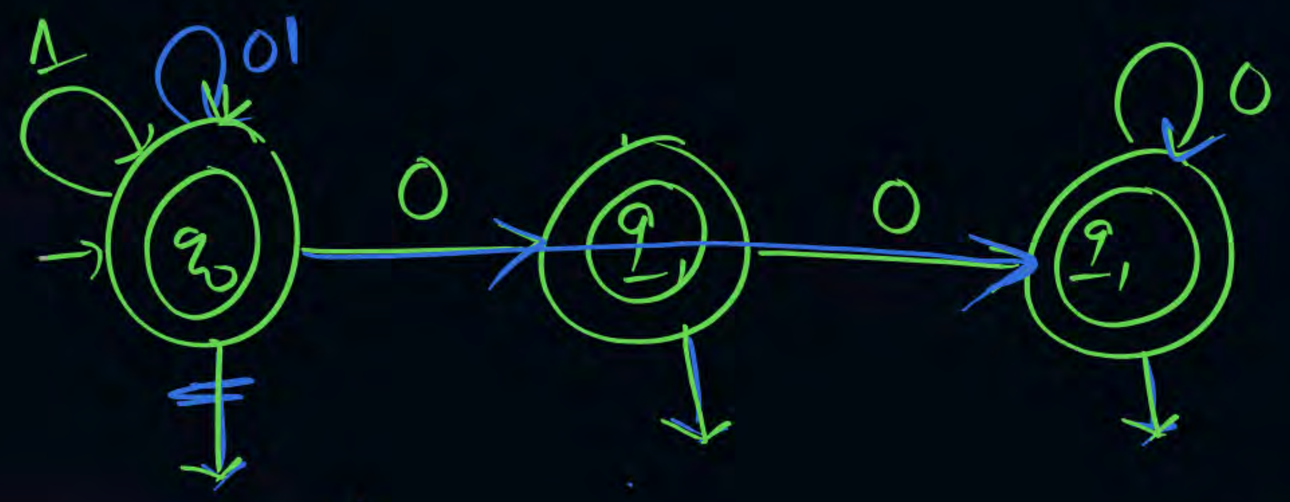
$+$

$\Rightarrow b(ab^*a)^*$

$$\boxed{a(aa^*b)^* + b(ab^*a)^*}$$

$$0^* + 0^* 1 (0+1)(0+1)^*$$

$$\boxed{0^* + 0^* 1 (0+1)^+}$$



$$(1+01)^* + (1+01)^*0 + (1+01)^*000^*$$

$$(1+01)^*\underbrace{[\epsilon+0+000^*]}$$

$$(1+01)^*0^*$$

THANK - YOU

CS & IT ENGINEERING

THEORY OF COMPUTATION

REGULAR EXPRESSION

Lecture No.- 07

By- Venkat sir

Recap of Previous Lecture

Topic

Regular expression Construction

?????

Properties of Regular expressions

Finite Automata to Regular expression

1. State elimination
2. Arden's method

Topics to be Covered

Topic Regular expression to F.A

Topic ??

Topic ??

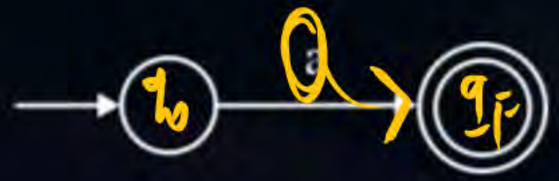
Topic ??

