COMPUTER ENGINEERING DEPARTMENT

ASSIGNMENT NO-01

Sub: Theory of Computer Science

COURSE: T.E. Year: 2020-2021 Semester: V
DEPT: Computer Engineering
SUBJECT CODE: CSC504 DUE DATE: 05/11/2020

==

Roll No. 50 Name: Amey Thakur

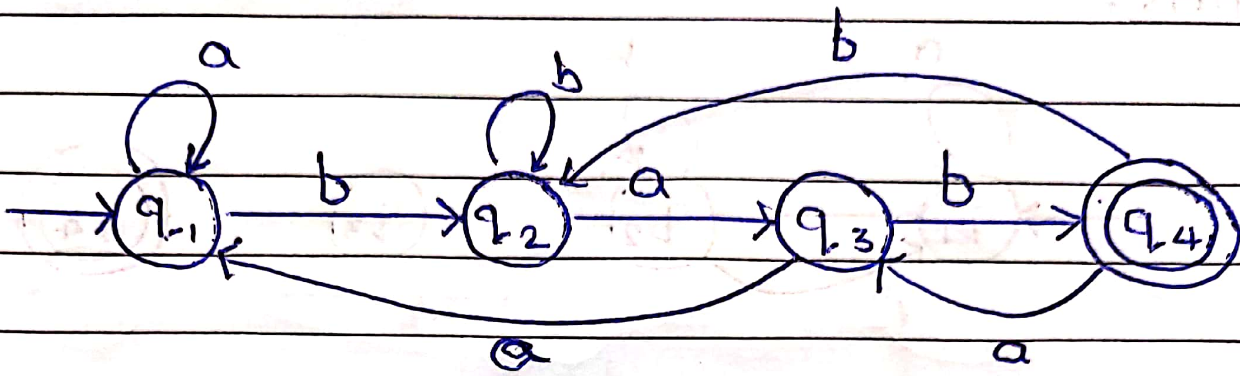
Class: TE-Comps B Date of Submission: 04/11/2020

Tutorial 1

1. Design FSM in which input is valid if it ends in 'bab' over $\sum$ ={a,b}
2. Design FSM to implement binary adder over $\sum$ ={0,1}
3. Design FSM to recognize a language in which every a's followed by b's over $\sum$ ={a,b}
4. Design FSM in which input is valid if it contains 1011 over $\sum$ ={0,1}
5. Design FSM in which input is valid if it does not contains 'bbb' over $\sum$ ={a,b}
6. Design FSM in which input is valid if it starts with 3 consecutive á's over $\sum$ ={a,b}

Q.1. Design FSM in which input is valid if it ends in 'bab' over $\Sigma = \{a, b\}$

Soln :



- The finite automation M_1

We can define M_1 formally by writing $M_1 = (Q, \Sigma, \delta, q_1, F)$ where

1. $Q = \{q_1, q_2, q_3, q_4\}$
2. $\Sigma = \{a, b\}$
3. δ is described as

	a	b
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_1	q_4
q_4	q_3	q_2

4. q_1 is the start state
5. $F = \{q_4\}$

Q.1

Simulation (string : abab)

$\rightarrow \delta(q_1, abab)$

$\rightarrow \delta(q_1, bab)$

$\rightarrow \delta(q_2, ab)$

$\rightarrow \delta(q_3, b)$

$\rightarrow \delta(q_4)$

$\because q_4$ is Final state

Hence Accepted

Q.2. Design FSM to implement binary adder over $\Sigma = \{0, 1\}$

Sol^n:

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

$q_0 \Rightarrow$ No Carry (NC)	$q_1 \Rightarrow$ Carry (C)
0 + 0 + NC = 0 NC	0 + 0 + C = 1 NC
0 + 1 + NC = 1 NC	0 + 1 + C = 0 C
1 + 0 + NC = 1 NC	1 + 0 + C = 0 C
1 + 1 + NC = 0 C	1 + 1 + C = 1 C

