

Symbol: a, b, c ..... single symbol

Alphabet  $\Sigma$ : Finite set of symbols  
 $\{0, 1\}$  = Binary Alphabet

String: w, x, y, z ..... represents Strings

- Is a list over alphabet  $\Sigma$  where each element of the string is a member of  $\Sigma$
- $\Sigma^*$  = Set of all string over the alphabet  $\Sigma$
- $\epsilon$  = Special string (Empty String) of length 0

Example:

$$\{0, 1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, \dots\}$$

$$\{a\}^* = \{\epsilon, a, aa, aaa, aaaa, \dots\}$$

$$\{ab\}^* = \{\epsilon, ab, abab, ababab, \dots\}$$

Language L: is a subset of  $\Sigma^*$

Example:

List of string of 0's and 1's with no two consecutive 1's

$$L = \{\epsilon, 0, 1, 00, 01, 10, 000, 001, 010, 100, 101, \dots\}$$

Transition Function  $\delta$ :

$\delta(q_1, a) = \{q_2\}$  means, is in state ' $q_1$ ' and input ' $a$ ' is received and transits to state set  $\{q_2\}$

## Extended Transition Function $\hat{\delta}$ :

- The effect of a string of inputs on an automaton
- Is computed for state 'q' and inputs  $a_1 a_2 a_3 \dots a_n$  by following a path in the transition starting at 'q' and selecting the edge with labels  $a_1 a_2 a_3 \dots a_n$  sequentially.

## Inductive Definition of $\hat{\delta}$ :

- Induction on length of String
- Basis:  $\delta(q, \epsilon) = q$
- Induction  $\delta(q, wa) = \delta(\delta(q, w), a)$

## Example:

		0	1	$\hat{\delta}(B, 011)$	$= \delta(\delta(B, 01), 1)$
$\rightarrow$	*	A	B		$= \delta(\delta(\delta(B, 0), 1), 1)$
	*	B	C		$= \delta(\delta(A, 1), 1)$
		C	C		$= \delta(B, 1)$
					$= C$

## Deterministic Finite Automata (DFA): Unique state for every input symbol

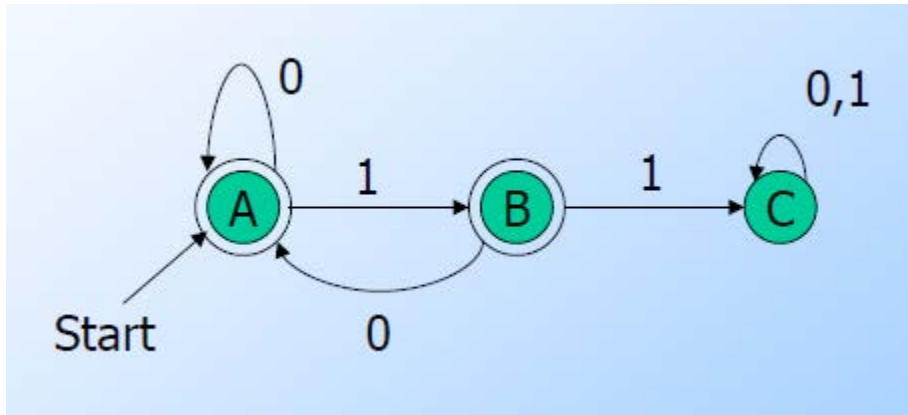
- A finite set of states  $Q$
- An input alphabet  $\Sigma$
- A transition function  $\delta$
- A starting state  $q_0$  in  $Q$
- A set of final/ accepting states ( $F \subseteq Q$ )

Language of DFA:

$L(A) = \text{the set of string 'w' such that } \hat{\delta}(q_0, w) \text{ is in } F$

Example:

$L(A) = \{w \mid w \text{ is in } \{0, 1\}^* \text{ and 'w' does not have two consecutive 1's}\}$



		0	1
→ *	A	A	B
*	B	A	C
	C	C	C