Regular grammar to Regular expression

Regular expression to Regular grammar

Regular Expression to Finite Automata

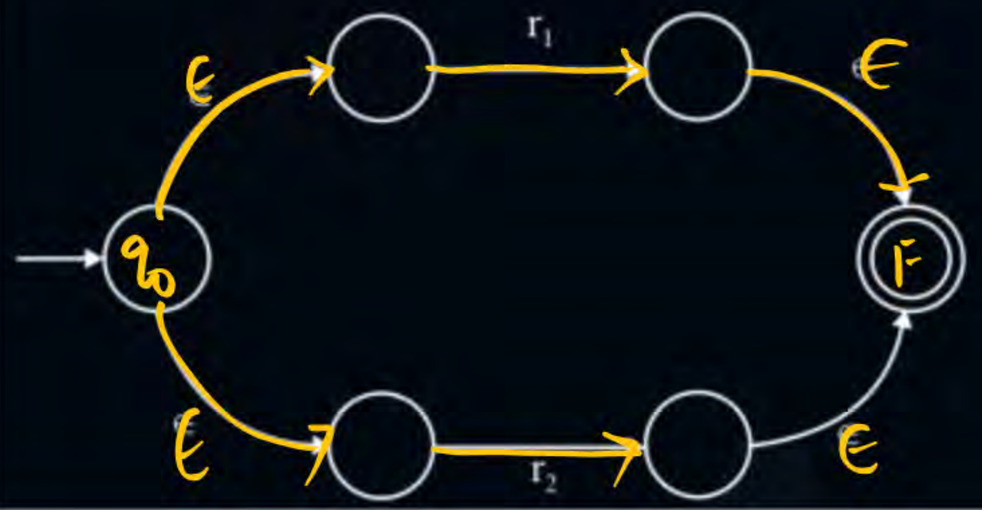
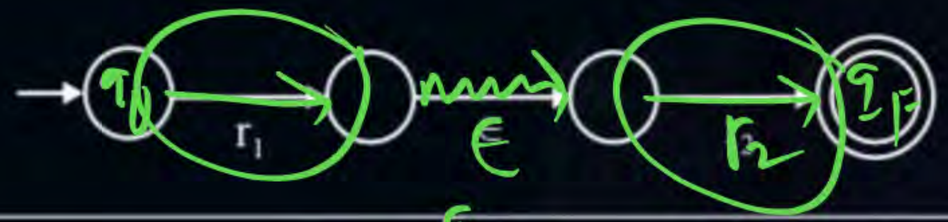
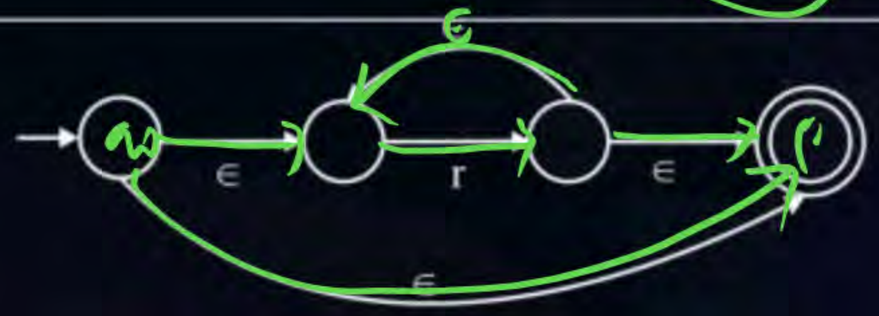
Thomson Construction Algorithm.

