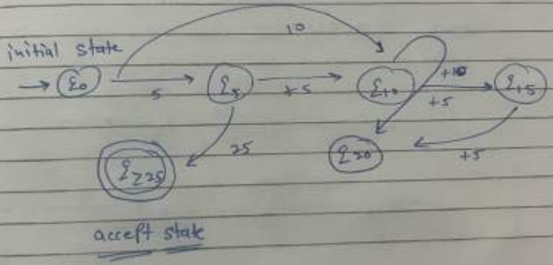
Operating System

1 chapter 1. Introduction

updates coverage of multicore systems, as well as new coverage of NUMA system and Hadoop clusters. old material has been updated, and new motivation has been added for the study of operating systems.

Theory of computing 16-Jan-2026



Initial state and accept state of machine.

if drop (5) cent then £0 to £5

if drop 10 cent then £0 to £10

switches will be done according to amount then stage varied.

figure 1.1

Ex1 → if string 1101 : left to Right

1

$\rightarrow q_1 \xrightarrow{1} q_2 \xrightarrow{1} q_2 \xrightarrow{0} q_3 \xrightarrow{1} q_2$

It ends up to at q_2 which is accepted state.

Ex II 2 0101010

$\rightarrow q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_3 \xrightarrow{1} q_2$

$\rightarrow q_1 \xrightarrow{0} q_1 \xrightarrow{1} q_2 \xrightarrow{0} q_3 \xrightarrow{1} q_2 \xrightarrow{0} q_3 \xrightarrow{1} q_2 \xrightarrow{0} q_3$

that accept state.

Exts Imp

$\varepsilon \in \phi$ (Emty) not anot!

This stage machine read nothing. (machine be at initial state).

it be in initial state always ofen, and that become the final state.

$\rightarrow (q_1)$

if string is empty it be only in the initial state.

formal def. of finite automata.

Q : finite set state.

Σ : is alphabet, finite set of symbols.

δ : $Q \times \Sigma \rightarrow Q$

$Q \times \Sigma \rightarrow Q$: Transition f^n

$q_0 \in Q$: start state,

$F \subset Q$: accept state / states

$M = \{Q, \Sigma, \delta, q_0, F\}$ look figure 1.1

$Q = \{q_1, q_2, q_3\}$ → set of state

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

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

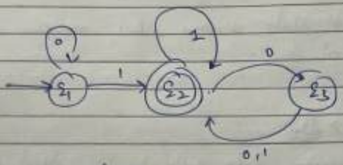
$\delta \rightarrow$ Transition f^n

Transition

very very imp

δ =

	0	1
q_1	q_1	q_2
q_2	q_3	q_2
q_3	q_2	q_2



language of finite set automation

input str : $w = w_1, w_2, w_3 - - w_n$

Sequence of state: $r_0, r_1, r_2, - - r_n$

$r_0 = q_1$

for $i = 0 - - - n-1$, $r_{i+1} = \delta(r_i, w_{i+1})$

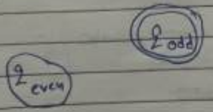
if $r_n \in F$: accepted w

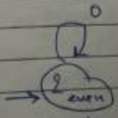
if $r_n \notin F$: rejected w.

special case if $n = 0 \Rightarrow w = \varepsilon$ = accept $\leftrightarrow r_0 \in F$

$A = \{ w : w$ has odd # of of 1's $\}$

any odd number of 1's 0111 ~~even~~ odd of 1's





ex :- String 0

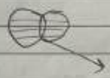
finite automate

30-Jan-2026

Lec-3 Operations & language......

DFA (Deterministic Finite Automata).....

$A \cup B$



union

$A \cap B$

inter-section

#4 A B are regular then Concatenation AB regular

A = 100 → B = 111

AB → <u>100111</u> → concatenation –

#5 if A is regular then A^* is regular.

put them in pieces. 0, 01, 11

Given → 001

ex → 00111 → anything from A.

- union : $A_1 \cup A_2 = \{w : w \in A_1 \text{ or } w \in A_2\}$

intersection : $A_1 \cap A_2 = \{w : w \in A_1 \text{ and } w \in A_2\}$

concat – $AB = \{vw \mid v \in A, w \in B\}$

Page No.
Date: / /20

Star

$A^* = \{ w_1, w_2 \dots w_k \mid k \geq 0, \text{ and each } w_i \in A \}$

Complement :-

$\overline{A} = \{ w \in \Sigma \mid w \notin A \} = \Sigma - A$

A Finite Automaton (FA) :

Union (OR)

intersection (AND)

$M_1 = \{ Q_1, \Sigma, \delta_1, q_1, F_1)$ $M_2 = \{ Q_2, \Sigma, \delta_2, q_2, F_2)$

$M_3 = \{ Q_3, \Sigma, \delta_3, q_3, F_3 \}$

$Q_3 = Q_1 \times Q_2 = \{ (r_1, r_2) : r_1 \in Q_1, r_2 \in Q_2 \}$

$q_3 = (q_1 q_2)$

$F_3 = \{ (r_1, r_2) : r_1 \in F_1, \text{ OR } r_2 \in F_2 \}$

$\delta_3 ((r_1, r_2), a) = (\delta_1 (r_1, a), \delta_2 (r_2, a))$

Page No.
Date: / /202

Star of A: A^* is obtained by taking any finite number of strings in the origional language in A and glowing them to.

