KLP Mishra
Peter Linz

Theory Of Computation.

Page No.
Date:

1. Finite AutoMata

excluding input).

2. Regular Languages
3. Context Free Grammer (CFG)
4. Pushdown AutoMata
5. Turing Machine.

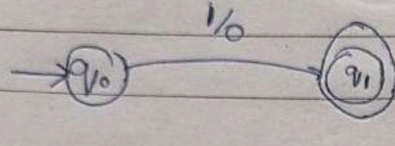
Unit 1 : Finite Automata

1. Finite Automata Model
2. Transition system
3. DFA Vs NFA
4. Mealy Machine And Moore Machine
   Conversion of Both Machine
   a) Mealy to Moore
   b) Moore to Mealy.
5. Conversion of DFA → NFA & NFA → DFA
6. Minimization of Finite Automata.

Page No.
Date:

Finite Automata

Analytical Definition

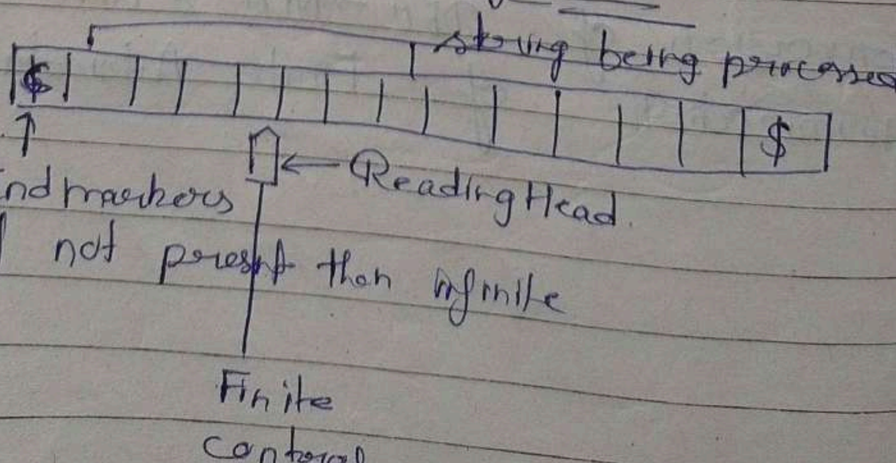


Finite Automata can be represented by five tuples
$(Q, \Sigma, \delta, q_0, F)$.

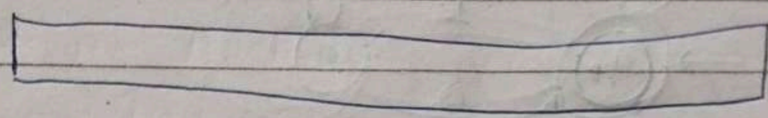
Q is finite non-empty set of state
Σ is finite non-empty set of input called the input alphabet.
$\delta \longrightarrow Q \times \Sigma$. All the states during transition
$[q_0 \times 1 \longrightarrow q_1]$ Transition

q_0 is initial state $q_0 \in Q$. [only one]
F is the subset Q $[F \subseteq Q]$ is the set of final states.

Block Diagram of F.A :-



Input Tape is section below divided into square.
Input Alphabets Σ i.e., {0, 1} {a, b, c}.

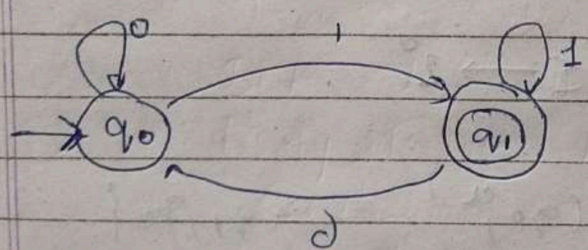


Restricted! Left to right
In case of F.A

Finite control changes state from other state.

Transition System!-

1) Transition diagram
2) Transition Table.



Table

	0	1
q_0	q_0	q_1
q_1	q_0	q_1

Acceptibility of String by F.A !-

	0	1
F.I. → q_0	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

1 1 0 0 1

~~1~~ ~~1~~ 0 1 0 1

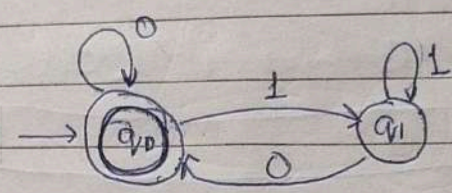
q_1 q_0 q_2 q_3 q_1

Finite Automat

(1) Deterministic F.A DFA — Non-Deterministic F.A NDFA/NFA

One path from curr. S to next S.

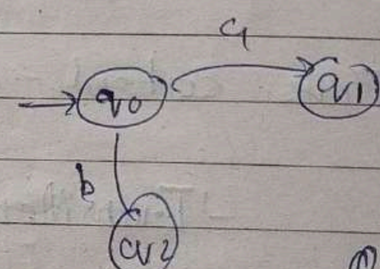
Design a Finite Automata with check wheather the given binary number is even.



