Complete formal def^n of NFA :

NFA M is a 5 tuple $= (Q, \Sigma, q_0, \delta, F)$

where - Q : finite set of states

Σ : i/p alphabet

q_0 : $q_0 \in Q$ is start state.

F : finite set of final states where $F \subseteq Q$. , $\delta : Q \times \Sigma \to 2^Q$ (transition function)

Formal def^n of extended transition function :

Definition by induction on the length of input strings-

Basis : $\hat{\delta}(q, \epsilon) = \{q\}$ - [without reading any i/p symbol, ~~we~~ ~~the~~ automaton M will be there in the same state].

Induction : Suppose w is of the form $w = xa$, where a is the last symbol and x is rest part of w.

$$\hat{\delta}(q, x) = \{p_1, p_2, \ldots, p_k\}. \text{ let}$$

$$\bigcup_{i=1}^{k} \delta(p_i, a) = \{r_1, r_2, \ldots, r_m\}, \text{ so}$$

now can write that $\hat{\delta}(q, w) = \{r_1, r_2, \ldots, r_m\}$

The language of NFA,

$$L(M) = \{w \mid \hat{\delta}(q_0, w) \cap F \neq \phi\}$$

Theorem : If DFA $D = (Q_D, \Sigma, \delta_D, q_0, F_D)$ is constructed from NFA $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$, then $L(D) = L(N)$.

Construction : $Q_D = 2^{Q_N}$, Σ same, $[q_0]$ - start state, $F_D = \{A \subseteq Q_N \mid F_N \cap A \neq \phi\}$

- Proof : We will prove by induction,

$$\hat{\delta}_D([q_0], w) = \hat{\delta}_N(\{q_0\}, w)$$

Note : Each of $\hat{\delta}$ function of DFA D, returns a state $q_i \in Q_D$. and each of the transition function of NFA N, returns a set of states $S \subseteq Q_N$.

$= S$

Basis : If $w = \varepsilon$, then

$$\hat{\delta}_D([q_0], \varepsilon) = [q_0],$$

$$\hat{\delta}_N(\{q_0\}, \varepsilon) = \{q_0\}$$

Induction : let $|w| = n+1$. We write w in the following form, $w = xa$ where a is the last symbol of w. By induction hypothesis -

$$\hat{\delta}_D([q_0], x) = [P_1, P_2, \ldots P_k].$$

$\hat{\delta}(\{q_0\}, x) = \{P_1, P_2, \ldots P$

the inductive part of the definition of NFA,

$$\boxed{\hat{\delta}_N(q_0, w) = \bigcup_{i=1}^{K} \delta_N(P_i, a)} \quad \ldots (1)$$

According to subset construction,

$$\delta_D([P_1, P_2, \dots P_K], a) = \bigcup_{i=1}^{K} \delta_N(P_i, a) \quad \dots (2)$$

Now using equ^n (1) and equ^n (2), we can write,

$$\hat{\delta}_D(q_0, w) = \delta_D(\hat{\delta}_D([q_0], x), a)$$
$$= \delta_D([P_1, P_2, \dots, P_K], a)$$
$$= \bigcup_{i=1}^{K} \delta_N(P_i, a)$$

So, we can write, $\hat{\delta}_D(q_0, w) = \hat{\delta}_N(q_0, w)$

and ~~if $\hat{\delta}_D(q_0, w) \in F$ and $\hat{\delta}_N(q_0, w) \cap F \neq \phi$,~~ then we can say that $L(D) = L(N)$ - (proved)

---

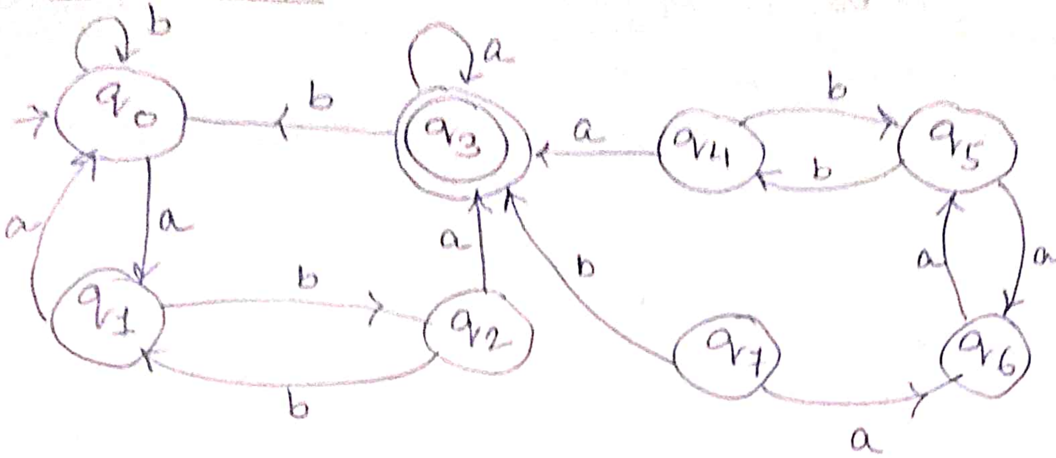
if $\hat{\delta}_D(q_0, w) \in F$ and

$\hat{\delta}_N(q_0, w) \cap F \neq 0$ or,

$\hat{\delta}_D(q_0, w) \notin F$ and

$\hat{\delta}_N(q_0, w) \cap F = \phi$ —

~~then we may writ~~



Unreachable states : q_4, q_5, q_6, q_7 – we need remove all of them.

	a	b
→ q_0	q_1	q_0
q_1	q_0	q_2
q_2	q_3	q_1
q_3	q_3	q_0

0-equivalence : $[q_0, q_1, q_2]$ $[q_3$

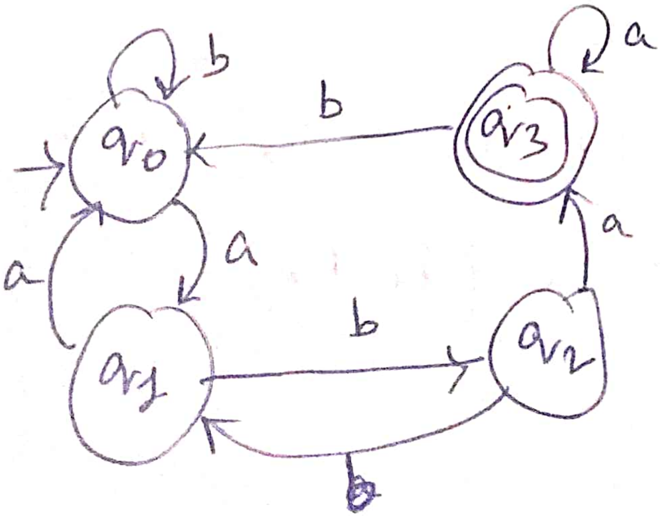
1-equivalence :

$[q_0, q_1]$ $[q_2]$ $[q_3]$

2-equivalence :

$[q_0]$ $[q_1]$ $[q_2]$ $[q_3]$

∴ the final minimized automaton would be -



Minimization of DFA

<u>equivalent states</u> : Two states p and Q are equivalent if the following conditions do hold.

