TURING MACHINE

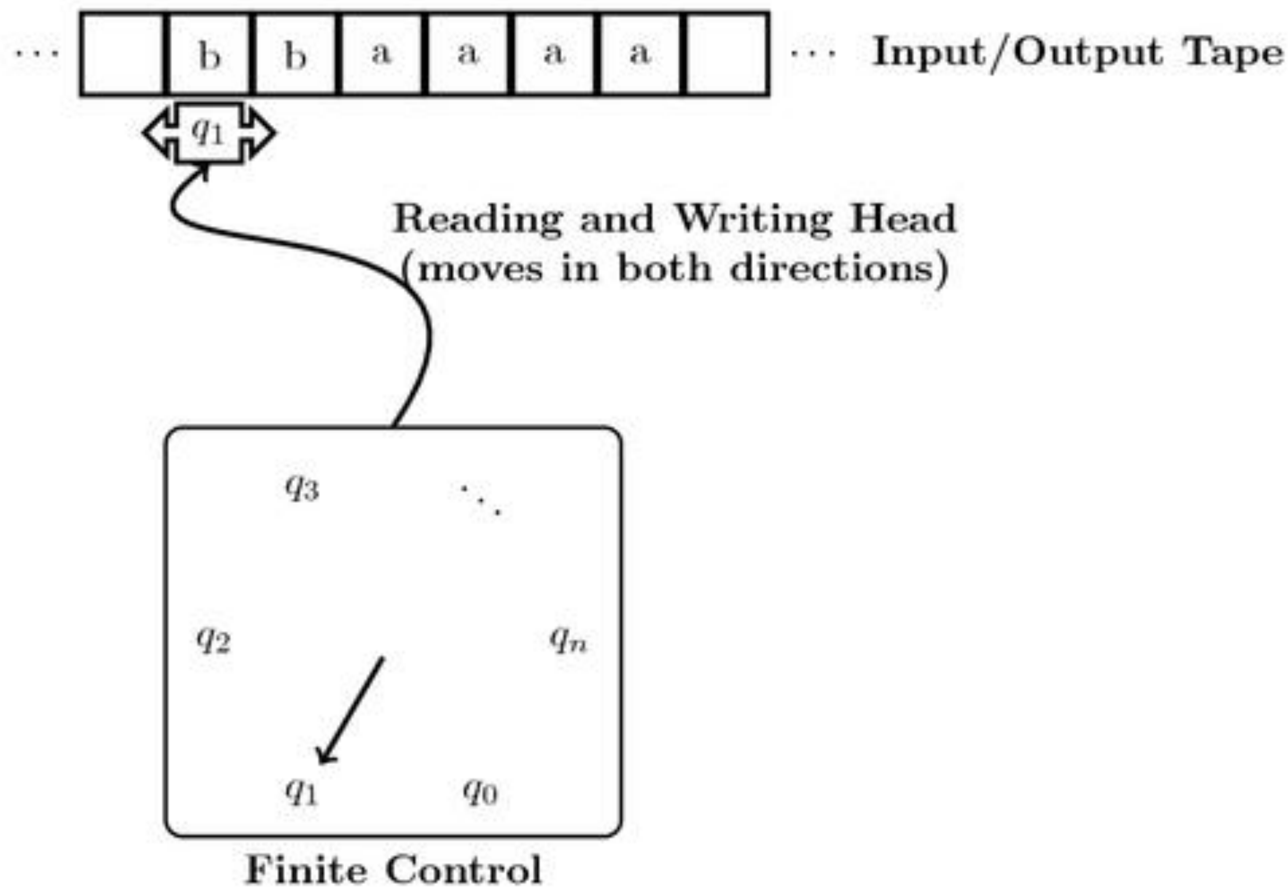
Beulah A.
AP/CSE

Devices of Increasing Computational Power

- So far:
  - Finite Automata – good for devices with small amounts of memory, relatively simple control
  - Pushdown Automata – stack-based automata
- But both have limitations for even simple tasks, too restrictive as general purpose computers
- Enter the **Turing Machine**
  - More powerful than either of the above
  - Essentially a finite automaton but with unlimited memory
  - Although theoretical, can do everything a general purpose computer of today can do
    - If a TM can't solve it, neither can a computer (Undecidable problems)

Turing Machine

- A TM consists of a finite control (i.e. a finite state automaton) that is connected to an infinite tape.



