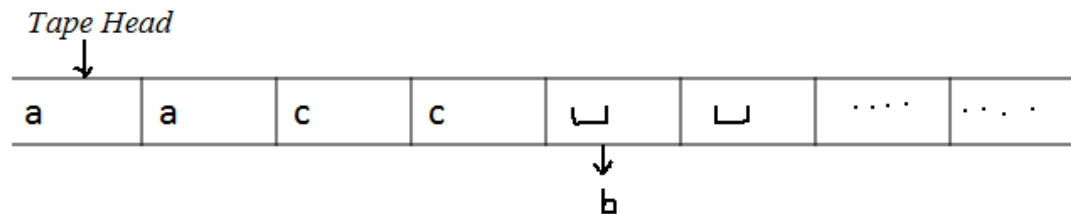




# Turing Machine

- FSA → Regular language
- PDA → CFG → Context Free language → Stack
- Turing Machine → recursively enumerable language
  - Alan Turing (1936)
  - Unrestricted Grammar

– **Tape**

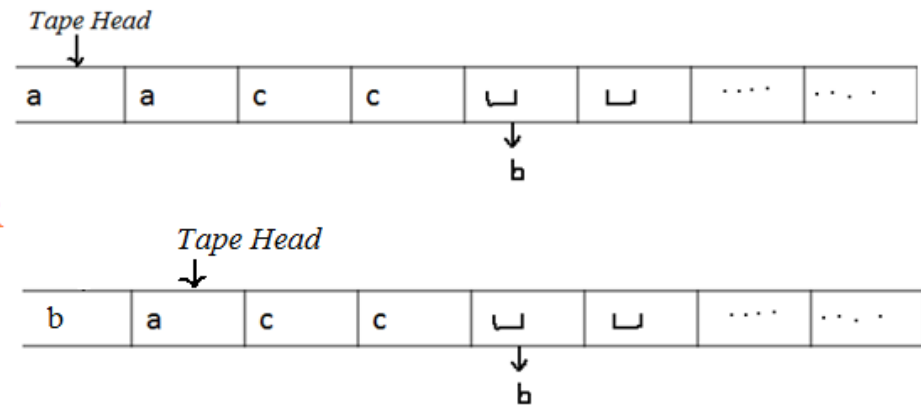
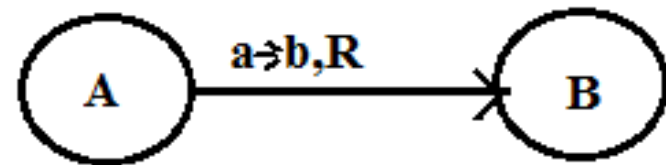


- TM → what can be computed
- **Model for Computer**
- Algorithm → TM can do its computation
- (Simulation)



# Graphical Notation

- $a \rightarrow$  Symbol to read
- $b \rightarrow$  Symbol to Write/ Update
- $R \rightarrow$  Move to Tape Head to right
- (Tape Head Moves  $\rightarrow L/R$ )
- $a \rightarrow b, R$  (Update cell)
- $a \rightarrow a, R$  (No update )