11

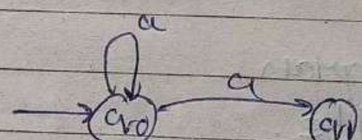
+ Deterministic Finite Automata +

$q_1 = \delta(q_0)a$



$Q \times \Sigma \Rightarrow Q$

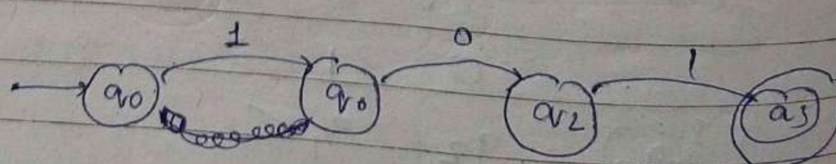
NDFA $\longrightarrow$



$Q \times \Sigma \longrightarrow 2^{Q}$

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

101 $\Sigma = \{0, 1\}$

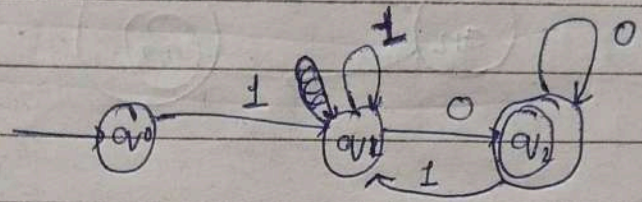


$Q = \{q_0, q_1, q_2, q_3\}$
$F = \{q_3\}$
Int $= q_0$
$\Sigma = \{0, 1\}$

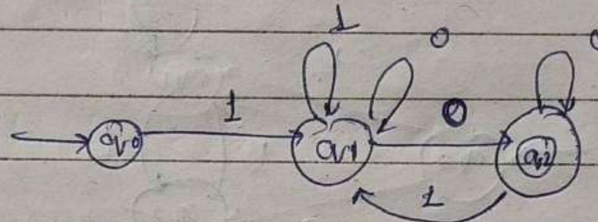
01,01

00

Q. Design F.A which accepts only those strings which starts with 1 and ends with 0.



DFA

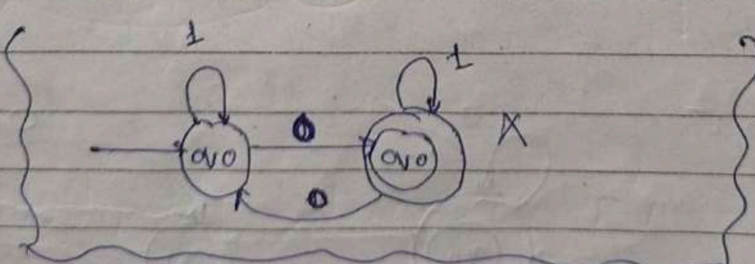


NDFA.