Notion for the Turing Machine

- A move of Turing machine includes:
  - Change state;
  - Write a tape symbol in the cell scanned;
  - Move the tape head left or right.

Formal Definition

- A Turing machine (TM) is a 7-tuple
  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ where
  - Q – A finite set of states of the finite control
  - Σ – A finite set of input symbols
  - Γ – A set of tape symbols, with Σ being a subset
  - Q_0 - The start state, in Q
  - B - The blank symbol in Γ, *not* in Σ (should not be an input symbol)
  - F - The set of final or accepting states

Formal Definition

- δ: a transition function $Q \times \Gamma \rightarrow Q \times \Gamma \times \{L,R\}$
- Example $\delta(q, X) = (p, Y, D)$
  - q - The current state, in Q
  - X - A tape symbol being scanned
  - p - The next state, in Q
  - Y - The tape symbol written on the cell being scanned, used to replace X
  - D - Either L (left) or R (right) telling the move direction of the tape head

Representation of TM

- Turing Machines are represented in 3 ways
  - Instantaneous Descriptions
  - Transition Table
  - Transition Diagram

Instantaneous Descriptions

- The *instantaneous description* (ID) of a TM is represented by
- $X_1 X_2 \ldots X_{i-1}\, q\, X_i X_{i+1} \ldots X_n$ in which
  - q is the current state
  - The tape head is scanning the *i*th symbol from the left
  - $X_1 X_2 \ldots X_n$ is the portion of the tape between the leftmost and the rightmost nonblank symbols

Instantaneous Descriptions

- Moves of a TM M denoted by $\vdash_M$ or $\vdash$ as follows:

  If $\delta(q, X_i) = (p, Y, L)$

$$X_1X_2\ldots X_{i-1}qX_iX_{i+1}\ldots X_n \vdash$$
$$X_1X_2\ldots X_{i-2}pX_{i-1}Y\,X_{i+1}\ldots X_n$$

- Right moves are defined similarly.