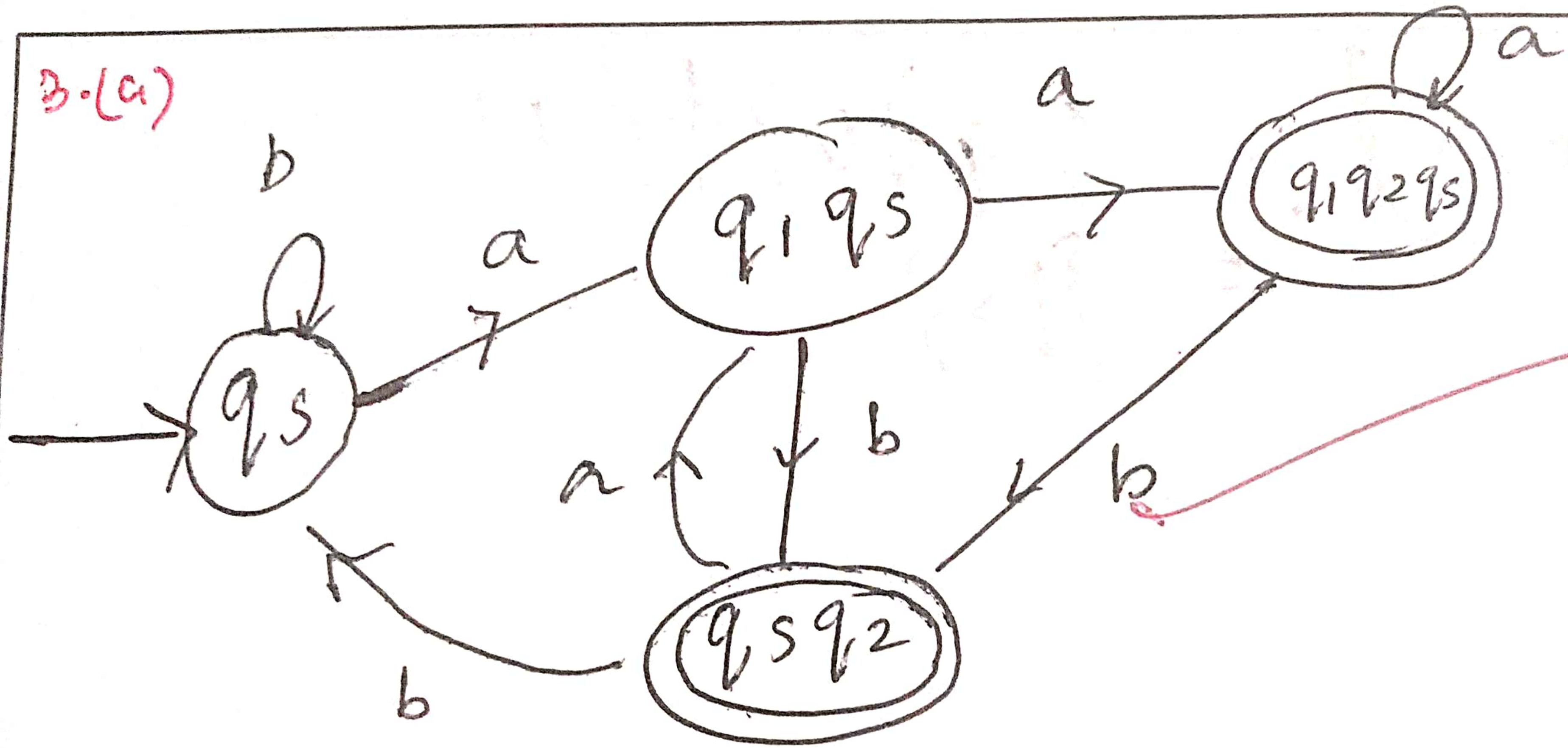
$\forall w \ \hat{\delta}(P, w) \in F \Leftrightarrow \hat{\delta}(q, w) \in F$

and

$\forall w \ \hat{\delta}(P, w) \notin F \Leftrightarrow \hat{\delta}(q, w) \notin F$

where $w \in \Sigma^*$, is an input string, δ : transition function $\times \Sigma \to Q$, $\hat{\delta}$-extended transition function, Q is a finite set of states and $p, q \in Q$, is a finite set of final states.

3.(a)



D

	0	1
q_s	$\{q_1, q_s\}$	$\{q_s\}$
q_1	$\{\phi\}$	$\{q_2\}$
q_2	$\{\phi\}$	$\{\phi\}$

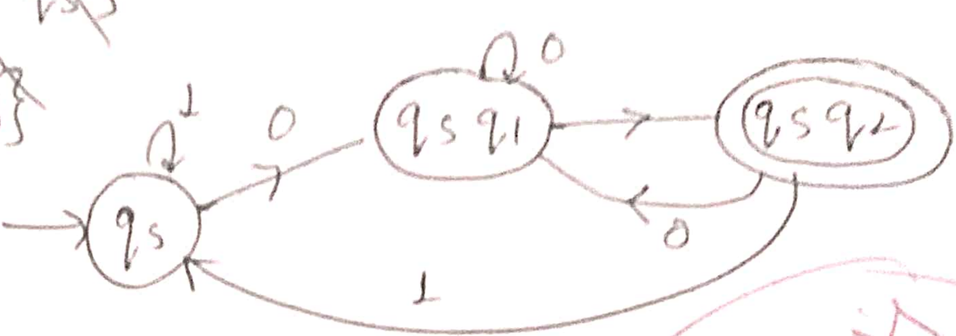
DFA

	0	1
$\rightarrow q_s$	$[q_1 q_s]$	$[q_s]$
$[q_1 q_s]$	$[q_1 q_s]$	$[q_s q_2]$
$* [q_s q_2]$	$[q_1 q_s]$	$[q_s]$