Req. Expr	∈NFA
1. ϕ	ϕ
2. ϵ	ϵ
3. a	 a

× DFA

× NFA

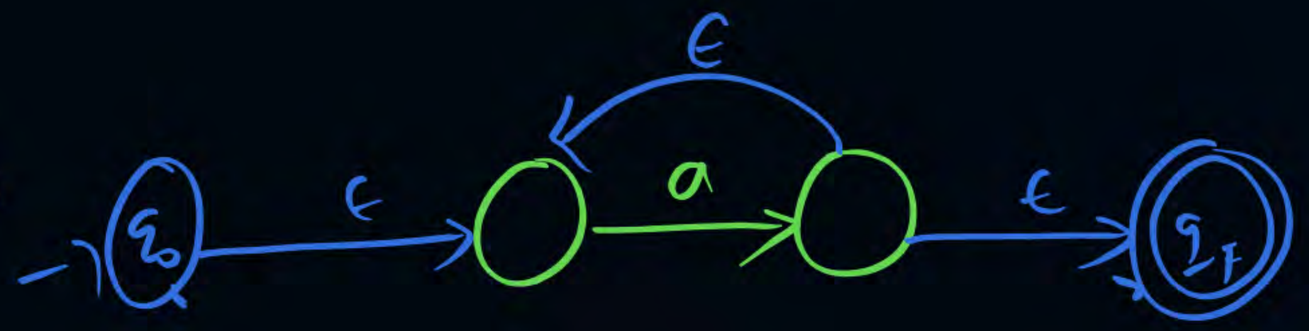
✓ ∈-NFA

Regular expression	ϵ - NFA
4. $r_1 + r_2$	
5. $r_1 \cdot r_2$	