# Formal Definition

A Turing Machine can be defined as a set of 7 tuples

$$(Q, \Sigma, \Gamma, \delta, q_0, b, F)$$

$Q$  → Non empty set of States

$\Sigma$  → Non empty set of Symbols

$\Gamma$  → Non empty set of Tape Symbols

$\delta$  → Transition function defined as

$$Q \times \Sigma \rightarrow \Gamma \times (R/L) \times Q$$

$q_0$  → Initial State

$b$  → Blank Symbol

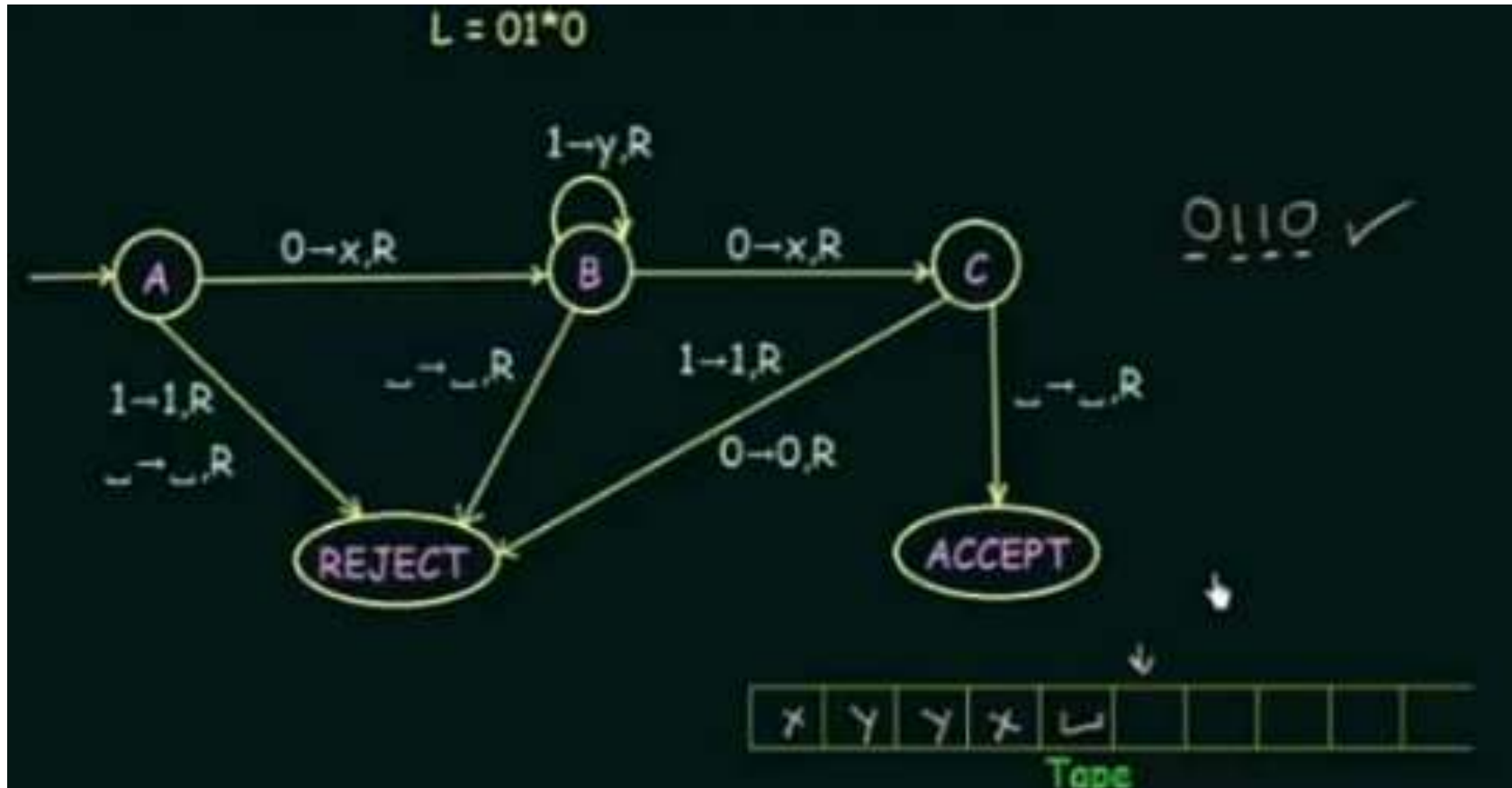
$F$  → Set of Final states (Accept state & Reject State)