$\cancel{\delta(q_s, 0) = \{q_1\ q_s\}}$

$\cancel{\delta(q_s, 1) = \{q_s\}}$

$\cancel{\delta\ q_s,}$



~~Processing of 00100 →~~

$\cancel{\delta(q_s, 0) = [q_s q_1]}$

$\delta(q_s, 0) = \{q_s \cdot q_1\}$

$\delta\ \delta(q_s, 1) = \{q_s\}$

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

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

$\{q_s, q_2\}$

$\delta(q_s, 0) \cup \delta(q_2, 0) = \{q_1, q_s\} \cup \phi = \{q_1, q_s$

	0 a	b 1
q_s	$\{q_1, q_s\}$	$\{q_s\}$
q_1	$\{\phi\}$	$\{q_2\}$
q_2	$\{\phi\}$	$\{\phi\}$

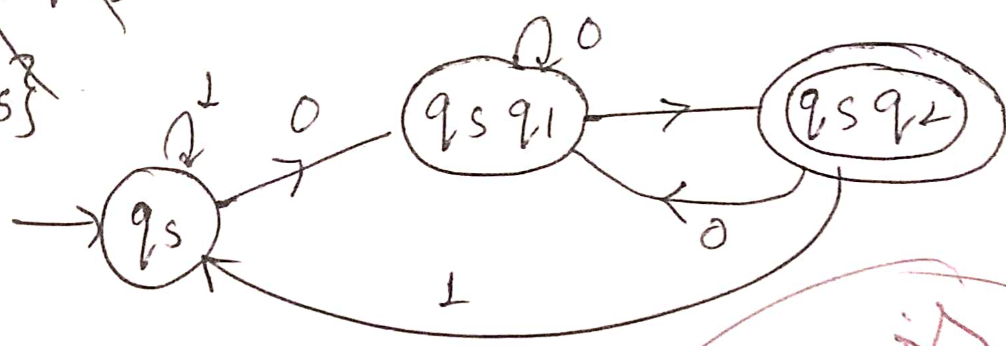
DFA

	a 0	1
$\rightarrow q_s$	$[q_1 q_s]$	$[q_s]$
$[q_1 q_s]$	$[q_1 q_s]$	$*[q_s q_2]$
$*[q_s q_2]$	$[q_1 q_s]$	$[q_s]$

$\cancel{\delta(q_s, 0) = \{q_1\ q_s\}}$

$\cancel{\delta(q_s, 1) = \{q_s\}}$

$\cancel{\delta\ q_s,}$



~~Processing of 00100 →~~

$\cancel{\delta(q_s, 0) = [q_s q_1]}$

$\delta(q_s, 0) = \{q_s \cdot q_1\}$

$\delta(q_s, 1) = \{q_s\}$

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

$= \{q_1, q_s\}$

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

$\{q_s, q_2\}$

~~what is this~~

$\delta(q_s, 0) \cup \delta(q_2, 0) = \{q_1 q_s\} \cup \phi = \{q_1 q_s\}$

Topic 2

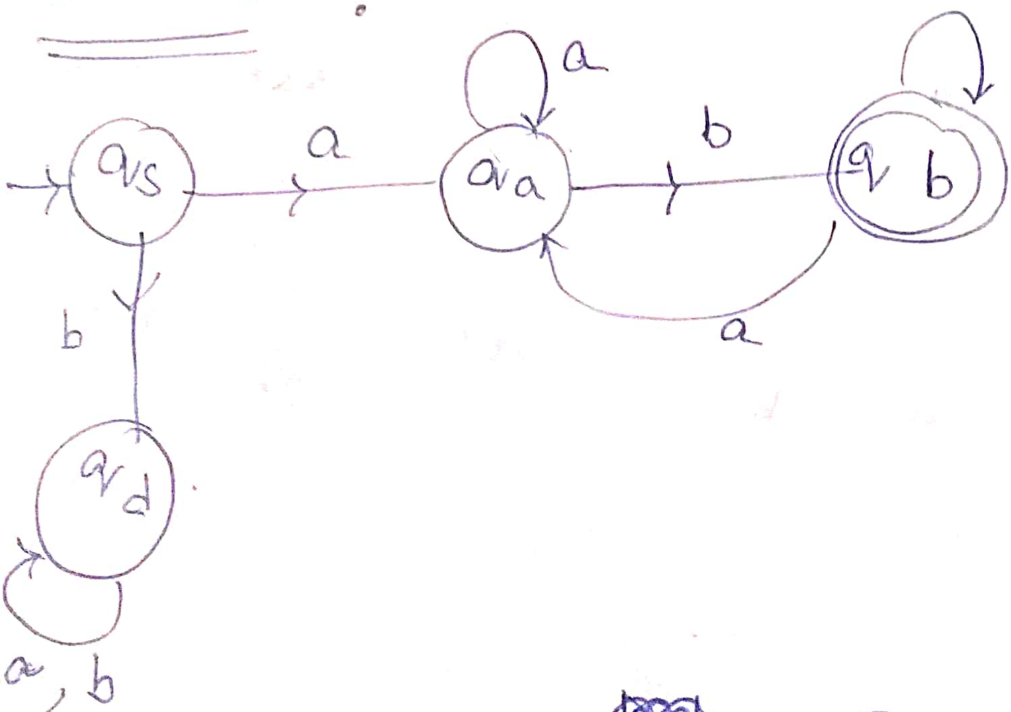
Union :