6. r^*	

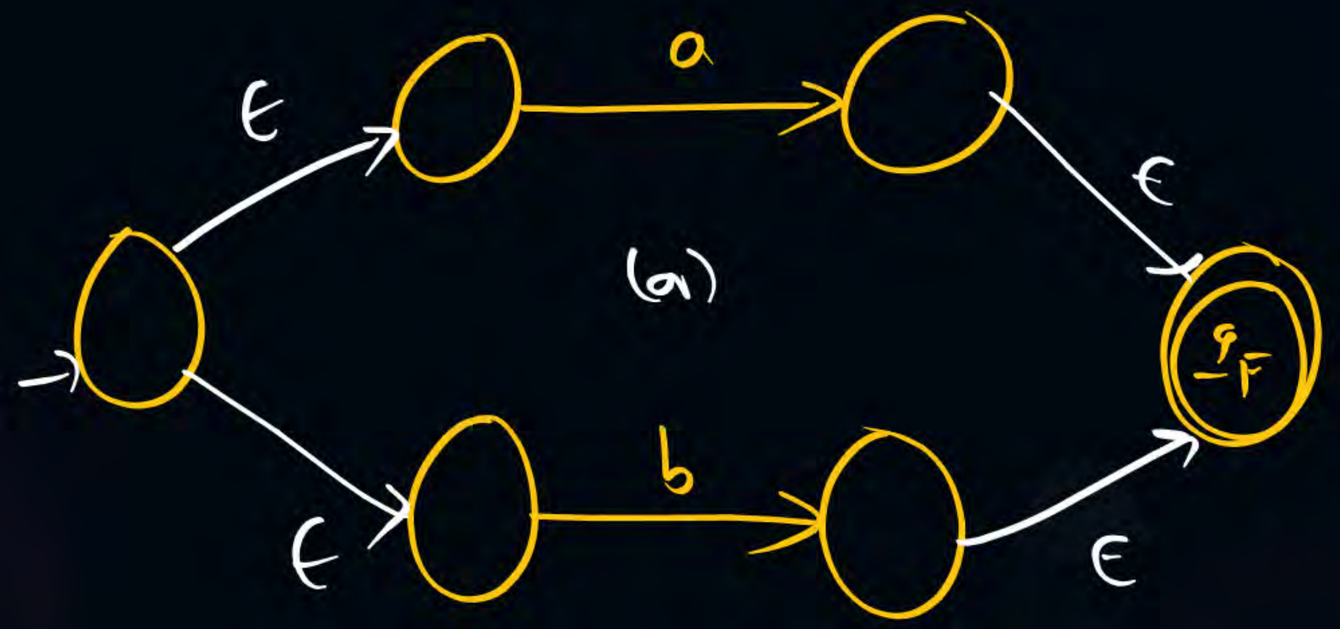
ϵ

a^*

a^+



$\underline{\underline{a+b}}$

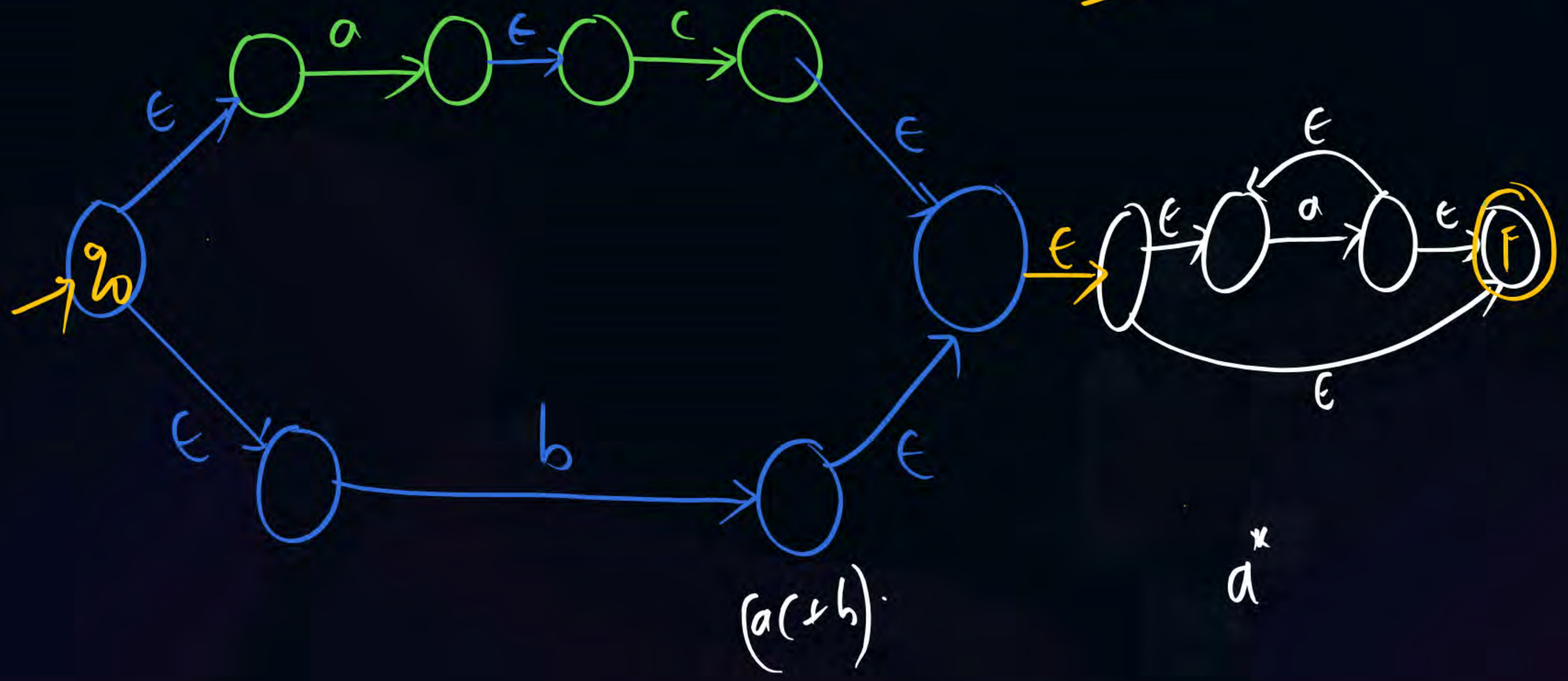


[NAT]