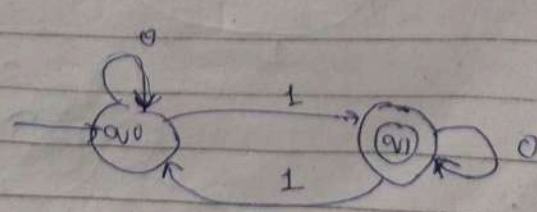
1 1 0 Φ 0

1 0 0 1 0 0

Q. Design a F.A which accepts odd no. of 1 and any no. 0's.



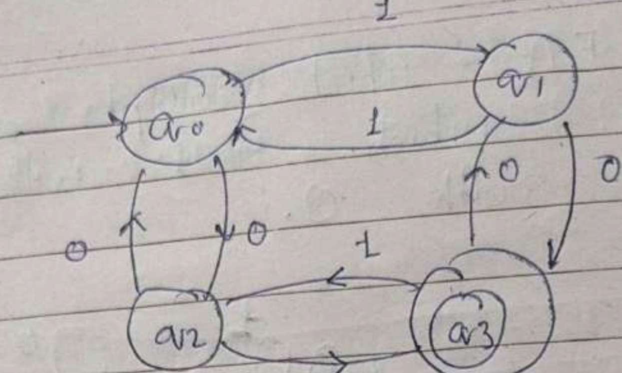
0111
010101

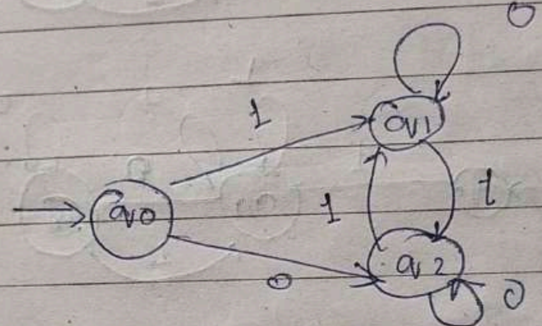


01,01,101

0000111

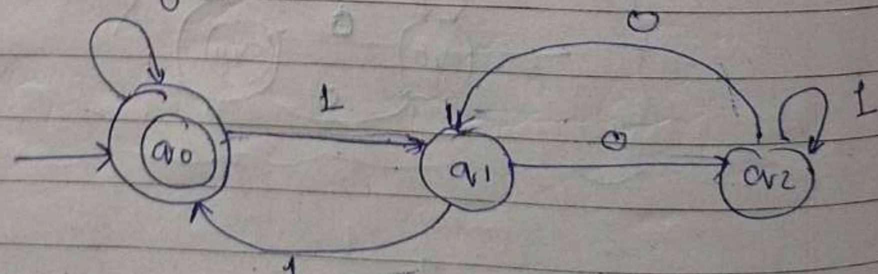
10

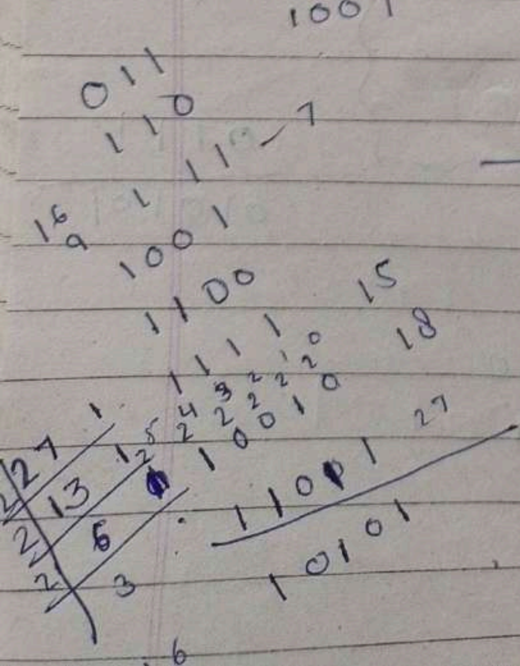


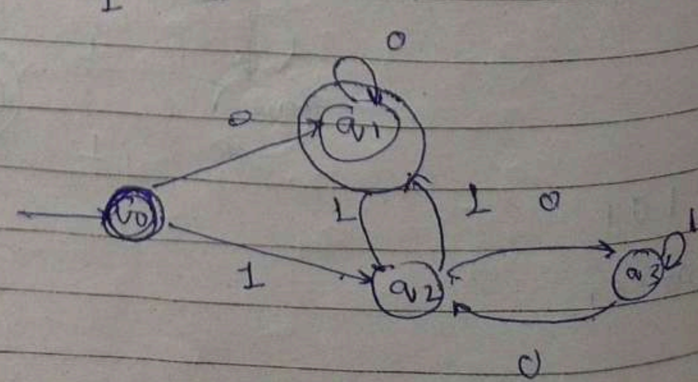


Design a finite automata which check divisible by three

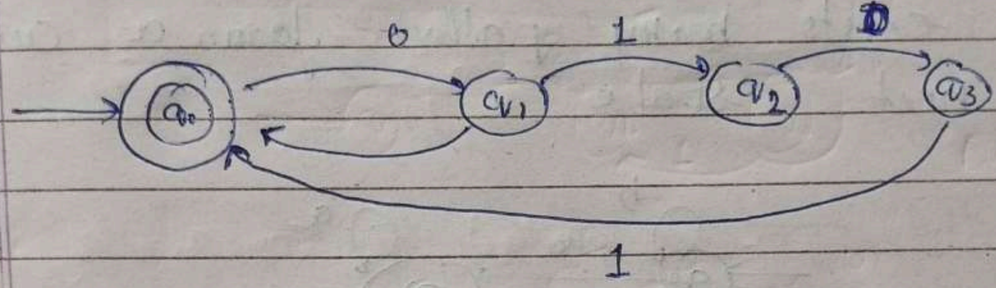
11

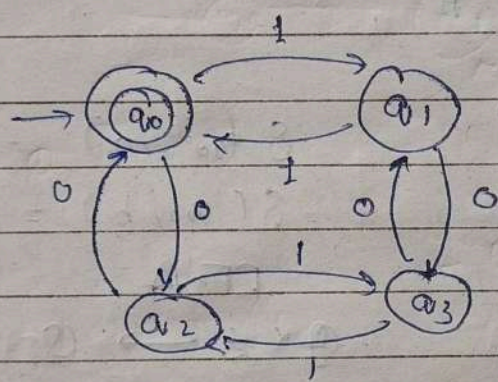




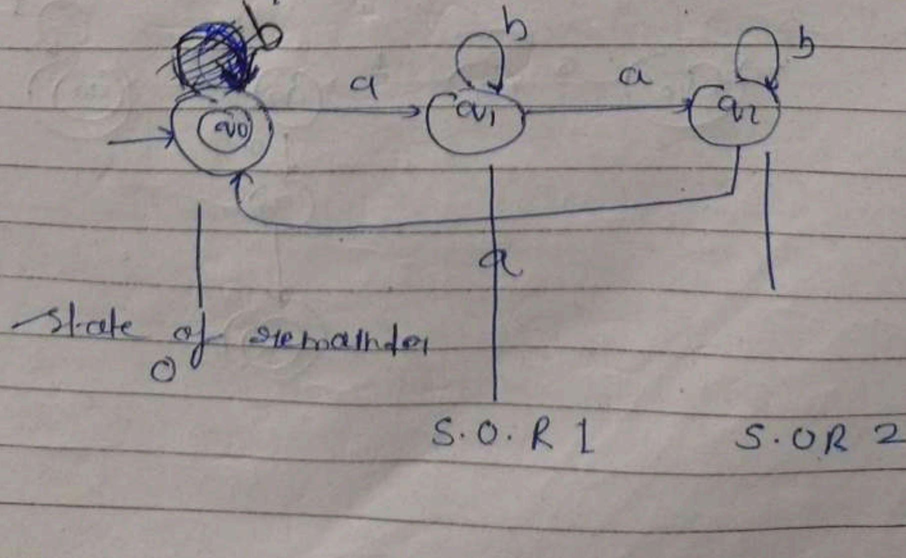


Design a F.A which accepts even no. of 0's and even no. 1's.





Accept language where all the string in L are subset total no. of 'a's in them divisible by 3.

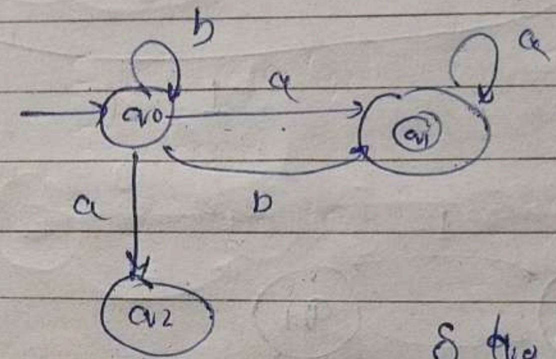


Date:

NDFA :-

Reverse of

There exists many paths ~~form~~ a current state to next state



$\delta(q_0, a) = \{q_1, q_2\}$

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

N·D·F·A

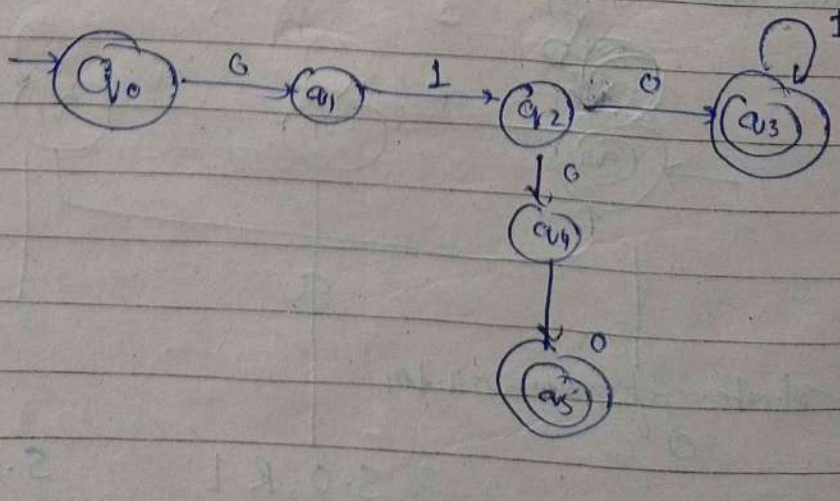
$Q \times \Sigma \rightarrow 2^Q$

DFA

$Q \times \Sigma \rightarrow Q$

NFA

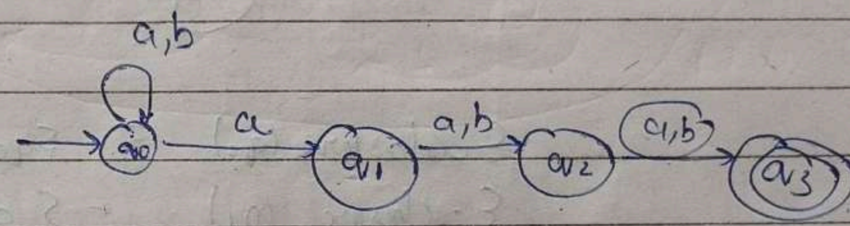
$(0101)^n \cup 0100 \; \{n \geq 0\} \quad \Sigma \{0, 1\}$



Q. Construct NFA

which accept Third symbol from right hand is always a.

$\boxed{a/b}$ a $\boxed{a/b}$ $\boxed{a/b}$



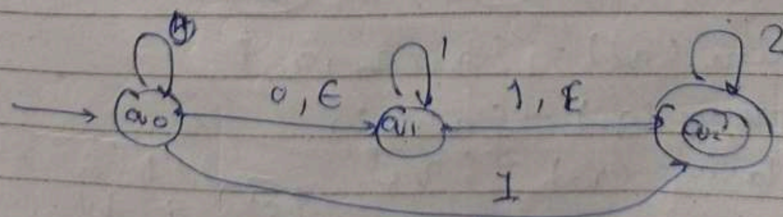
NFA

Q. Design NFA to accep string with a's and b's the string end with aa.

NFA with Epsilon Transition:-

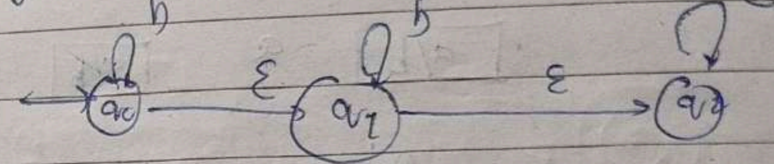
$\Sigma = \{0, 1\}$
$= \{a, b\}$

You can move from one state to another without consuming anything.



1 $q_0 \xrightarrow{\epsilon} q_1$ 1

ε - Closure :- The ε-closure (q_0) is a set of all states which are reachable from state q_0 on ϵ transition

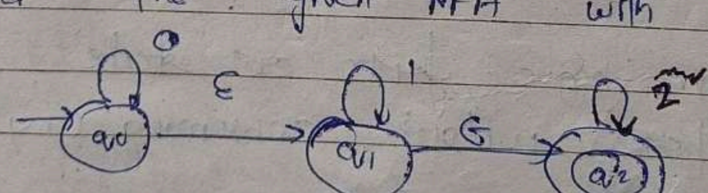


ε closure (q_0) = $\{q_0, q_1, q_2\}$
ε - closure (q_1) = $\{q_1, q_2\}$
ε - closure (q_2) = $\{q_2\}$

Eliminating ε - Transition :-

NFA with ε $\longrightarrow$ NFA without ε $\longrightarrow$ DFA.
We have to find reachable state with the help of ~~ε~~ closure.

Convert the given NFA with ε to NFA without ε



ε closure (q_0) = $\{q_0, q_1, q_2\}$
ε - closure (q_1) = $\{q_1, q_2\}$
ε - closure (q_2) = $\{q_2\}$

$\longrightarrow$ δ' del dash.

$\delta'(q_0, 0)$	$\delta'(q_1, 0)$
$\delta'(q_0, 1)$	$\delta'(q_1, 1)$
$\delta'(q_0, 2)$	$\delta'(q_1, 2)$

Page No.
Date:

$\delta'(q_0, 0) = \varepsilon\text{-closure}(\delta(\hat{\delta}(q_0, \epsilon), 0))$

$= \varepsilon\text{ closure}(\delta(\epsilon\text{ closure}(q_0), 0)$

$= \varepsilon\text{-closure}(\delta(q_0 q_1, q_2), 0)$

$= \varepsilon\text{-closure}(\delta(q_0, 0) \cup \delta(q_1, 0) \cup \delta(q_2, 0)$

$= \varepsilon\text{-closure}(q_0 \cup \phi \cup \phi)$

$= \varepsilon\text{-closure}(q_0)$

$\boxed{\delta'(q_0, 0) = \{q_0, q_1, q_2\}}$

$\delta'(q_0, 1) = \varepsilon\text{ closure}(\delta(\hat{\delta}(q_0, \epsilon), 1)$

$= \varepsilon\text{-closure}(\delta(\epsilon\text{ closure}(q_0), 1)$

$= \varepsilon\text{-closure}(\delta(q_0, 1) \cup \delta(q_1, 1) \cup \delta(q_2, 1))$

$= \epsilon\text{closure}(\phi \cup q_1 \cup \phi)$

$= \varepsilon\text{-closure}(q_1)$

$\delta'(q_0, 1) = \{q_1, q_2\}.$

$\delta'(q_0, 2) = \varepsilon\text{-closure}(\delta(\delta'(q_0, \epsilon), 2)$

$= \varepsilon\text{-closure}(q_2)$

$\delta'(q_0, 2) = \{q_2\}$

(ε-closure)

$\delta'(q_1, 0) = \epsilon\text{ closure}(\delta(\delta'(q_1, \epsilon), 0)$

$= \varepsilon\text{ closure}(\delta(\epsilon\text{ closure}(q_1), 0)$

$= \varepsilon\text{ closure}(\delta(q_1, 0) \cup \delta(q_2, 0))$

$= \varepsilon\text{-closure}(\phi \cup \phi)$

(ε-closure)

