Date: Aug 19, 2025 Lecture # 01 Day: Tuesday

Theory of Automata

Applications of TOA:

→ Compiler Construction

Language : collection of words. ← collection of alphabets

→ set → distinct elements

Language Construction

Base Condition | Prior Language

$L_1 = \{a, b\}$ $\Sigma = \{a, b\}$ → set of alphabets

$|L_1| = n(L_1) = 2$ cardinality

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

$L = \{x \mid x \in \Sigma ; 0 \leq |x| \leq 2\}$

$L = \{\Lambda, a, b, aa, ab, ba, bb\}$

$|L| = 7$

$\Sigma = \{a, b\}$ (Base)

$L = \{x \mid x \in \Sigma ;$ is a palindrome; $0 \leq |x| \leq 3\}$

$L_2 = \{\Lambda, a, b, aa, bb, aba, bab, bbb\}$ → aaa

$|L| = 9$

$L_1 \vee L_2$

$L_1 = \{x \mid x \in \Sigma ;$ is a even palindrome $0 \leq |x| \leq 3\}$

$L_1 = \{\Lambda, aa, ab, ba, bb\}$ $0 \leq |x| \leq 3$

$L = L_1 \cap L_2 = \{\Lambda, aa, ba\}$

Lecture # 02

Language

Date: Aug 20, 2025 — Day: Thursday

$x \rightarrow$ subsequence

Prefix, substring, suffix

* Make 10 words:

$$L = \{ x \mid x \in \Sigma^* ;\ x \text{ contains 'aba'} \}$$
$$\Sigma = \{a, b, c\}$$

= { aba, aaba, baba, caba, abaa, ababa, abac, ababa, baaba, acaba }

abba

aba is the subsequence

"x ends with 0"

$$L = \{ x \mid x \in \Sigma^* ;\ x \text{ as a no. is divisible by } 2 \}$$
$$\Sigma = \{0, 1\}$$

L = { 0, 00, 10, 000, 100, 010, 110, 0000, 1000, 0100, 0010 }

15 words

$\boxed{L = L_1 \cap L_2}$

$$L = \{ x \mid x \in \Sigma^* ;\ \overbrace{x \text{ as a no.} \div 2}^{L_1} \ \& \ \underbrace{\text{sum of digits also} \div \text{by } 2}_{L_2} \}$$

L = { 0, 00, 000, 110, 0000, 1100, 1010, 0110, 00000, 00110, 11000, 01010, 10010, 11110, 01100 }

Lecture # 03

Date: Aug 26, 2025 Day: Tuesday

Topic : Machine Creation

based on transi

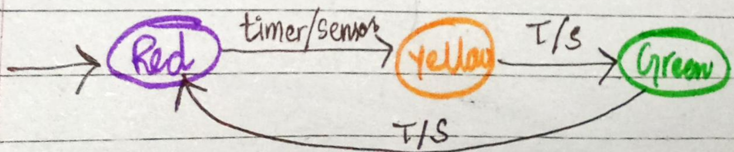
→ FSM (Finite state Machine) → DFA, NFA, NFA-A
→ FA (Finite Automata)
has 5 tuples

Σ — Set of Alphabets
Q — Set of States
q_0 — Start State
could be empty → f_n — Set of Final States
δ — Transition $Q \times \Sigma \rightarrow Q$

Σ = {timer, sensor}
Q = {Red, yellow, green}
q_0 = {Red}
f_n = {ϕ}
δ = δ(Red, timer) → yellow

Notations :

○ state
→○ start state
$(A) \xrightarrow{Y} (B)$ transition
Self Loop
◎ Final state



single choice ← DFA : Deterministic finite Auto-

Date:________ Day:________

$\delta(\text{Red}, \text{timer}) \rightarrow \text{Yellow}$
$\delta(\text{Red}, \text{sensor}) \rightarrow \text{Yellow}$ $\Big] \in Q$