$(ac+b)\cdot a^*$

#Q. Construct ε-NFA for the following Regular Expression

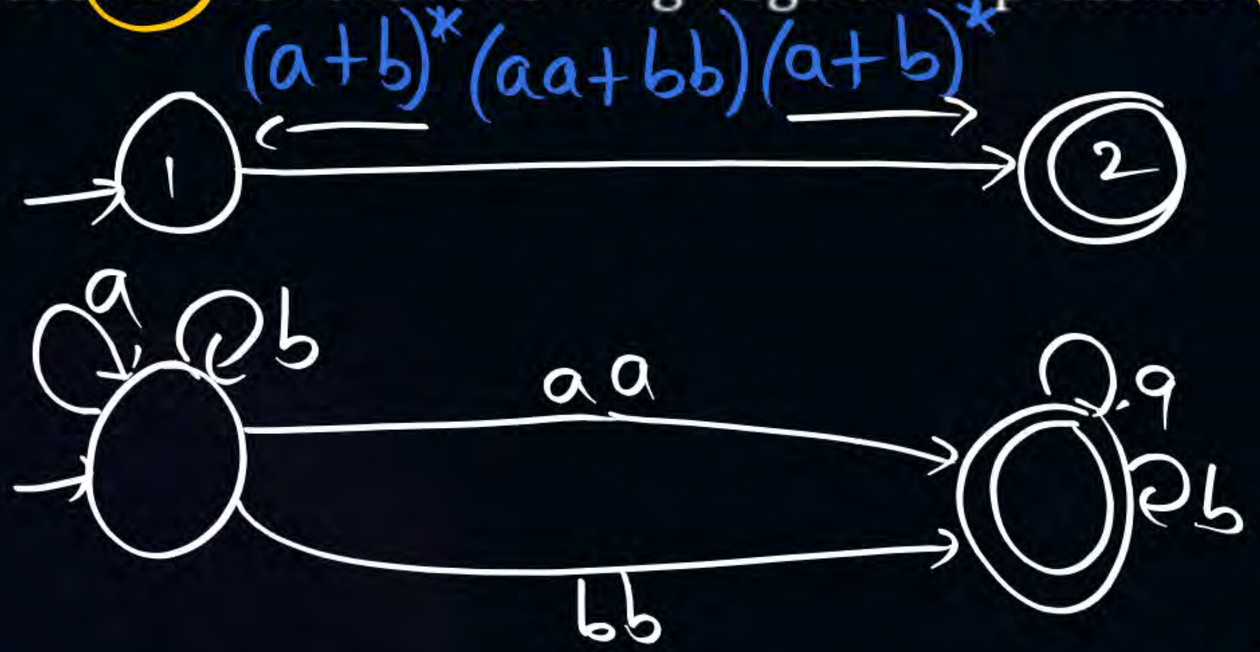
ε-NFA



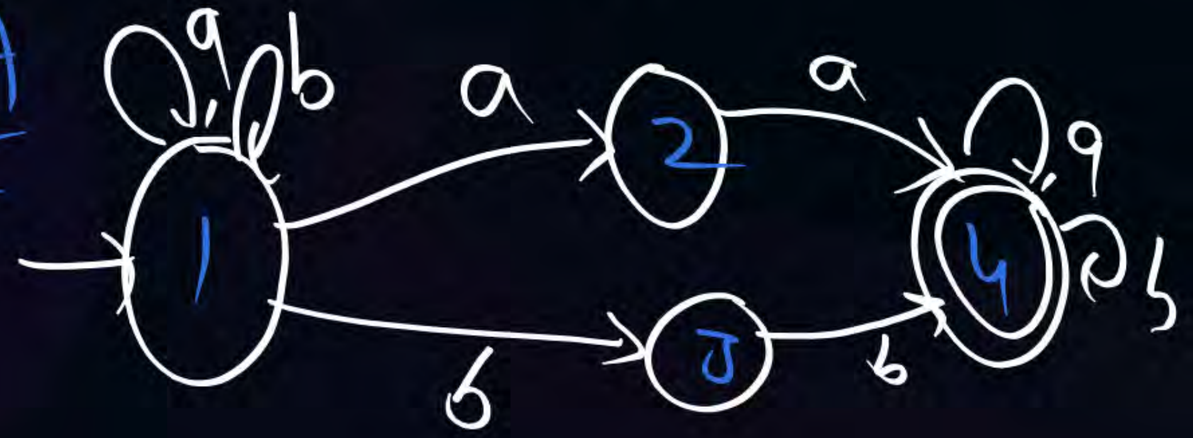
(Q) Construct ϵ-NFA for $(ab)^* a \cdot b^*$

[NAT]

#Q. Construct NFA (without ∈) for the following Regular Expression $(a+b)^*(aa+bb)(a+b)^*$