$\delta'(q_1, 0) = \varepsilon\text{-closure}(\phi)$

$\boxed{\delta'(q_1, 0) = \phi}$

$\delta'(q_1, 1) = \epsilon\text{ closure}(\delta(\delta'(q_1, \epsilon), 1)$

$= \varepsilon\text{ closure}(\delta(q_1, q_2), 1)$

$= \varepsilon\text{ closure}(\delta(q_1, 1) \cup \delta(q_2, 1)$

$= \varepsilon\text{-closure}(q_1)$

$\delta'(q_1, 1) = \{q_1, q_2\}$

$$\delta'(q_1, 2) = \varepsilon\text{-closure}\,(\delta(\delta'(q_1, \varepsilon), 2))$$
$$= \varepsilon\text{-closure}\,(\delta(q_1, q_2), 2)$$
$$= \varepsilon\text{-closure}\,(\delta(q_1, 2) \cup \delta(q_2, 2)$$
$$\phi \cup q_2$$
$$\varepsilon\text{-closure}\,(q_2)$$
$$\delta'(q_1, 2) = \{q_2\}$$

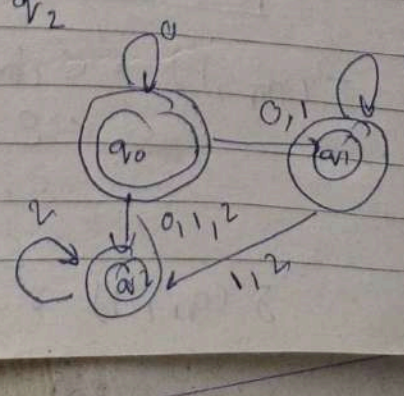
$$\delta'(q_2, 0) = \varepsilon\text{-closure}\,(\delta(\delta'(q_2, \varepsilon), 0)$$
$$= \varepsilon\text{-closure}\,(\delta\{q_2\}, 0)$$
$$= \varepsilon\text{-closure}\,(\delta(q_2, 0))$$
$$= \varepsilon\text{-closure}\,\{\phi\}$$
$$= \phi$$

$$\delta'(q_2, 1) = \varepsilon\text{-closure}\,(\delta(\delta'(q_2, \varepsilon), 1)$$
$$= \varepsilon\text{-closure}\,(\delta(q_2, 1))$$
$$= \varepsilon\text{-closure}\,(\phi)$$
$$\delta'(q_2, 1) = \phi$$

$$\delta'(q_2, 2) = \varepsilon\text{-closure}\,(\delta(\delta'(q_2, \varepsilon), 2)$$
$$= \varepsilon\text{-closure}\,(\delta(q_2, 2)$$
$$\delta'(q_2, 2) = \{q_2\}$$

	0	1	2
q_0	q_0, q_1, q_2	q_1, q_2	q_2
q_1	ϕ	q_1, q_2	q_2
q_2	ϕ	ϕ	q_2

NFA without ε



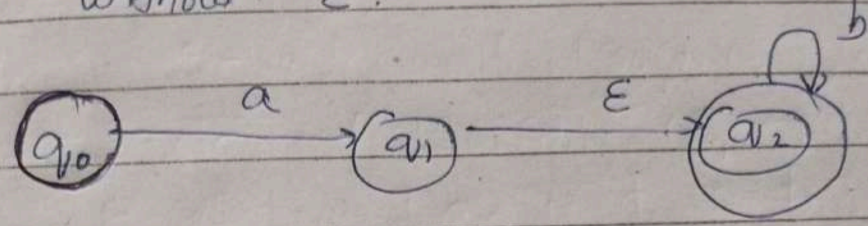
final state

ε-closure $(q_0) = \{q_0, q_1, q_2\}$

$(q_1) = \{q_1, q_2\}$

$(q_2) = \{q_2\}$

Convert the following NFA with 'ε' to NFA without ε.



ε-closure $(q_0) = \{q_0\}$
ε-closure $(q_1) = \{q_1, q_2\}$
ε-closure $(q_2) = \{q_2\}$

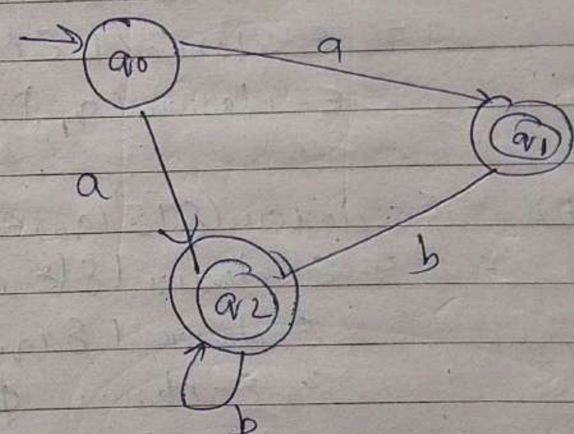
$\delta'(q_0, a) = \varepsilon\text{-closure}(\delta(\delta'(q_0, \varepsilon), a)$
$= \varepsilon\text{-closure}(\delta(\varepsilon\text{-closure}(q_0), a)$
$= \varepsilon\text{-closure}(\delta(q_0, a))$
$= \varepsilon\text{-closure}(q_1) = \{q_1, q_2\}$

$\delta'(q_0, b) = \varepsilon\text{-closure}(\delta(\delta'(q_0, \varepsilon), b)$
$= \varepsilon\text{-closure}(\delta(\varepsilon\text{-closure}(q_0), b)$
$= \varepsilon\text{-closure}(\delta(q_0, b))$
$= \varepsilon\text{-closure}\ \phi$
$= \phi$