(5) - DFA - start & end with different symbol. $\Sigma = \{a, b\}$

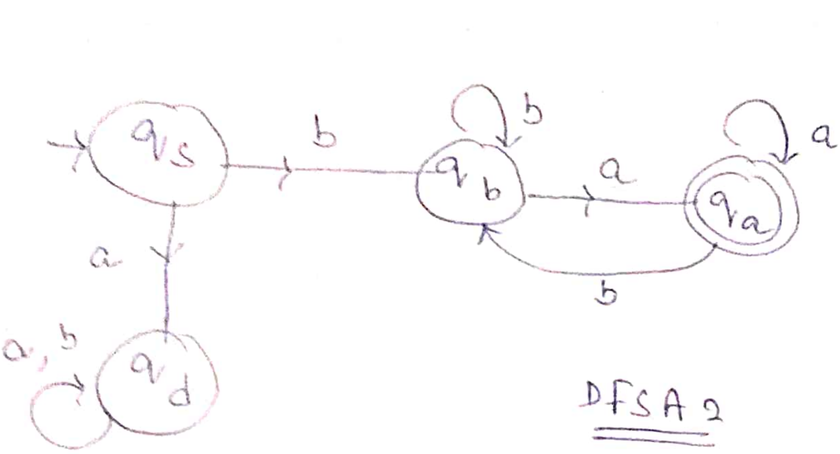
$L_1 = \{ab, aab, abb, \ldots\}$ - starts with a and ends with b.

$L_2 = \{ba, baa, bba, \ldots\}$ - starts with b and ends with a.

for L_1 :

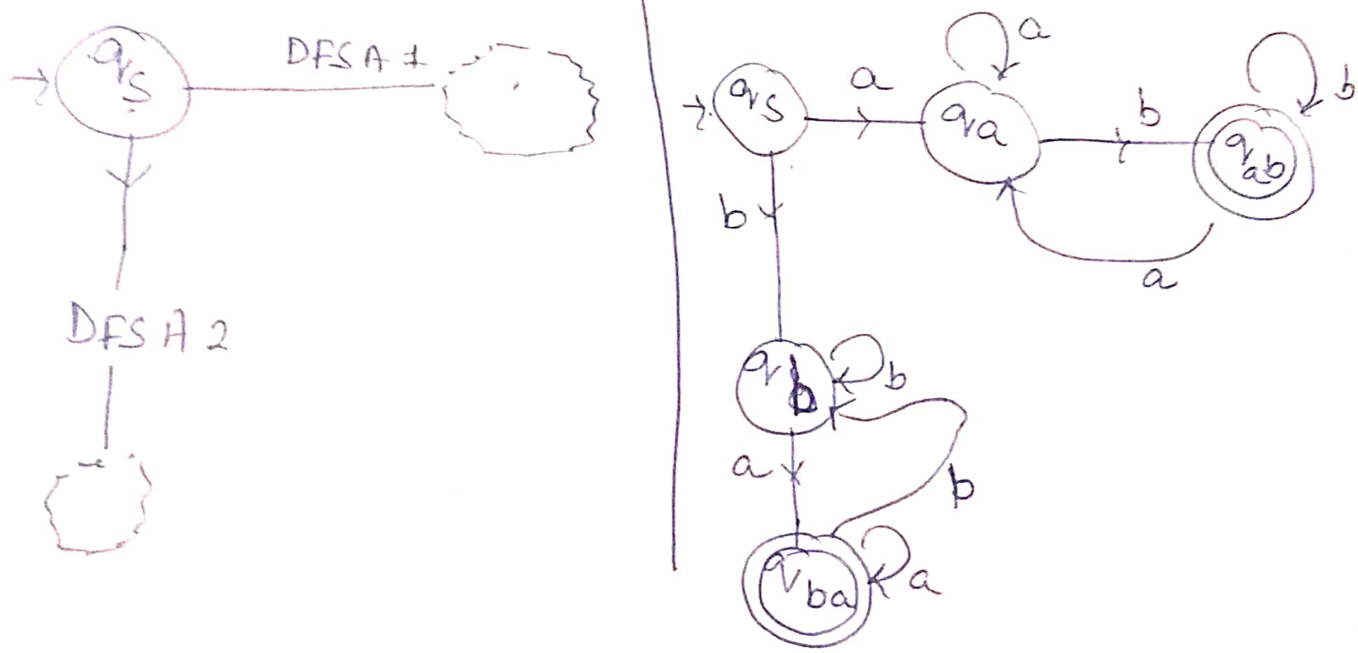


DFSA 1



DFSA 2

Now $L_1 \cup L_2$:



- Set of all strings - starts and ends with different symbol.

(A) final states of DFA1 and DFA2 - will remain final states.

(B) start states of DFA1 and DFA2 _ will remai start states.

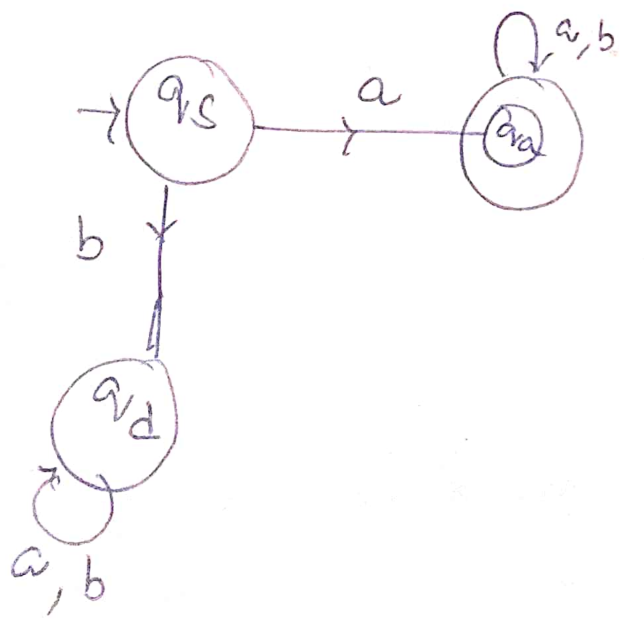
Concatenation

L_1 = {set of all strings - starts with a
L_2 = " " " " - ends " b

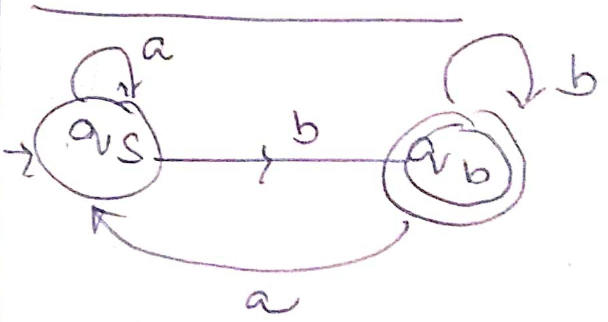
$L_1 = \{a, aa, ab, abb, \ldots\}$
$L_2 = \{b, ab, bb, abb \ldots\}$

$L_1 \cdot L_2 = \{ab, aab, aab, aaab \ldots\}$
= Set of all string - starts with 'a' and ends with 'b'.