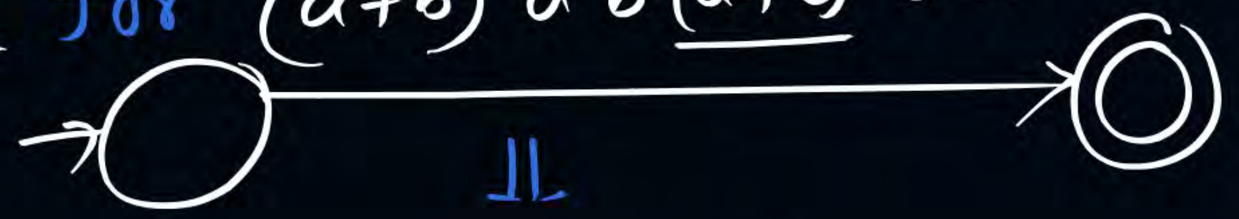


NFA

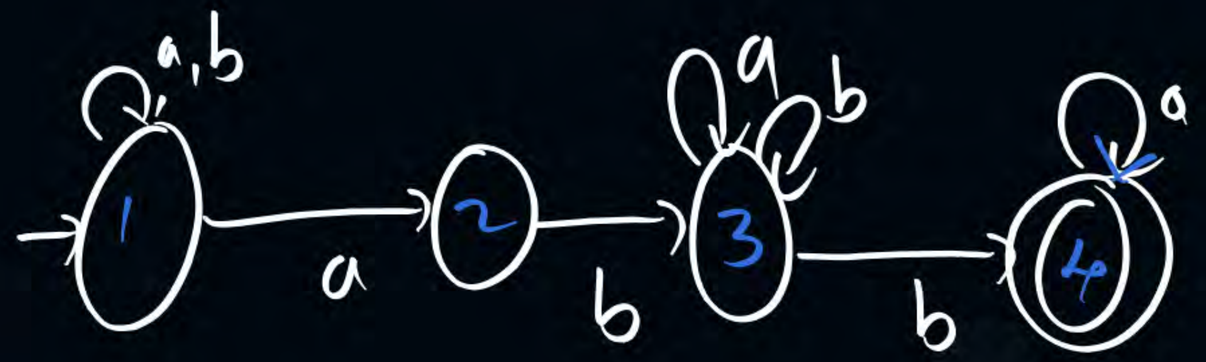


4 states

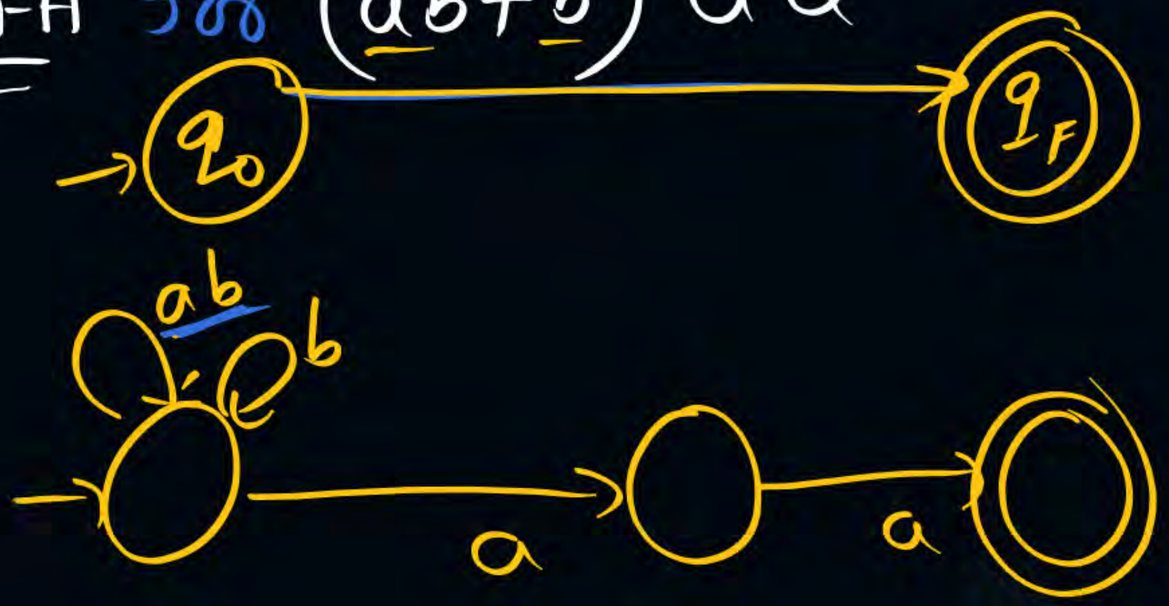
(Q) Construct NFA for $(a+b)^* a b (a+b)^* b a^*$



$\Downarrow$



(Q) Construct DFA for $(ab+b)^* aa$



DFA ⟸ NFA

States?