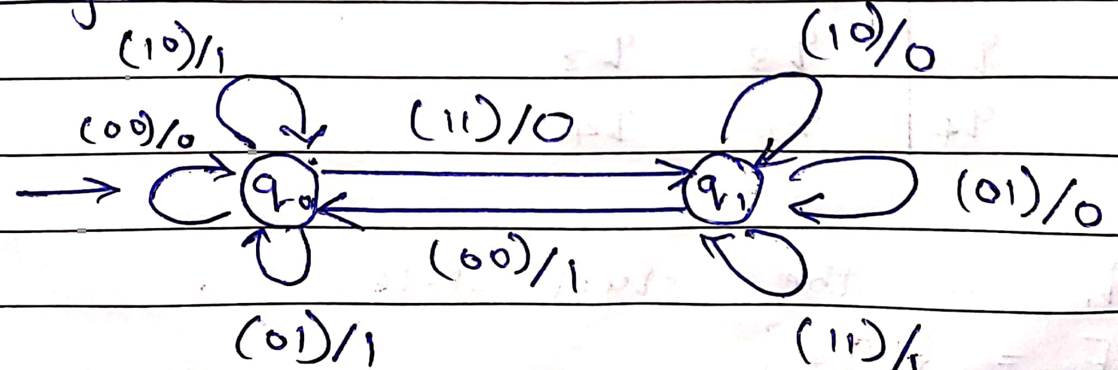
δ :

	(0,0)	(0,1)	(1,0)	(1,1)
q_0	q_0	q_0	q_0	q_1
q_1	q_0	q_1	q_1	q_1

λ :

	(0,0)	(0,1)	(1,0)	(1,1)
q_0	0	1	1	0
q_1	1	0	0	1

Diagram:



Q.2

Simulation [Add $(6)_{10}$ & $(3)_{10}$]

$\rightarrow (q_0, 110, 011)$

$\rightarrow (q_0, 10, 11) \rightarrow 1$

$\rightarrow (q_1, 0, 1) \rightarrow 0 \; c$

$\rightarrow (q_1, \epsilon, \epsilon) \rightarrow 0 \; c$

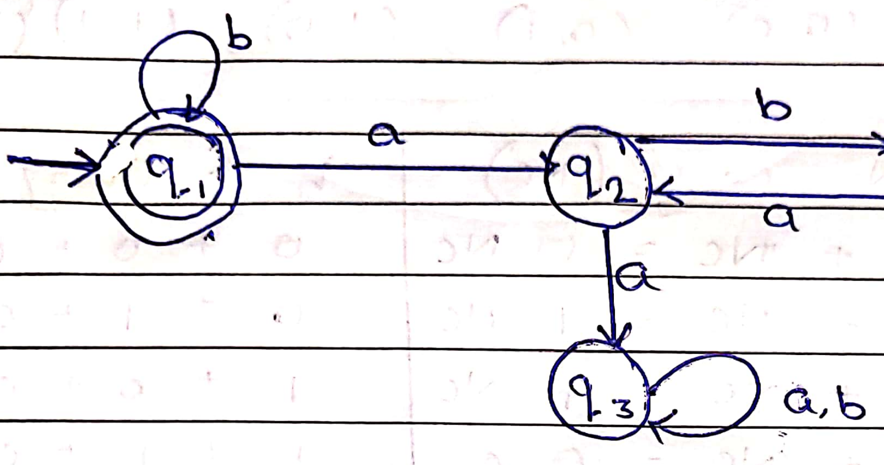
1

(1001)

$\Rightarrow (1001) = (9)_{10}$

Q.3. Design FSM to recognize a language in which every a's followed by b's over $\Sigma = \{a, b\}$

Soln:



- The finite automation M_1

We can define M_1 formally by writing $M_1 = (Q, \Sigma, \delta, q_1, F)$ where

① $Q = \{q_1, q_2, q_3, q_4\}$

② $\Sigma = \{a, b\}$

③ δ is described as

	a	b
q_1	q_2	q_1
q_2	q_3	q_4
q_3	q_3	q_3
q_4	q_2	q_4

④ q_1 is the start state

⑤ $F = \{q_1, q_4\}$

Q3

Simulation (string: abab)

→ $\delta(q_1, abab)$

→ $\delta(q_2, bab)$

→ $\delta(q_4, ab)$

→ $\delta(q_2, b)$

→ $\delta(q_4)$

∵ q_4 is final state

Hence Accepted

or

Simulation (string: b)