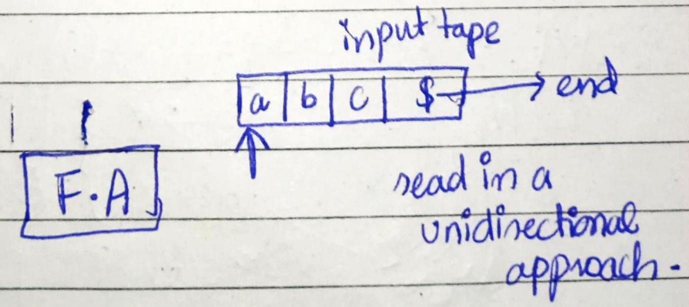
$\delta(\text{Yellow}, \text{timer}) \rightarrow \text{Green}$
$\delta(\text{Yellow}, \text{sensor} \rightarrow$ "

$\delta(\cancel{\text{Yellow}}\ \text{Green}, \text{timer}) \rightarrow \text{Red}$
$\delta(\text{Green}, \text{sensor}) \rightarrow \underline{\emptyset}$

* if there is a conflict b/w domain and range for the transition then it is not DFA.

// for DFA: $\underset{q \in Q}{\text{outDegree}(q)} = |\Sigma|$

* "The outdegree of any state should be equal to the cardinality of alphabet set."



Date:

Day:

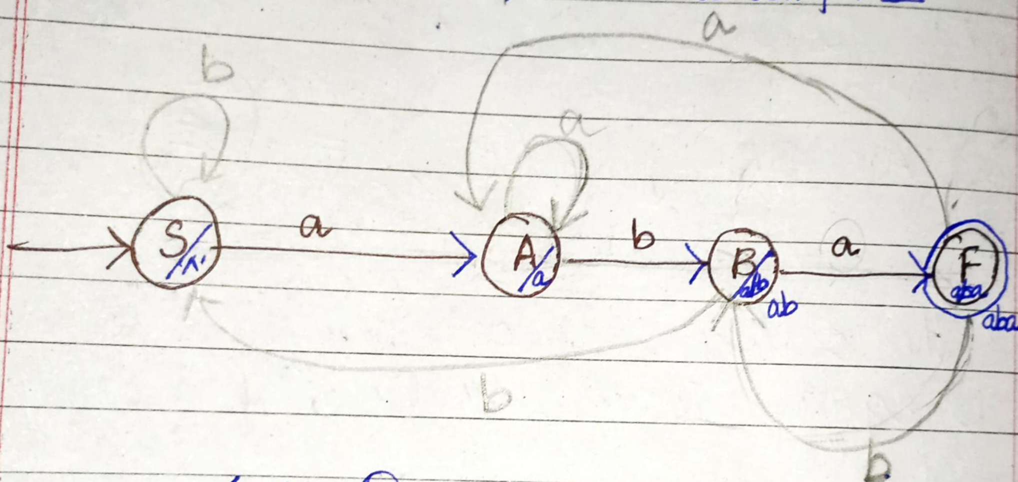
$\mathcal{L} = \{x \mid x \in \Sigma^* ;\ x \text{ ends with 'aba'}\}$

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

$L = \{ \overset{\Sigma^0}{aba},\ a\,aba,\ b\,aba,\ \overbrace{aa}^{\Sigma^2}aba,\ ababa,\ baaba,\ bbaba,\ ----- $ (above "aaba, baba": Σ^1)

$\Rightarrow \Sigma^* aba$

no. of state : |Smallest Pattern| + 1



$M = \{\Sigma, Q, q_0, f_n, \delta\}$

$L(M) = \{x \in \Sigma^* \mid \delta^*(q_0, x) \in f_n\}$

starting state

$(S, aba) \in f_n$.

Λ — $S \notin f_n$

a — $A \notin f_n$

aa — $A \notin f_n$

ab — $B \notin f_n$

aba — $F \in f_n$

$f_n = \{F\}$

aa

Date:__________ Day:__________

Longest Subsequence Matching

prefix ← Pattern | x → string : suffic

	'aba'	abb ^	abaa	abab
L = 0	^	^	^	^
L = 1	a	b	a	b
L = 2	ab	bb	aa	ab
L = 3	aba	abb	~~aa~~baa	bab

↓

contexual meaning of state

Lecture # 04

Date: 28 Aug 25 Day: Thursday

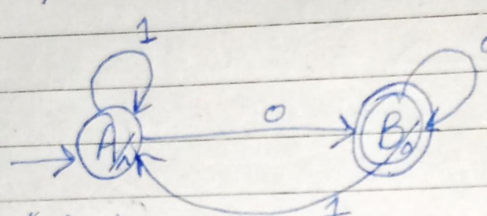
Construct a DFA for the given Language:

$L = \{x \mid x$ over Σ; x as a no. is divisible by 2 $\}$ // ends with '0'

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

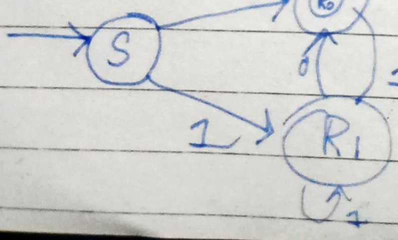
$L_{acc} = \{0, 00, 10, 000, 010, 100, 110, 0000, 0010, 0100, 1000\}$

no. of states = 2



	Pattern 0	suffix $\wedge 0$	$\wedge 1$	00	01
L = 0	$\wedge$	$0\wedge$	$\wedge$	$\wedge$	$\wedge$
L = 1	0	0.	1	0	1
L = 2	0	$\wedge 0$	$\wedge 1$	00	01
L = 3	0	$\wedge 0$	$\wedge 1$	00	01

$L_{rej} = \{\wedge, 1, 01, 11, 11, \ldots\}$



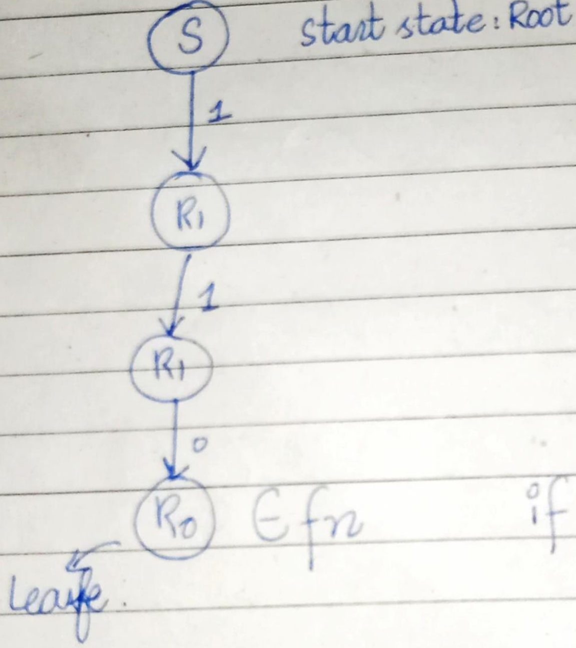
for divisibility by 2:
3 states

String Processing
has 2 methods

→ Trace Tree
→ Extented Transition Function δ^* [2.12]

Trace Tree:

110



if leaf of the Tree ...
if Ro belongs to final state so Accepted.

δ^*:

$\delta^*(q_0, x)$ breaks into yz

$\delta^*(S, 110)$

∵ Finite Automata process characters not strings.

(cut from suffix) → should be 1 char on suffix

$\delta^*(q_0, x) \longrightarrow \delta(\delta^*(S, 11), 0)$

simple transition function can be applied on 2 characters

string

∵ $\delta(\delta^*(q, y), z)$

$\delta(\delta(\delta^*(S, 1), 1), 0)$

$\delta(\delta(\delta(\boxed{\delta^*(S, \lambda)}, 1), 1), 0)$

Date:

$\delta^*(q, \wedge) = q_0$ extented eliminated from eq

$= \delta(\delta(\boxed{\delta(\delta,1)},1),0)$

$= \delta(\boxed{\delta(R_1,1)},0)$

$= \delta(R_1,0)$

$= R_0 \in f_n$ Accepted

for 1010

$\cancel{\delta(\delta(\delta(\delta^*(S,\wedge),1),}$

$\cancel{\delta(\delta^*(\delta,10),1)}$

$\delta(\delta^*(\delta,10),1)$

$\delta(\delta(\delta^*(S,1),0),1)$

$\delta(\delta(\delta^*(R_1,0),1))$

$\cancel{\delta(\delta}$ $\delta(R_0,1)$

R_1 Ans $\notin f_n$ Rejected.