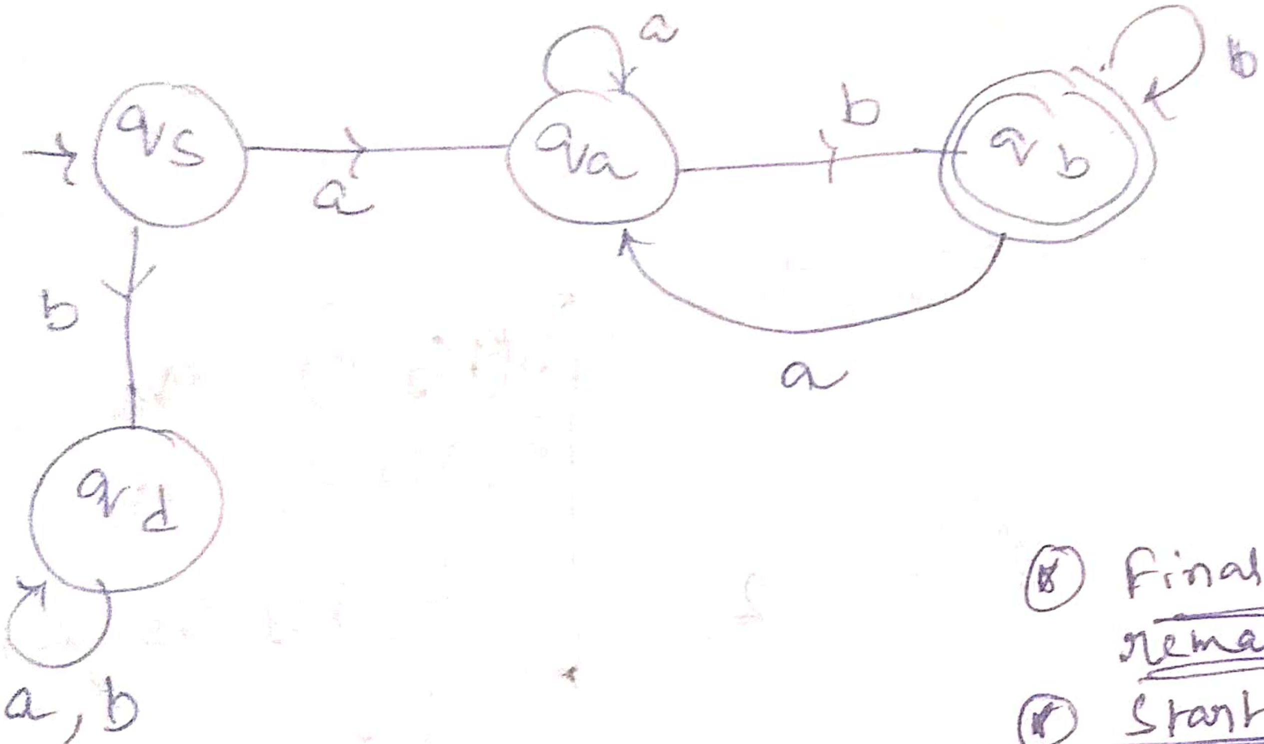
DFA1 for L1 :



DFA2 for L2



Now : DFA1 . DFA2 . for $L_1 \cdot L_2$.



* Need to make a connection between the final state of DFA1 and starting state of DFA2.

* Final state of DFA1 - won't remain final state.

* Start state of DFA2 - won't remain start state.

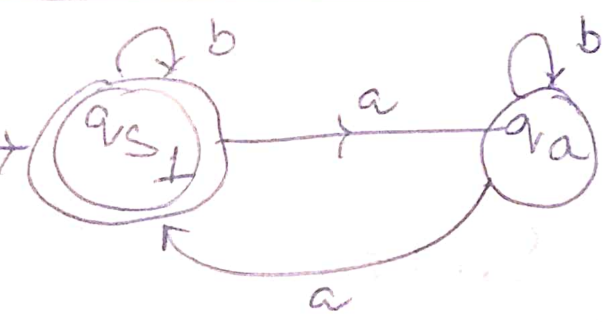
Cross Product

$L_1 = \{b, bb, \epsilon, aa, baa, aba, \ldots\}$

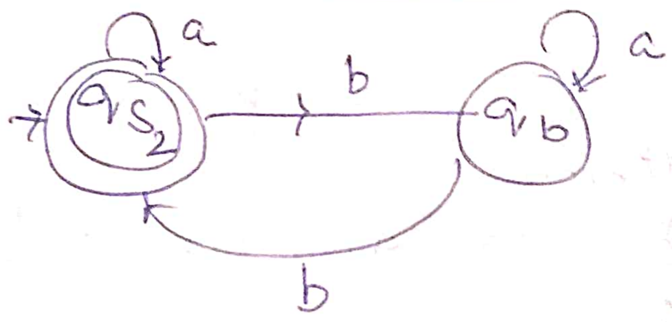
= Set of all strings containing even no of a's.