Thus, the Production rule of Turing Machine will be written as

$$\delta(q_0, a) \rightarrow (q_L, \gamma, R)$$

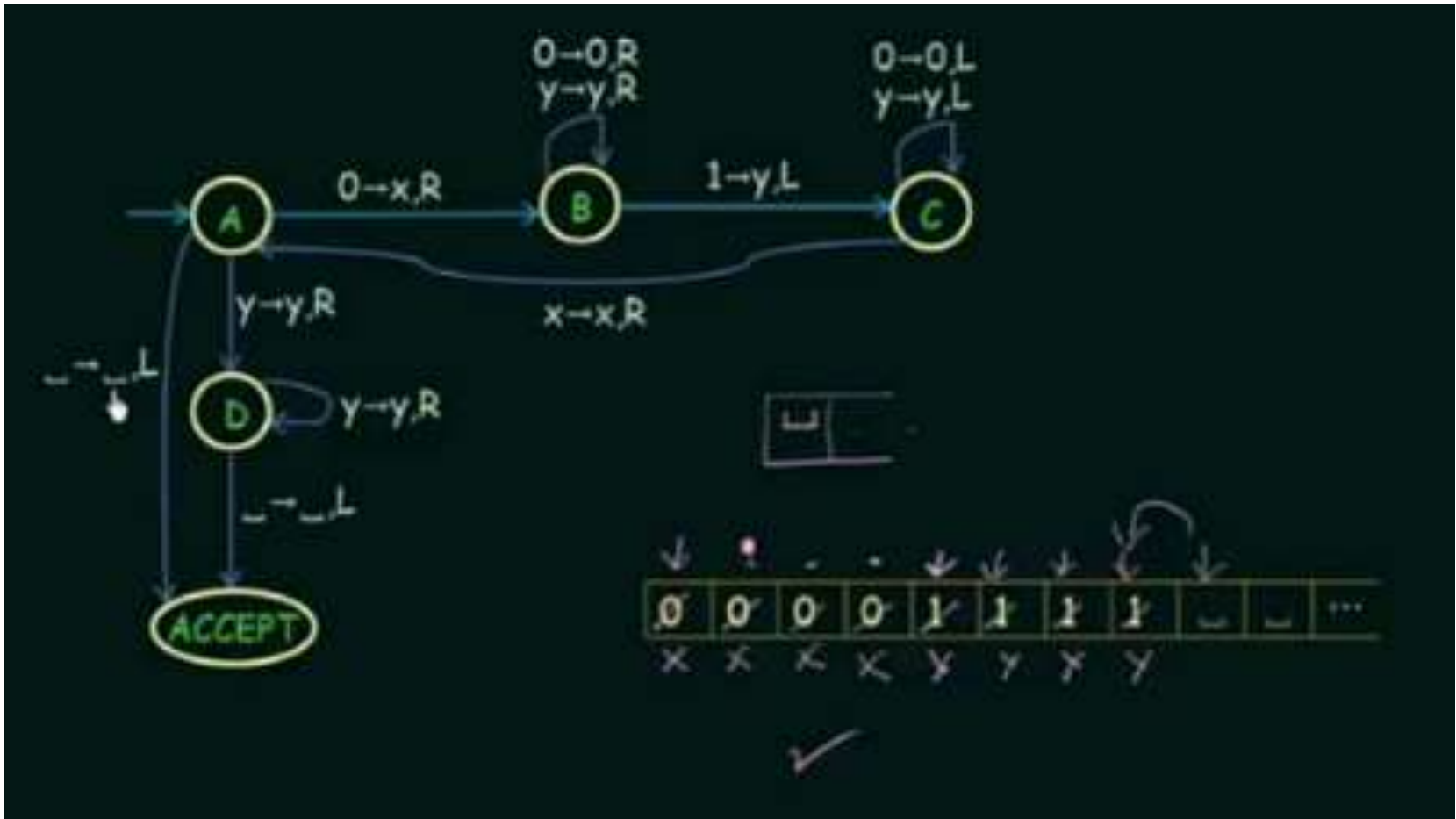


# Turing Machine – Example 1





# Turing Machine – Example2 ( $L=0^n1^n$ )





# Applications & Questions



- Applications
  - Computer Networks
  - Artificial Intelligence
  - Machine Learning
- Turing Machine was invented by
  - Alan Turing          Turing taring          Nickel Turing
- In one move the turing machine
  - May change the state
  - Move one tape position by Left or Right
  - Write the Symbol on the cell
  - All these
- Turing machine is more powerful than
  - PDA          -FSA          -Both these