→ $\delta(q_1, b)$

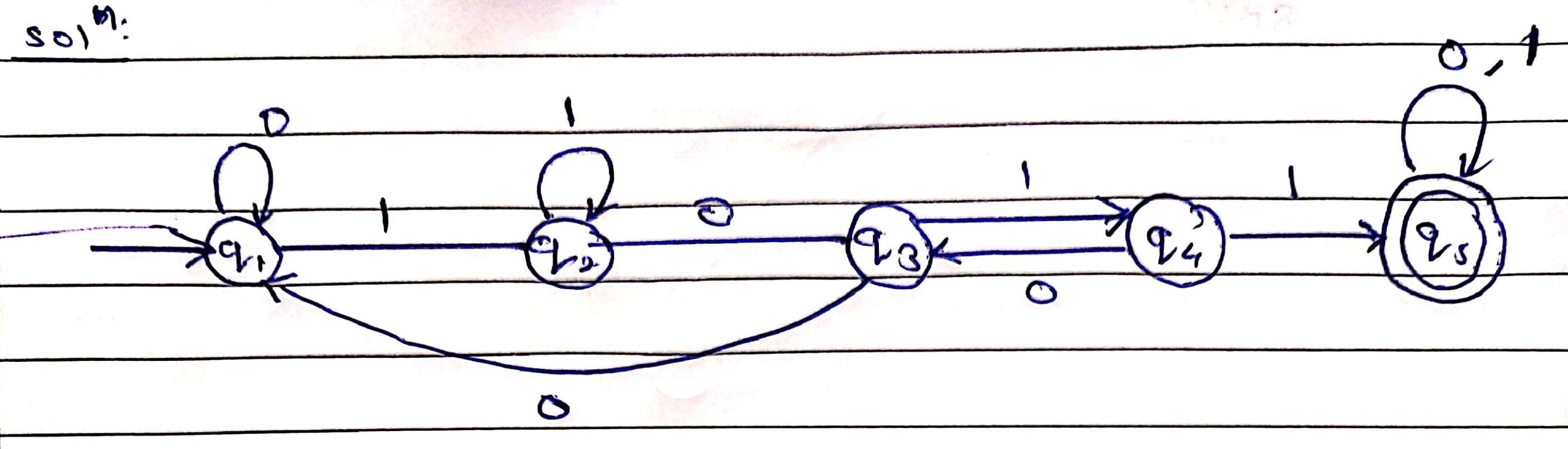
→ $\delta(q_1)$

∵ q_4 is final state

Hence Accepted

Q.4 Design FSM in which input is valid if it contains 1011 over $\Sigma = \{0, 1\}$

solⁿ:



- The finite automation M_1

We can define M_1 formally by writing where
$M_1 = (Q, \Sigma, \delta, q_0, f)$

① $Q = \{q_1, q_2, q_3, q_4, q_5\}$
② $\Sigma = \{0, 1\}$
③ δ is described as

	0	1
q_1	q_1	q_2
q_2	q_2	q_3
q_3	q_1	q_4
q_4	q_3	q_5
q_5	q_5	q_5

④ q_1 is the start state
⑤ $F = \{q_5\}$

Q 4

Simulation (string : 1010110)

$\rightarrow \delta(q_1, 1010110)$

$\rightarrow \delta(q_2, 010110)$

$\rightarrow \delta(q_3, 10110)$

$\rightarrow \delta(q_4, 0110)$

$\rightarrow \delta(q_3, 110)$

$\rightarrow \delta(q_4, 10)$

$\rightarrow \delta(q_5, 0)$

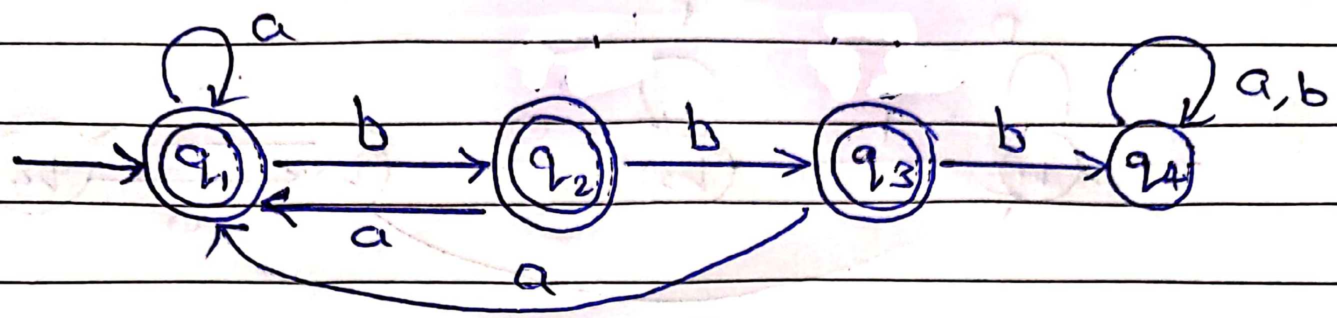
$\rightarrow \delta(q_5)$

$\because q_5$ is Final state

Hence Accepted

Q.5. Design FSM in which input is valid if it does not contains 'bbb' over $\Sigma = \{a, b\}$