$L_2 = \{a, aa, \epsilon, bb, abb, bab, \ldots\}$

DFA 1 for L_1



DFA 2 for L_2



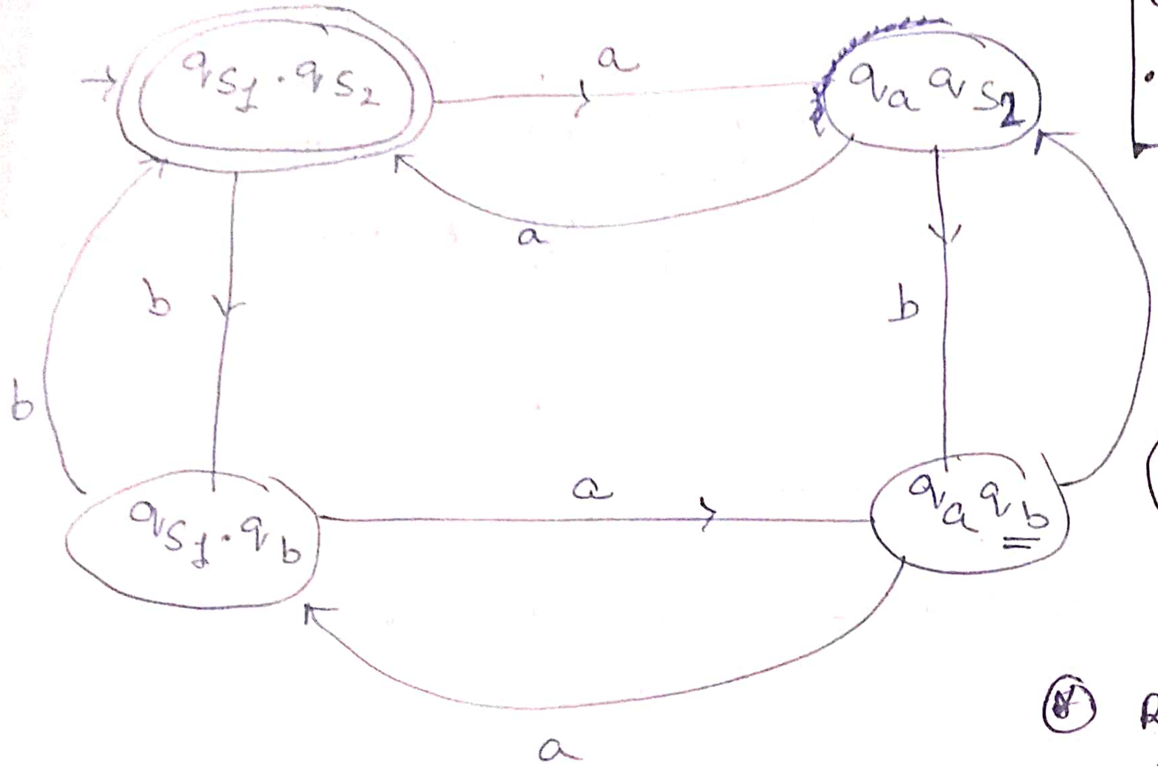
DFA$_3$ for cross product of L1 L2

$\{q_{s_1}, q_a\} \times \{q_{s_2}, q_b\}$

$= \{q_{s_1} q_{s_2}, q_{s_1} q_b, q_a q_{s_2}, q_a q_b\}$

$\delta(q_{s_1}, a) = q_a$
$\delta(q_{s_2}, a) = q_{s_2}$
$\therefore \delta(q_{s_1} \cdot q_{s_2}, a)$
$= (q_a, q_{s_2})$

$\delta(q_{s_1}, b) = q$
$\delta(q_{s_2}, b) = q$
$\therefore \delta(q_{s_1} q_{s_2}, b)$
$= (q_{s_1} \cdot q_b$

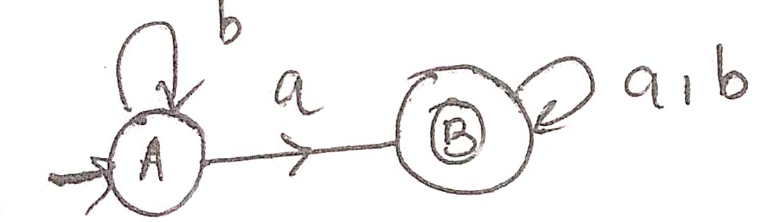


Rest are done in the same way.

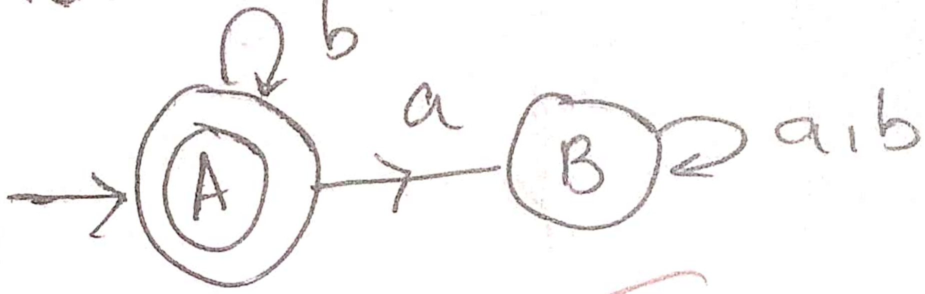
Complement

: Make a swap between ~~start~~ final state and non final states.

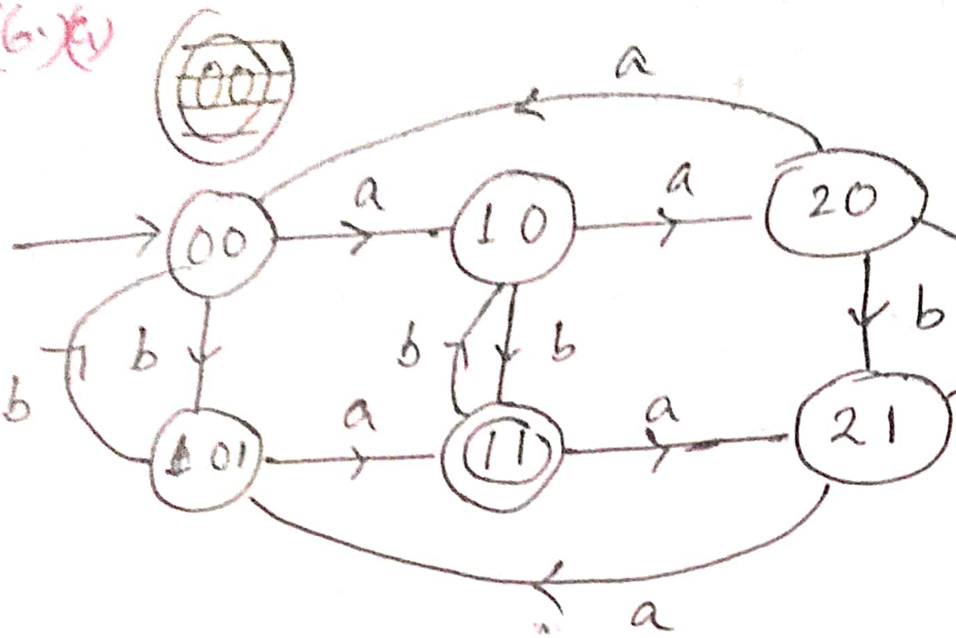
$\sim L_1$ - not contain a.