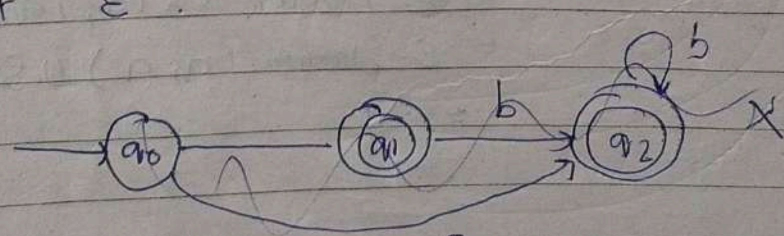
$\delta'(q_1, a) = \varepsilon\text{-closure}(\delta(\varepsilon\text{-closure}\ q_1), a)$
$= \varepsilon\text{-closure}(\delta(q_1, q_2), a)$
$= \varepsilon\text{-closure}(q_1, a) \cup \delta(q_2, a)$
$= \phi$

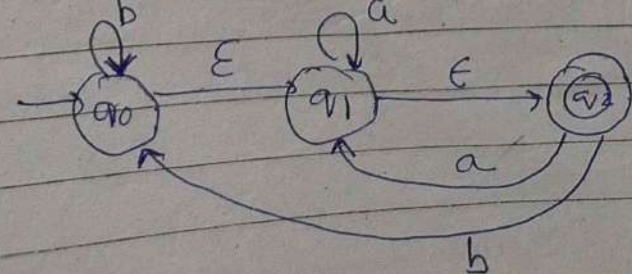
$= \{q_0, q_1, q_2\}$
$= \{q_1, q_2\}$
$= \{q_2\}$

State	a	b
q_0	q_1, q_2	ϕ
q_1	ϕ	q_2
q_2	ϕ	q_2



Q Construct give NFA with 'ε' to NFA without ε.





Conver

ε-closure $(q_0) = \{q_0, q_1, q_2\}$

ε-closure $(q_1) = \{q_1, q_2\}$

ε- $(q_2) = \{q_2\}$.

$\delta'(q_0, a) = \varepsilon\text{-closure}\,(\delta(\delta'(q_0, \varepsilon), a)$

$= \varepsilon\text{-closure}\,(\delta(\varepsilon\text{-closure}(q_0), a)$

$= \varepsilon\text{-closure}(\delta\{q_0, q_1, q_2\}, a)$

$= \varepsilon\text{-closure}\,(\delta(q_0, a) \cup \delta(q_1, a) \cup \delta(q_2, a)$

$= \varepsilon\text{-closure}\,(\phi \cup q_1 \cup q_1)$

$= \varepsilon\text{-closure}\,(q_1)$

$= \{q_1, q_2\}$

$\delta'(q_0, b) = \varepsilon\text{-closure}(\delta(\delta'(q_0, \varepsilon), b)$

$= \varepsilon\text{-closure}(\delta(q_0, b) \cup \delta(q_1, b) \cup \delta(q_2, b))$

$= \varepsilon\text{-closure}\,(q_0 \cup \phi \cup q_0)$

$= \varepsilon\text{-closure}\,(q_0)$

$= \{q_1, q_2, q_0\}$.

$\delta'(q_1, a) = \varepsilon\text{-closure}\,(\delta(\delta'(q_1, \varepsilon), a)$

$= \varepsilon\text{-closure}\,(\delta(q_1, a) \cup \delta(q_2, a))$

$= \varepsilon\text{-closure}(q_1 \cup q_1)$

$= \varepsilon\text{-closure}\,(q_1)$

$= \{q_1, q_2\}$.

$\delta'(q_1, b) = \varepsilon\text{-closure}\,(\delta(q_1, b) \cup \delta(q_2, b))$

$= \varepsilon\text{-closure}(\phi \cup q_0)$

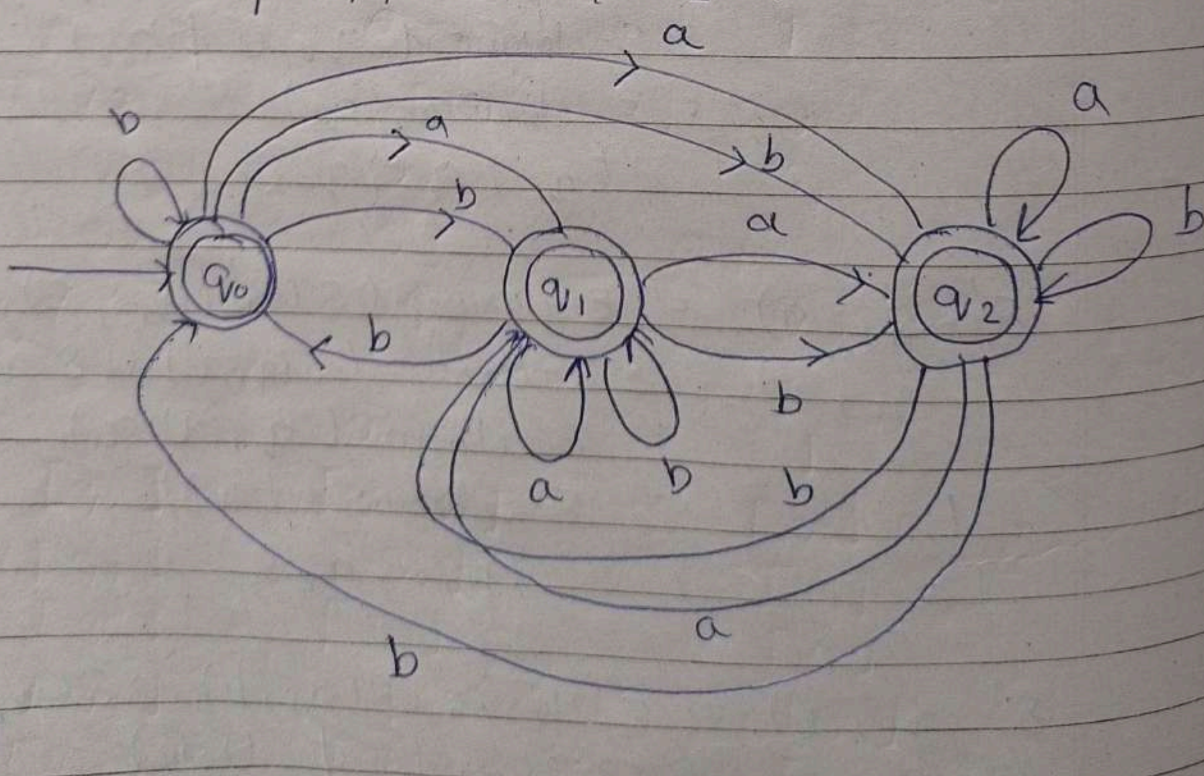
$= \{q_1, q_2, q_0\}$

$\delta'(q_2,$

$$\delta'(q_2, a) = \varepsilon\text{-closure}(\delta(\delta'(q_2, \varepsilon), a))$$
$$= \varepsilon\text{-closure}(\delta(q_2, a))$$
$$= \varepsilon\text{-closure}(q_1)$$
$$= \{q_1, q_2\}$$