4 States

DFA

4 States

(Q) How many states present in minimal DFA for

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

3 states

[NAT]

min DFA

min

#Q. Construct DFA for the following Regular Expression

$a^* b^* \longrightarrow 3$

$a^* b^* c^* \longrightarrow 4$

$a^* b^* c^* d^* \longrightarrow 5$

$\vdots$

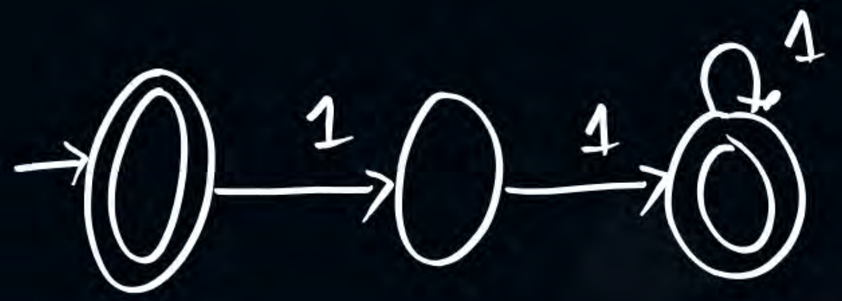
$a^* b^* c^* d^* \cdots z^* \longrightarrow 27$

$0^* 1^* 2^* \cdots 9^* \longrightarrow 11$

[NAT]

#Q. Construct min DFA for the following Regular Expression

$$r = (11 + 111)^* = \{\epsilon, 1^2, 1^3, 1^4, 1^5, 1^6, 1^7 \ldots\}$$



3 states DFA

[NAT]

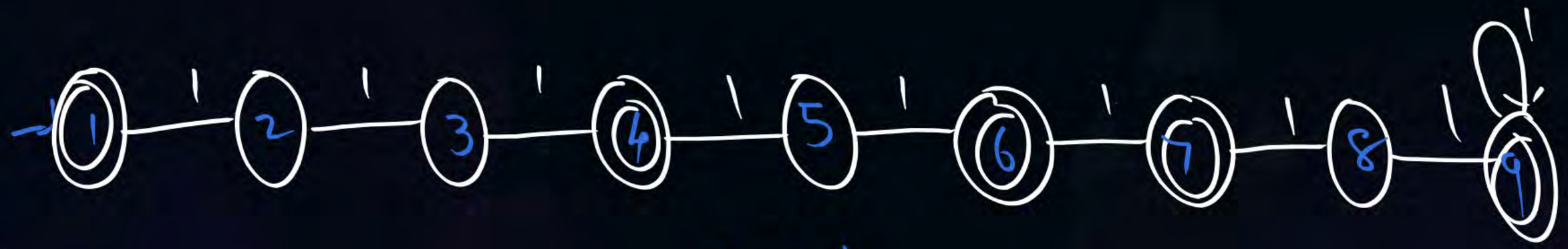
How many states in min

#Q. ~~Construct~~ DFA for the following Regular Expression

$(a+b)^* = (111+11111)^*$

$r = (111+11111)^*$

$\{\epsilon, 1^3, 1^5, 1^6, 1^8, 1^9, 1^{10}, 1^{11}, 1^{12}, 1^{13}, 1^{14}, 1^{15} \ldots\}$



9 States

[MCQ]

