

# CS & IT ENGINEERING

## Theory of Computation

Regular Languages

Lecture No.- 14



Mallesham Devasane Sir

# Recap of Previous Lecture



Topic

Regular Expressions





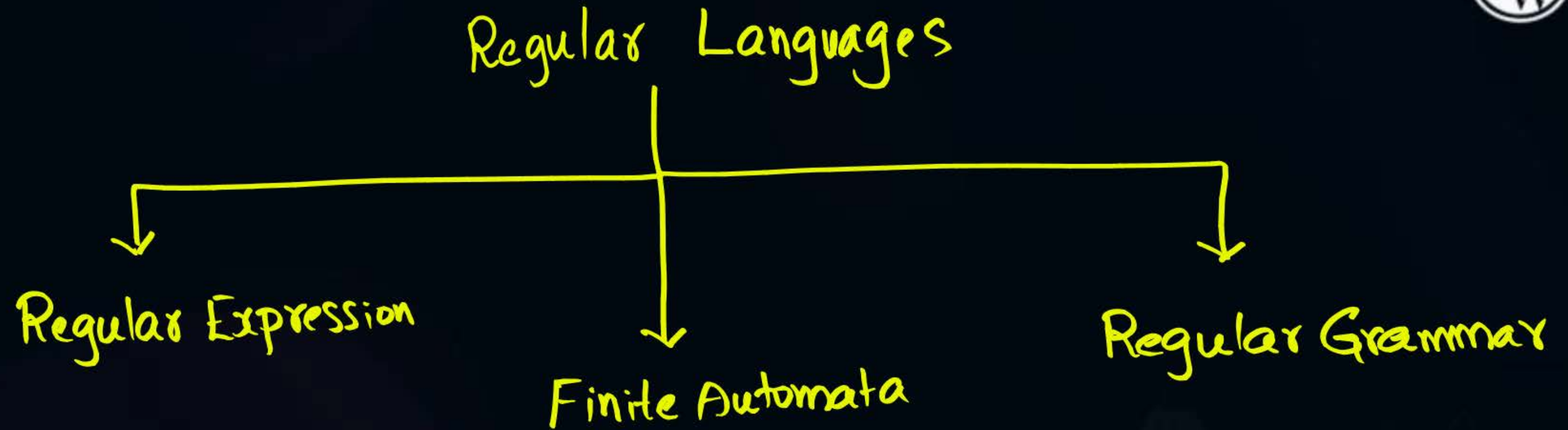
# Topics to be Covered



Topic

Finite Automata





Automata  
↳ plural

Automaton  
↳ singular

## Automata

- It is a machine
- It represents a language

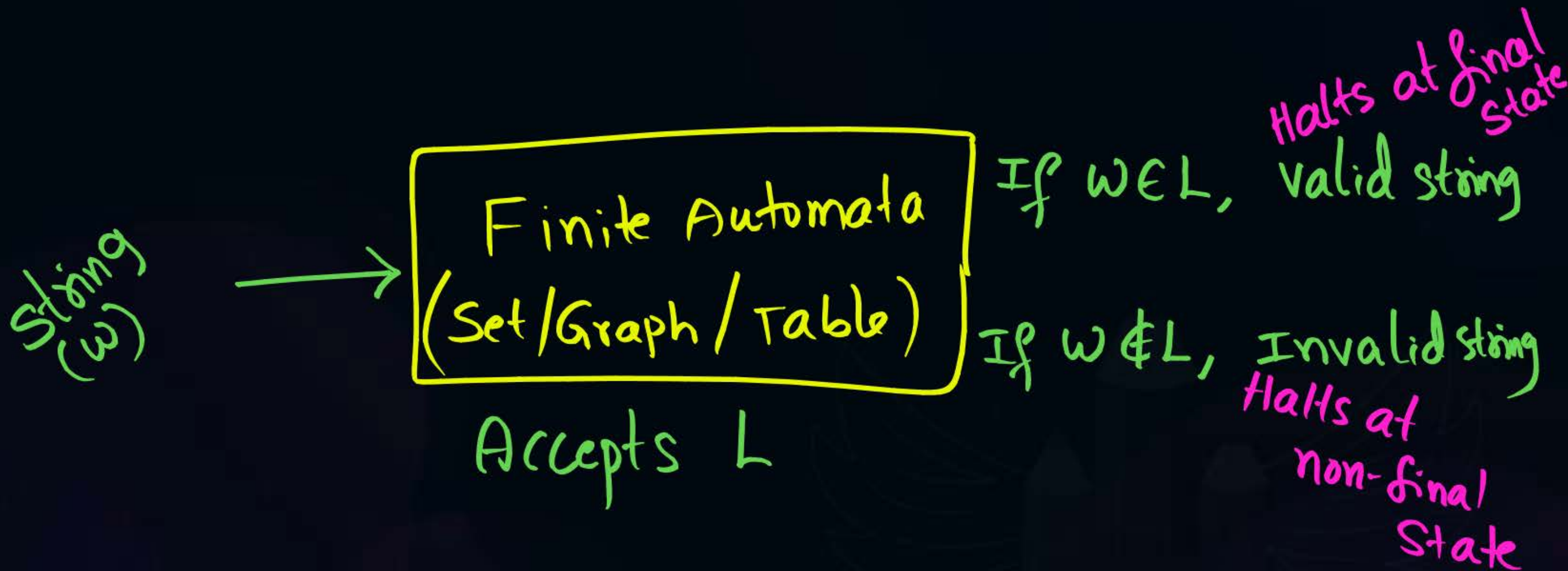