Soln:



- The finite automation M_1

We can define M_1 formally by writing $M_1 = (Q, \Sigma, \delta, q, f)$ where

① $Q = \{q_1, q_2, q_3, q_4\}$

② $\Sigma = \{a, b\}$

③ δ is described as

	a	b
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_1	q_4
q_4	q_4	q_4

④ q_1 is the start state

⑤ $F = \{q_1, q_2, q_3\}$

Q5

Simulation (string: abbb)

→ $\delta(q_1, abbb)$
→ $\delta(q_1, bbb)$
→ $\delta(q_2, bb)$
→ $\delta(q_3, b)$
→ $\delta(q_4)$

∵ q_4 is not final state

Hence **not Accepted**

or

Simulation (string: abab)

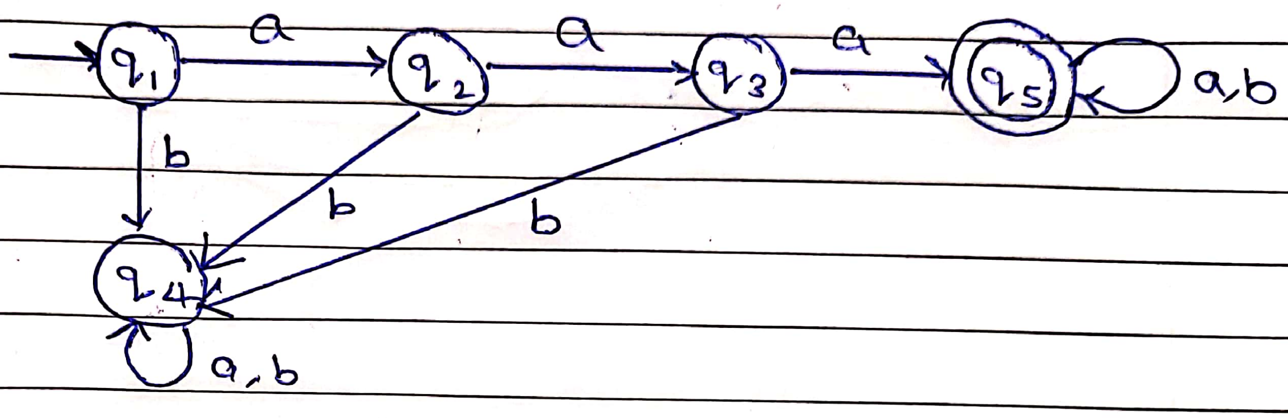
→ $\delta(q_1, abab)$
→ $\delta(q_1, bab)$
→ $\delta(q_2, ab)$
→ $\delta(q_2, b)$
→ $\delta(q_2)$

∵ q_2 is final state

Hence **Accepted**

Q6. Design FSM in which input is valid if it starts with 3 consecutive a's over $\Sigma = \{a, b\}$

Sol^n:



- The finite automation M_1

We can define M_1 formally by writing $M_1 = (Q, \Sigma, \delta, q_0, F)$ where

1. $Q = \{q_1, q_2, q_3, q_4, q_5\}$
2. $\Sigma = \{a, b\}$
3. δ is described as

	a	b
q_1	q_2	q_4
q_2	q_3	q_4
q_3	q_4	q_4
q_4	q_4	q_4
q_5	q_5	q_5

4. q_1 is the start state
5. $F = \{q_5\}$

Q.6

Simulation (string : aaba)

→ $\delta(q_1, aaba)$
→ $\delta(q_2, aba)$
→ $\delta(q_3, ba)$
→ $\delta(q_4, a)$
→ $\delta(q_4)$

∵ q_4 is not final state

Hence not Accepted

or

Simulation (string : aaaba)

→ $\delta(q_1, aaaba)$
→ $\delta(q_2, aaba)$
→ $\delta(q_3, aba)$
→ $\delta(q_5, ba)$
→ $\delta(q_5, a)$
→ $\delta(q_5)$

∵ q_5 is final state

Hence Accepted