$\sim L_1$



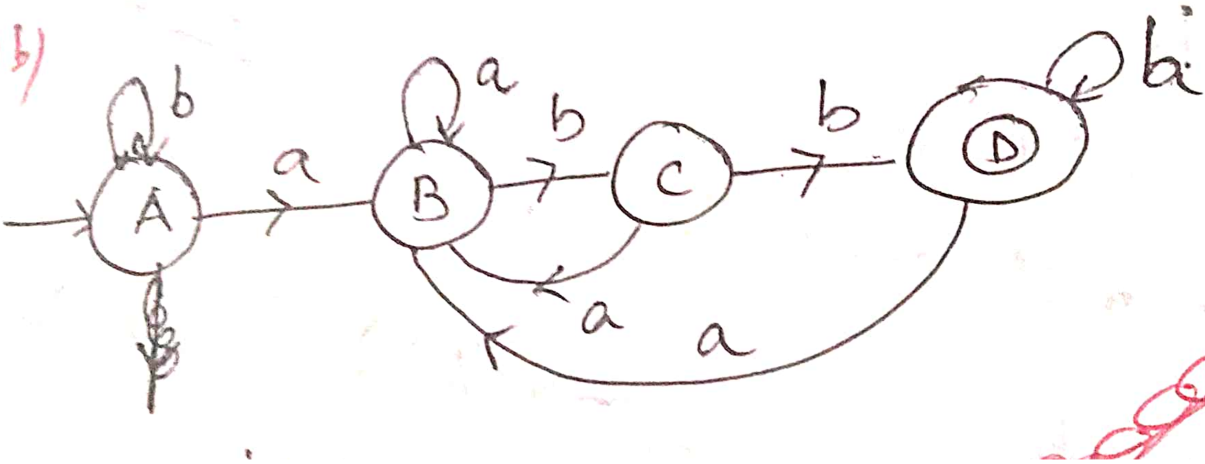
(6.)(v)



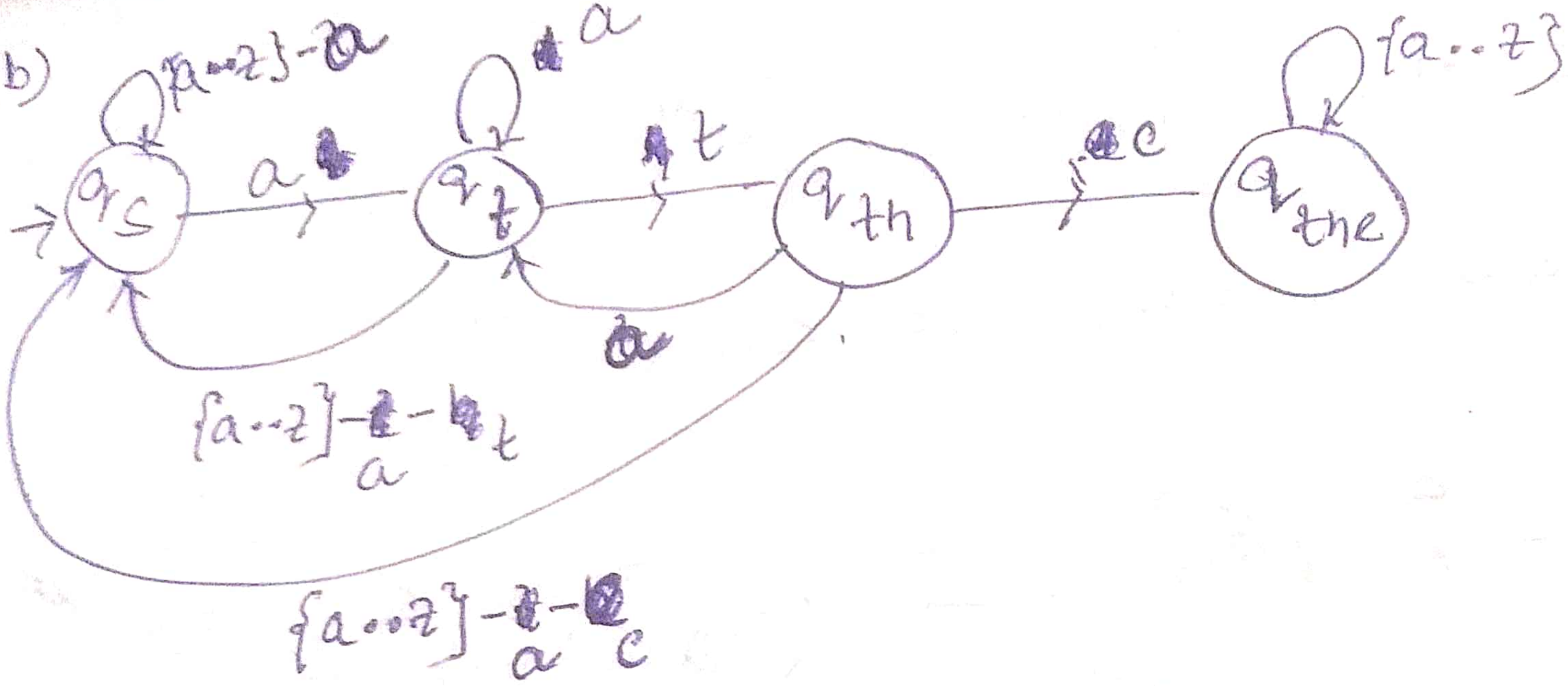
11 →

1. a followed by bb (at least)