→ It is a machine that represents a regular language  
(accepts)  
(recognizes)

(Finite State Machine) (FSM)  
Finite Automata (FA)







## Configuration :

$$FA = (Q, \Sigma, \delta, q_0, F)$$

→ Set of final states

→ state (starting/initial)

→ Set of transitions [Transition function]  
(function)

→ Set of input symbols

→ Set of states

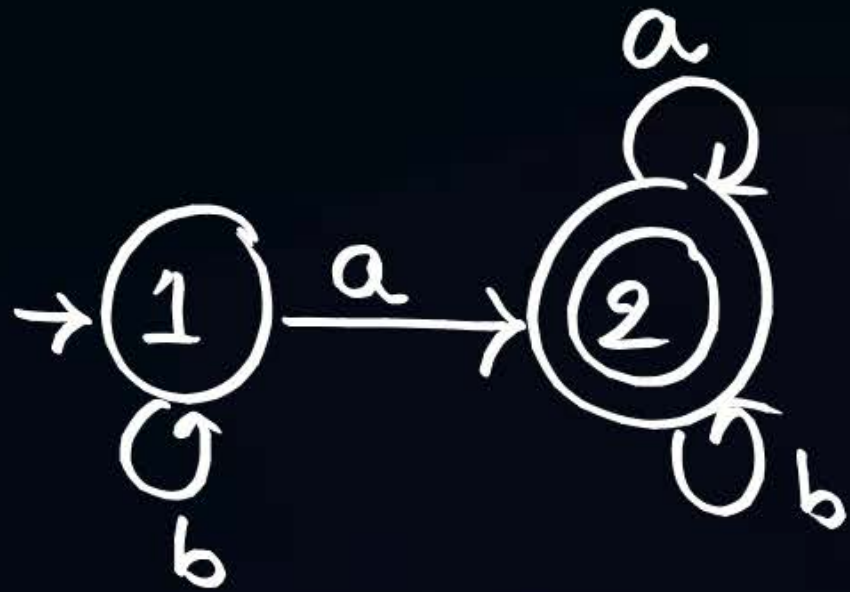
$$\delta_{\text{DFA}} : Q \times \Sigma \rightarrow Q$$

