

# Class 27: Universal Turing Machines



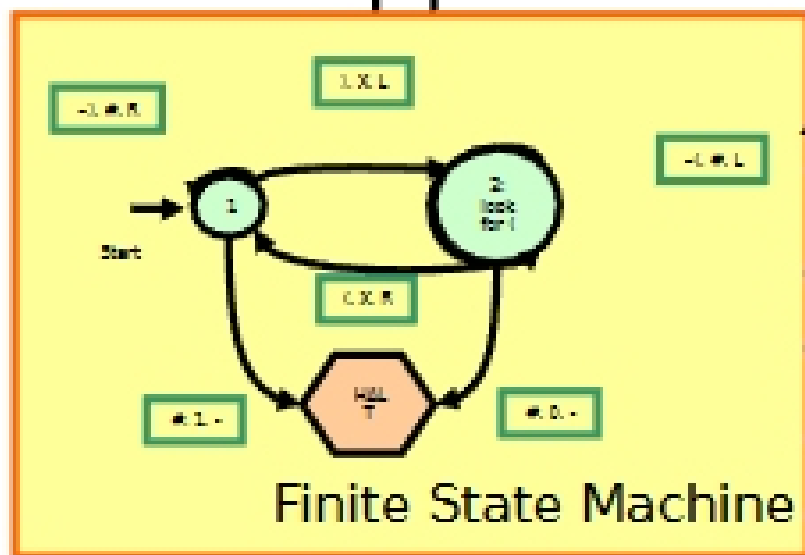
CS150: Computer Science  
University of Virginia  
Computer Science



**David Evans**

<http://www.cs.virginia.edu/evans>

# Turing Machine (1936)



$TuringMachine ::= \langle Alphabet, Tape, FSM \rangle$

$Alphabet ::= \{ Symbol^* \}$

$Tape ::= \langle LeftSide, Current, RightSide \rangle$

$OneSquare ::= Symbol \mid \#$

$Current ::= OneSquare$

$LeftSide ::= [ Square^* ]$

$RightSide ::= [ Square^* ]$

Everything to left of *LeftSide* is #.

Everything to right of *RightSide* is #.

# Describing Finite State Machines

*TuringMachine ::= < Alphabet, Tape, **FSM** >*

*FSM ::= < States, TransitionRules, InitialState, HaltingStates >*

*States ::= { StateName\* }*

*InitialState ::= StateName*      must be element of *States*

*HaltingStates ::= { StateName\* }*      all must be elements of *States*

*TransitionRules ::= { TransitionRule\* }*

*TransitionRule ::=*

*< StateName, ;; Current State*

*Transition Rule is a procedure:*

*OneSquare, ;; Current square*

*Inputs: StateName,*

*StateName, ;; Next State*

*OneSquare*

*OneSquare, ;; Write on tape*

*Outputs: StateName,*

*Direction > ;; Move tape*

*OneSquare,*

*Direction*

*Direction ::= L, R, #*