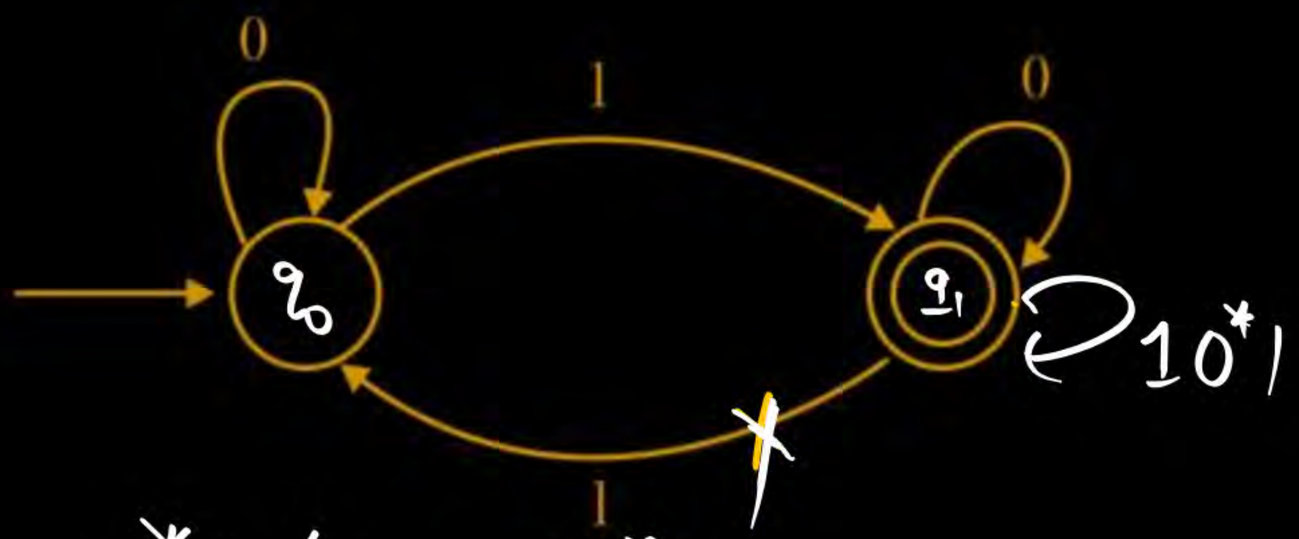
#Q. Which one of the following regular expressions is equivalent to the language accepted by the DFA given below? [GATE-CS-shift-II-24: 1M]

A. 0*1 (0 + 10*1)*

B. ~~0* (10*11) *0*~~ ×

C. 0 (1 + 0*10*1) *0*

D. 0*1(010*1) *0*



$0^*1(0+10^*1)^*$

[MCQ]

#Q. Let M be the 5-state NFA with ∈ - transitions shown in the diagram below. Which one of the following regular expressions represents the language accepted by M? [GATE-CS-shift-II-24: 2M]

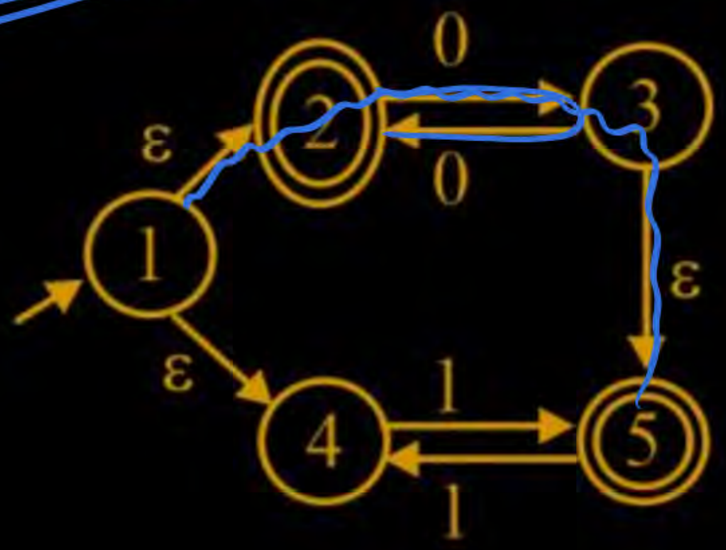
elimination

A. $0^* + (1 + 0\,(00)^*)\,(11)^*$

B. $(00)^* + 1\,(11)^*$

C. $(00)^* + (1 + (00)^*)\,(11)^*$

D. $0^+ + 1(11)^* + 0(11)^*$



$\{\epsilon, 0, 0^2, 0^3 \ldots\}$

0^*

[NAT]

#Q. Construct min DFA for the following Regular Expression DFA

① $(a+b)^3 \xrightarrow{\text{exactly}} 5$

② $(a+b+\epsilon)^5 \xrightarrow{\text{atmost}} 7$

③ $\left[(a+b)^4\right]^* \xrightarrow{\text{Div by 4}} 4$ states

④ $(a+b)^* b(a+b)(a+b) \xrightarrow{3^{rd}\ \text{R.HS}} 8$ states

THANK - YOU