$A^* = \{u_1 u_2 u_3 \ldots u_k ; k \geq 0, u \in A\}$

$A = \{0, 10\} : A^* = \{\epsilon, 0, 10, 00, 010, 100, 1010, \ldots \ldots 10100100010\}$

Q? if i have A regular & * line this how i can .....

Non-Deterministic finite Automata.

01/02/2026

- union → $A \cup B$ → or
- Intersection $A \cap B$ → AND
- Concatanation AB
- Star: $A^* = \{w_1 w_2 \ldots w_k \mid k \geq 0$
- Complement: $\bar{A}$ → $\Sigma - A$

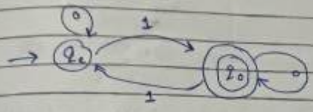
- Finite Automata (FA): $M = (Q, \Sigma, \delta, q, F)$.

Q = finite set of states
Σ - called the subeset, is a finite set of symbol
δ : $Q \times \Sigma \to Q$: Transition State
$q \in Q$: start state
$F \subset Q$: Accept state

for inf

Constructing DFA from Complement

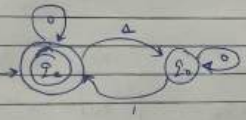
* In General, Given a DFA M for language A,



M

0 010,- -]

$\bar{M}$



$M = (Q, \Sigma, \delta, q_1, F)$

$\bar{M} = (Q, \Sigma, \delta, q, \Sigma - q)$

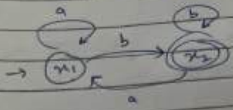
Ex Consider the following DFAs and languages over

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

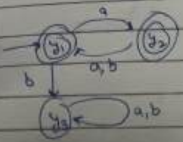
. DFA M_1 accept language $A_1 = L(M_1)$

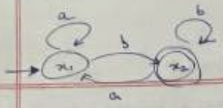
DFA M_2 accept language $A_2 = L(M_2)$

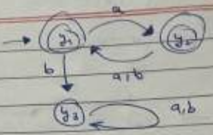
DFA M_1 for A_1



DFA M_2 for A_2



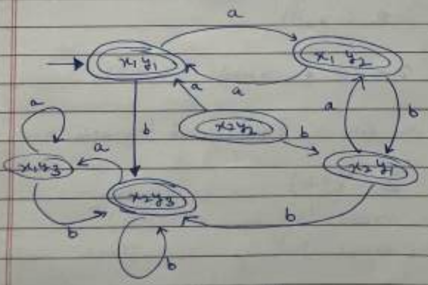




we now want DFA M_3 for $A_1 \cup A_2$ (UNION) OR $\Rightarrow$

<u>$A_1 \cup A_2$</u> lets go & figure out the Transition f^n

$x_1 \xrightarrow{a} x_1y_2$, $y_1 \xrightarrow{b} x_2y_3$ | $x_1 \xrightarrow{a} x_1y_1$, $y_2 \xrightarrow{b} x_2y_1$ | $x_1 \xrightarrow{a} x_1y_3$, $y_3 \xrightarrow{b} x_2y_3$

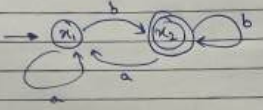


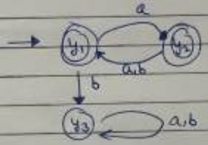
$x_2 \xrightarrow{a} x_1y_2$, $y_1 \xrightarrow{b} x_2y_3$ | $x_2 \xrightarrow{a} x_1y_1$, $y_2 \xrightarrow{b} x_2y_1$ | $x_2 \xrightarrow{a} x_1y_3$, $y_3 \xrightarrow{b} x_2y_3$

Regular languages Closed under Intersection

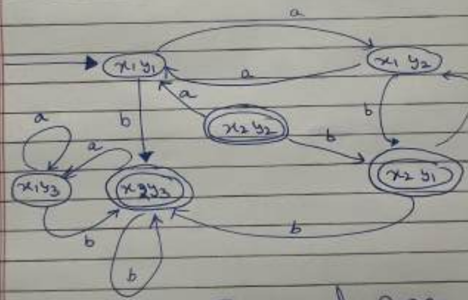
Theorem: The set of regular languages are Closed under Intersection

DFA for M_1 for A_1 DFA M_2 for A_2





we now want a DFA M_3 for $\boxed{A_1 \cap A_2}$



Solution for $A_1 \cap A_2$

$x_2 \xrightarrow{a} x_1y_1$

$y_2 \xrightarrow{b} x_2y_1$

$x_1 \xrightarrow{a} x_1y_2$	$x_1 \xrightarrow{a} x_1y_1$	$x_1 \xrightarrow{a} x_1y_3$	$x_2 \xrightarrow{a} x_1y_2$	$x_2 \xrightarrow{a} x_1y_3$
$y_1 \xrightarrow{b} x_2y_3$	$y_2 \xrightarrow{b} x_2y_1$	$y_3 \xrightarrow{b} x_2y_3$	$y_1 \xrightarrow{b} x_2y_3$	$y_3 \xrightarrow{b} x_2y_3$