$$\delta'(q_2, b) = \varepsilon\text{-closure}(\delta(\delta'(q_2, \varepsilon), b))$$
$$= \varepsilon\text{-closure}(\delta(q_2, b))$$
$$= \varepsilon\text{-closure}(q_0)$$
$$= \{q_0, q_1, q_2\}$$

State	a	b
q_0	q_1, q_2	q_0, q_1, q_2
q_1	q_1, q_2	q_0, q_1, q_2
q_2	q_1, q_2	q_1, q_2, q_0



Conversion From NFA to DFA

Let $M = (\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \{q_1\})$

be NFA where $\delta(q_0, 0) = \{q_0, q_1\}$.

$\delta(q_0, 1) = \{q_1\}$, $\delta(q_1, 0) = \phi$, $\delta(q_1, 1) = \{q_0, q_1\}$

Construct the equivalent DFA.

Two states - q_0, q_1
Input = 0, 1
Initial state = q_0
Final state = q_1
Transition state = δ.

I.

States.	0	1
q_0	q_0, q_1	q_1
q_1	ϕ	q_0, q_1

II.

	0	1
$\rightarrow [q_0]$	$[q_0, q_1]$	$[q_1]$
$[q_1]$	$[\phi]$	$[q_0, q_1]$
$[q_0, q_1]$	$[q_0, q_1]$	$[q_0, q_1]$

Equivalent DFA.

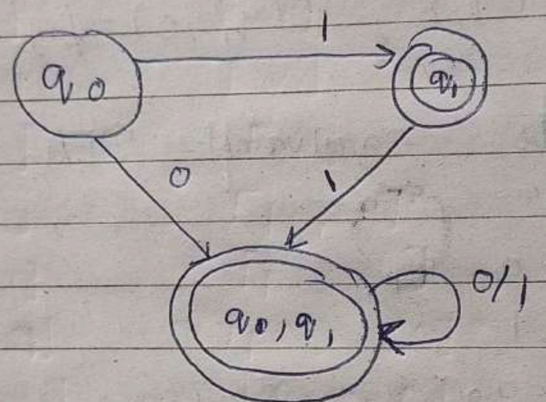
$\delta((q_0, q_1), 0) = \delta(q_0, 0) \cup \delta(q_1, 0)$
$= \{q_0, q_1\} \cup \phi$

Date:

Find will be that where part Final lies.

$\rightarrow q_1$, $[q_1, q_2]$

Let $q_0 = a$, $q_1 = b$, $q_0, q_1 = c$.