$$\delta_{\text{NFA}} : Q \times \Sigma \cup \{\epsilon\} \rightarrow 2^Q$$



$$FA = (Q, \Sigma, \delta, q_0, F)$$

Graph



→ ○ : Initial state

⊙ : Final state

○ : Non final

Table

$\delta$	a	b
→ 1 Initial	2	1
* 2 final	2	2

⊙

$$\delta: Q \times \Sigma \rightarrow Q$$

Set

$$Q = \{1, 2\}$$

$$q_0 = 1$$

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

$$F = \{2\}$$

$$\delta(1, a) = 2$$

$$\delta(1, b) = 1$$

$$\delta(2, a) = 2$$

$$\delta(2, b) = 2$$

$$\delta(\underline{1}, \underline{a}) = \underline{2}$$

$$\delta(1, b) = 1$$

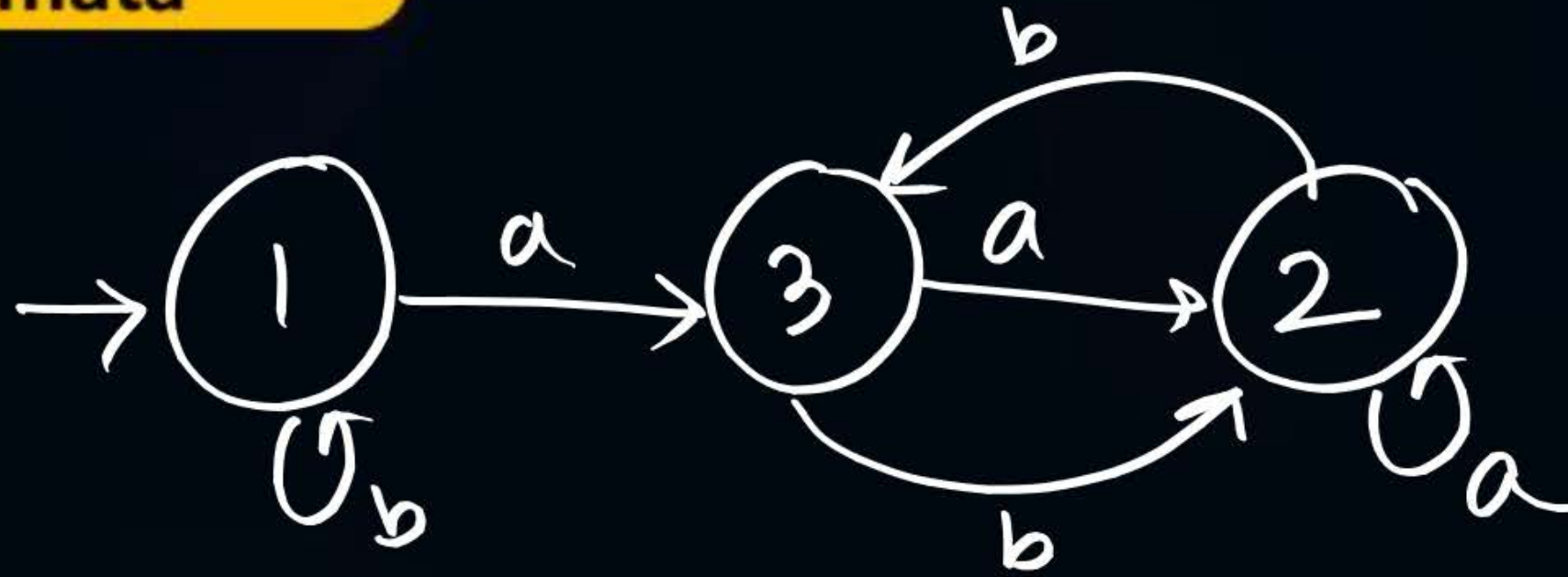
$$\delta(2, a) = 2$$

$$\delta(2, b) = 2$$

$$\delta: Q \times \Sigma \rightarrow Q$$

$$\Rightarrow \delta = \left\{ \left( \underbrace{(1, a)}_{\text{Domain}}, \underbrace{2}_{\text{Co-domain}} \right), \left( (1, b), 1 \right), \left( (2, a), 2 \right), \right. \\ \left. \left( (2, b), 2 \right) \right\}$$





	a	b
1	3	1
2	2	3
3	2	2

$$Q = \{1, 2, 3\}$$

$$F = \{ \}$$

$$= \emptyset$$

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

$$\delta(1, a) = 3$$

$$\delta(1, b) = 1$$

$$\delta(2, a) = 2$$

$$\delta(2, b) = 3$$

$$\delta(3, a) = 2$$

$$\delta(3, b) = 1$$





## 2 mins Summary



Topic

Finite Automata

**THANK - YOU**