[ends with 1 ✗

Date: Day:

$\mathcal{L} = \{x \mid x$ over Σ; x as a no $\div$ by 2 & sum of digits $\div$ 2$\}$

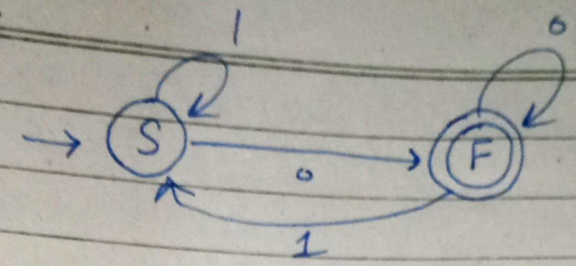
$\Sigma = \{0, 1\}$

$\mathcal{L} = \{0, 00, 000, 110, 0000, 0110, 1100, 1010, 00000, 01100, \}$

Date:

Day:

L_1 : M_1 :

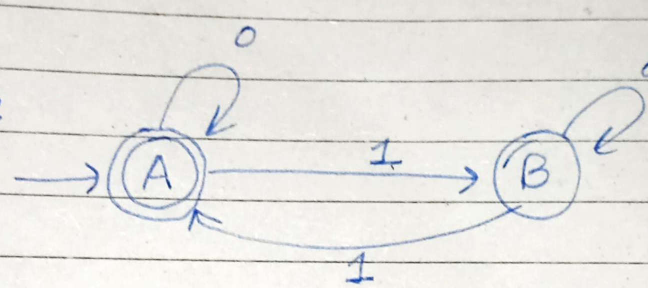


$Q_1 = \{S, F\}$; $q_{01} = S$

$\Sigma_1 = \{0, 1\}$; $f_n = \{F\}$

$\delta = Q_1 \times \Sigma_1 \to Q_1$

M_2 :



$Q_2 = \{A, B\}$; $q_{02} = A$

$\Sigma = \{0, 1\}$; $\delta = Q_2 \times \Sigma_2 \to Q_2$

$F = \{A\}$

$M = M_1 * M_2$

$\Sigma = \Sigma_1 = \Sigma_2$

$Q = Q_1 * Q_2$

$q_0 = q_{01} \, q_{02} \Rightarrow SA$

$\delta = \delta_1 \delta_2 \Rightarrow Q_1 Q_2 * \Sigma \to Q_1 Q_2$

Tabular Method

Day:

Date:

M_1 :

(states) Q

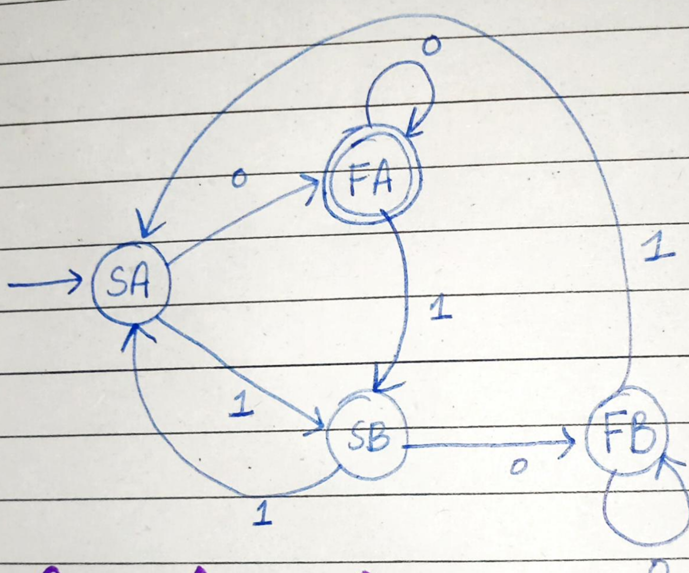
	Σ (alphabets) 0	1
-S	F	S
+F	F	S

M_2 :

Q

	Σ (alphabets) 0	1
$\mp$ A	A	B
B	B	A

$fn = M_1 \cap M_2$



$fn = fn_1 * fn_2$

↳ cartesian prod.

* $M_1 \cap M_2'$

$fn = fn_1 * (Q - fn_2)$

* $M_1 \cup M_2$

$fn = (fn_1 * Q_2) \cup (fn_2 * Q_1)$