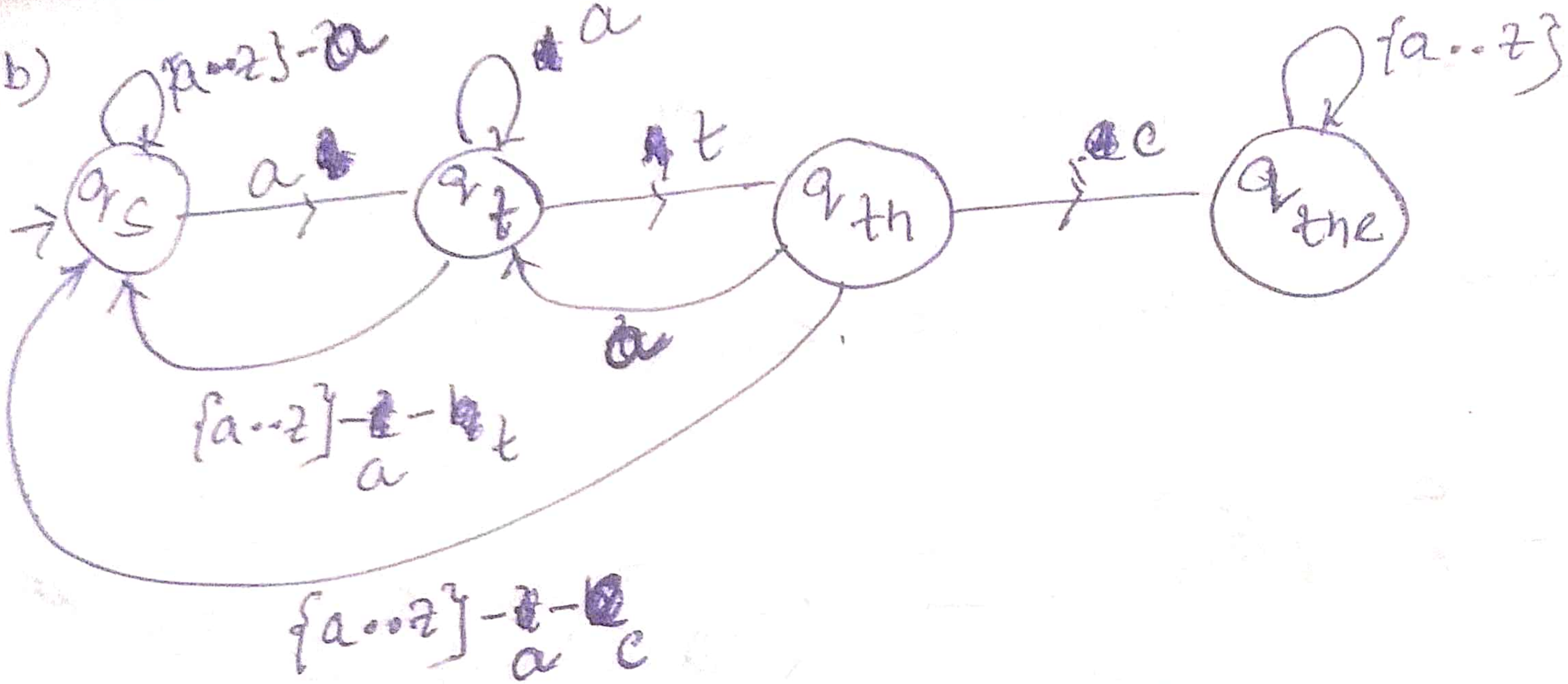
(6.) b)



2(b)



2(b)



USN | | | | | | | | | | |

CMRIT — CMR INSTITUTE OF TECHNOLOGY, BENGALURU. ACCREDITED WITH A+ GRADE BY NAAC

Internal Assessment Test 1 – November 2021

Sub:	Automata Theory and Computability					Sub Code:	18CS54	Branch:	ISE
Date:	13/11/2021	Duration:	90 min's	Max Marks:	50	Sem/Sec:	V A, B & C		OBE

Answer any FIVE FULL Questions — MARKS | CO | RBT

		MARKS	CO	RBT
1)	a) Write down the formal definition of NFA (Non deterministic finite automata), explain extended transition function and language acceptance of NFA.	[4]	CO1	L2
	b) Prove the equivalence between NFA and DFA. (Deterministic finite automata)	[6]	CO1	L2
2)	Construct minimal DFA for the following languages. a) $L = \{a^3bwa^4 \mid w \in (a+b)^*\}, \sum = \{a,b\}$. b) Set of strings containing "atc" anywhere in a string of {a-z}, that is, $\sum = \{a,..,z\}$. E.g., "atmeatcpqat" will be accepted but not "axydaa" etc	[5+5=10]	CO3 CO3	L3 L3
3)	a) Convert the following NFA to DFA. Here q_s is the starting state, q_2 is the final state, and $\{\Phi\}$ denotes the null set.	[5+5=10]	CO2	L3

	a	B
q_s	$\{q_1, q_s\}$	$\{q_s\}$
q_1	$\{q_2\}$	$\{q_2\}$
q_2	$\{\Phi\}$	$\{\Phi\}$

(b) Consider the transition table of the following NFA. Describe the processing of the input string '00100' using transition function and extended transition function. — L3

Here q_s is the starting state, q_2 is the final state

	0	1
q_s	$\{q_1, q_s\}$	$\{q_s\}$
q_1	$\{\Phi\}$	$\{q_2\}$
q_2	$\{\Phi\}$	$\{\Phi\}$

		MARKS	CO	RBT
4)	(a) Define equivalent states of a DFA. (b) Consider the following transition of table of a DFA and apply minimization algorithm on it to find out the minimal DFA. Here q_s is the starting state, q_3 is the final state.	[3+7=10]	CO2 CO2	L2 L3

	a	b
q_s	q_1	q_s
q_1	q_s	q_2
q_2	q_3	q_1
q_3	q_3	q_s
q_4	q_3	q_5
q_5	q_6	q_4
q_6	q_7	q_5
q_7	q_6	q_3

5	Discuss the following operations of DFA with example. (a) Union (b) Concatenation (c) Cross product (d) Complement	[2.5+2.5+2.5+2.5=10]	CO2	L2
6	Construct DFA for the following languages. a) Set of all strings over ∑ = {a,b} where (number of a)%3 = 1 and (number of b)%2 = 1. Here % denotes remainder. E.g : aaaabbb – accepted, aaaaaaabbbbb – accepted, aaabbb – rejected, aaaabbbb – rejected etc	5+5=10	CO3	L3
	b) Set of all strings where every ‘a’ is immediately followed by at least two ‘bs’, ∑ = {a,b}. E.g : bbabbaba – it will be rejected, bbabbabbbb – will be accepted etc		CO3	L3

Faculty Signature CCI Signature HOD Signature