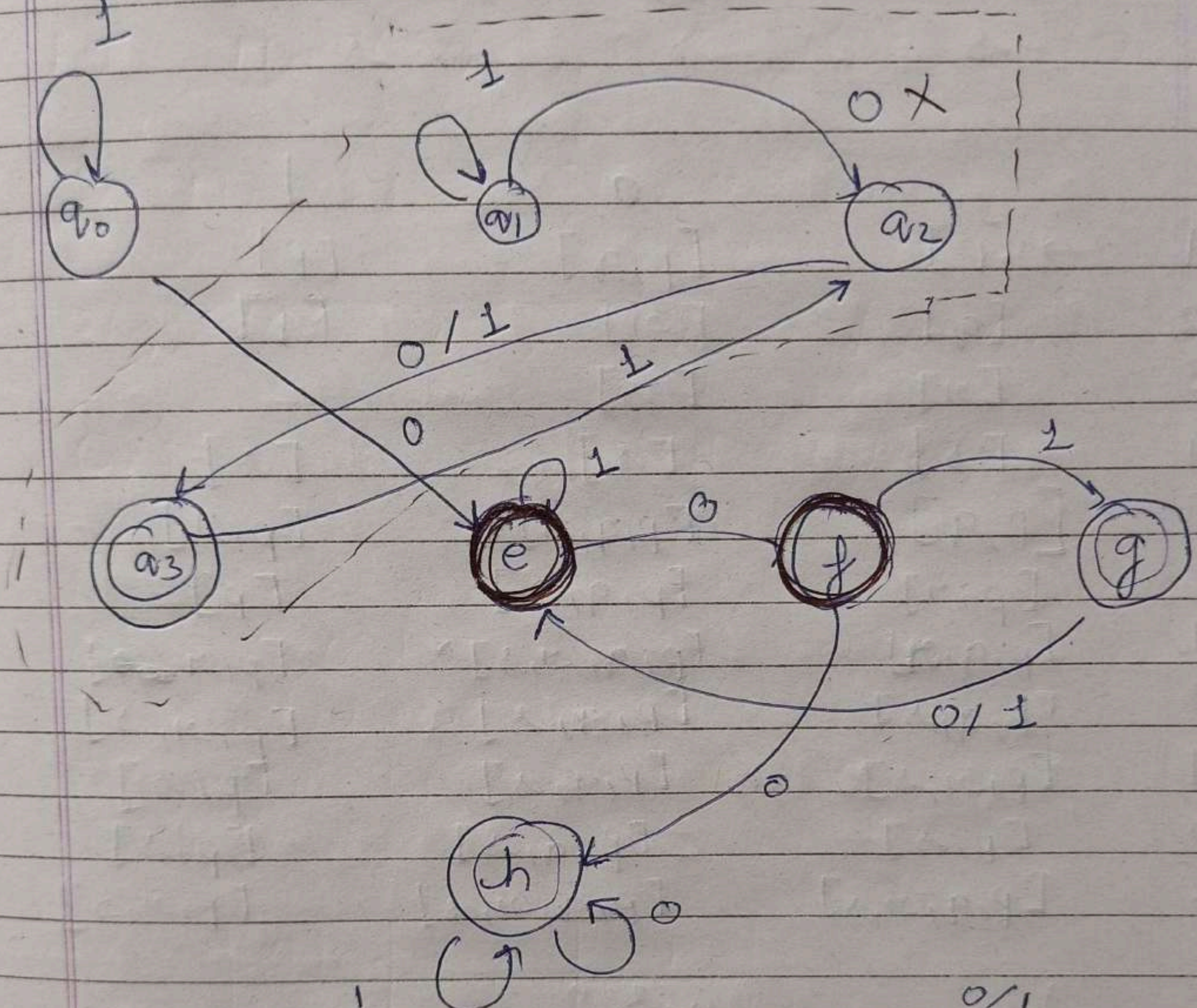


2.

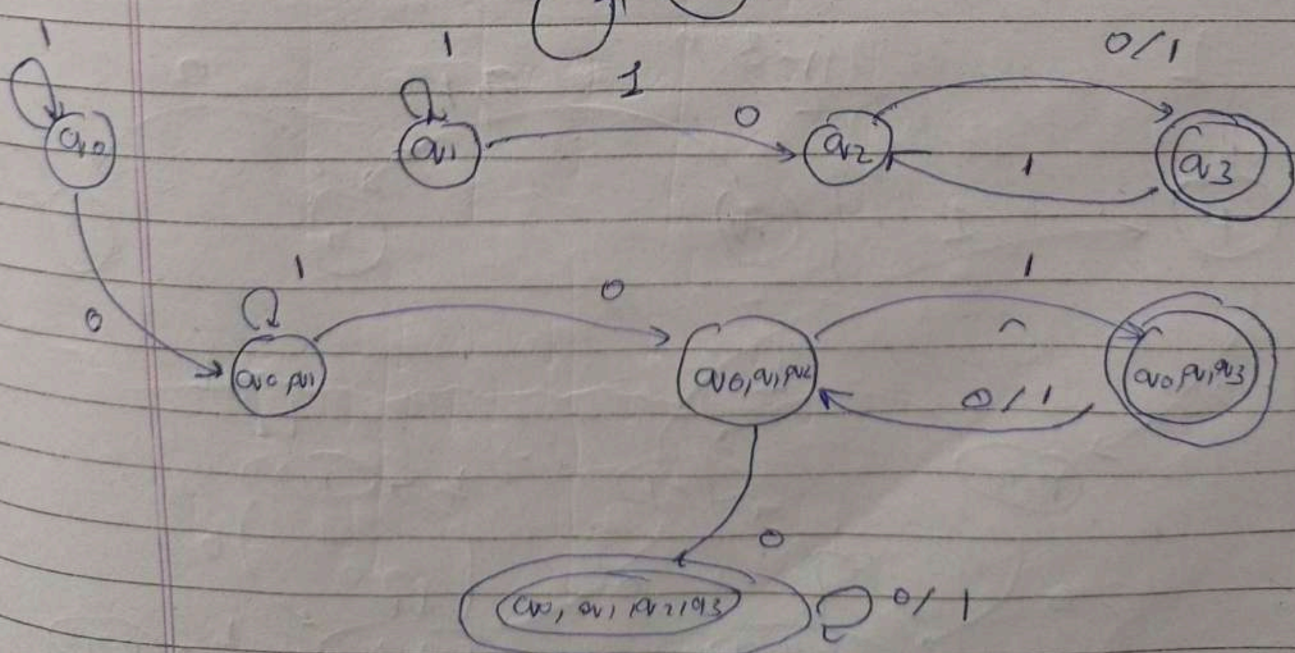
	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	q_2	q_1
q_2	q_3	q_3
q_3	ϕ	q_2

		0	1
a	$[q_0]$	$[q_0, q_1]$	$[q_0]$
b	$[q_1]$	$[q_2]$	$[q_1]$
c	$[q_2]$	$[q_3]$	$[q_3]$
d	$[q_3]$	ϕ	$[q_2]$
e	$[q_0, q_1]$	$[q_0, q_1, q_2]$	$[q_0, q_1]$
f	$[q_0, q_1, q_2]$	$[q_0, q_1, q_2, q_3]$	$[q_0, q_1, q_3]$
g	$[q_0, q_1, q_3]$	$[q_0, q_1, q_2]$	$[q_0, q_1, q_2]$
h	$[q_0, q_1, q_2, q_3]$	$[q_0, q_1, q_2, q_3]$	$[q_0, q_1, q_2, q_3]$

Date:



q0
q1
q2
q3
e
f
g
h
1
0 ✗
0/1
1
0
1
0
1
0/1
0
0



q0
q1
q2
q3
1
1
0
0/1
1
q0,q1
q0,q1,q2
q0,q1,q3
q0,q1,q2,q3
0
1
0
1
0/1
0
0/1

Page No.
Date:

Q.3.

		0	1
	→ p	{p, q}	p
	q	r	r
	r	s	—
	s	s	s

		0	1
1	→ [p]	[p, q]	[p]
2	[q]	[r]	[r]
3	[r]	[s]	—
4	[s]	[s]	[s]
5	[p, q]	[p, q, r] ×	[p, r]
6	[p, r]	[p, q, s]	[p]
7	[p, q, r]	[p, q, r, s] ×	[p, r]
8	[p, q, s]	[p, q, s]	[p, r, s]
9	[p, r, s]	[p, q, s]	[p, s]
10	[p, s]	[p, q, s]	[p, s]
11.	[p, q, r, s]	[p, q, r, s]	[p, r, s]

11 - 3 = 8