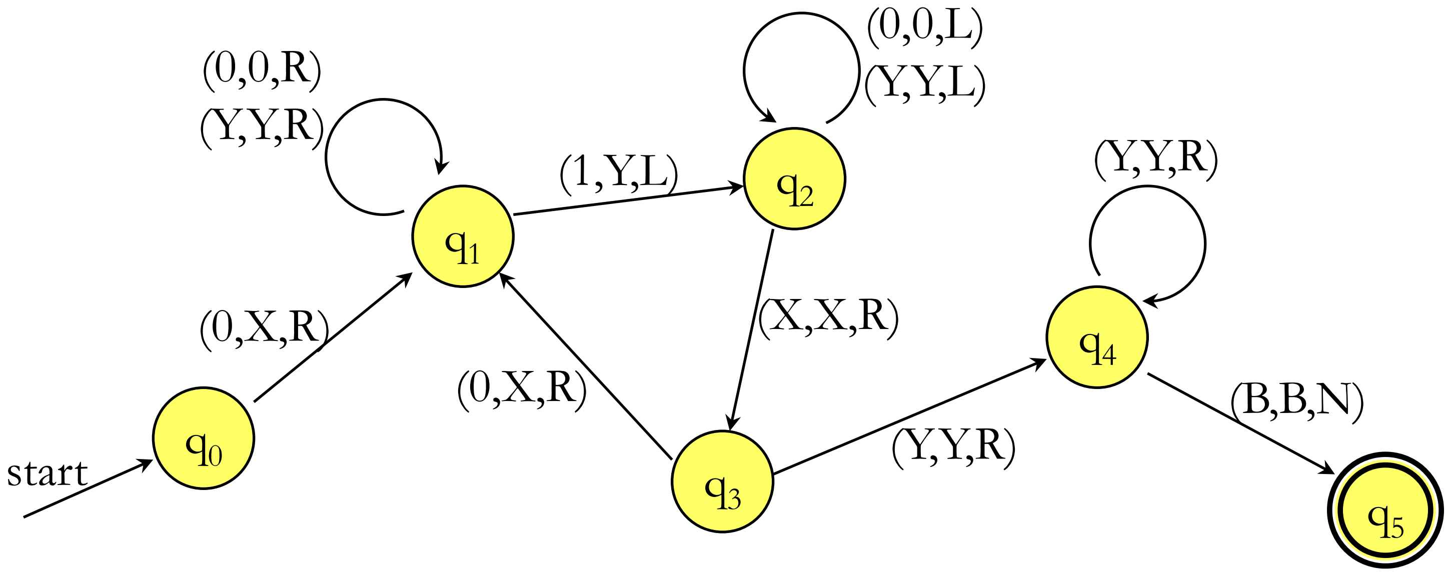
Transition Table

- $L = \{0^n 1^n \mid n \geq 1\}$

	0	1	X	Y	B
q_0	(q_1, X, R)	-	-	-	-
q_1	$(q_1, 0, R)$	(q_2, Y, L)	-	(q_1, Y, R)	-
q_2	$(q_2, 0, L)$	-	(q_3, X, R)	(q_2, Y, L)	-
q_3	(q_1, X, R)	-	-	(q_4, Y, R)	-
q_4	-	-	-	(q_4, Y, R)	(q_5, B, N)
q_5	-	-	-	-	-

- : undefined and the machine halts.

Transition Diagram



Language Acceptance of TM

- Let $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ be a TM.
- The language accepted by M is

$$L(M) = \{w \mid w \in \Sigma^* \text{ and } q_0 w \mid\!- \alpha p \beta \text{ with } p \in F, \alpha, \beta \in \Gamma^* \}$$

- Turing machine can accept the string by entering accepting state
- TM can reject the string by entering non-accepting state.
- TM can enter an infinite loop so that it never halts.

Designing a TM

- The fundamental objective in scanning a symbol by R/W head is to 'know' what to do in the future.
- The machine must remember the past symbols scanned.
- Change the states only when there is a change in the written symbol or when there is a change in the movement of R/W head.

Subtraction m - n

For example, proper subtraction m – n is defined to be

m – n for m >= n, and

zero for m < n.

The TM M = ({q0,q1,...,q6}, {0,1}, {0,1,B}, ∂, q0, B, {})

The function ∂ is described below.

$\partial(q0,0) = (q1,B,R)$ Begin. Replace the leading 0 by B.

$\partial(q1,0) = (q1,0,R)$ Search right looking for the first 1.
$\partial(q1,1) = (q2,1,R)$

$\partial(q2,1) = (q2,1,R)$ Search right past 1's until encountering a 0. Change that 0 to 1.
$\partial(q2,0) = (q3,1,L)$

$\partial(q3,0) = (q3,0,L)$ Move left to a blank. Enter state q0 to repeat the cycle.
$\partial(q3,1) = (q3,1,L)$
$\partial(q3,B) = (q0,B,R)$

If in state q2 a B is encountered before a 0, we have situation i described above. Enter state q4 and move left, changing all 1's to B's until encountering a B. This B is changed back to a 0, state q6 is entered and M halts.

$\partial(q2,B) = (q4,B,L)$
$\partial(q4,1) = (q4,B,L)$
$\partial(q4,0) = (q4,0,L)$
$\partial(q4,B) = (q6,0,R)$

If in state q0 a 1 is encountered instead of a 0, the first block of 0's has been exhausted, as in situation (ii) above. M enters state q5 to erase the rest of the tape, then enters q6 and halts.

$\partial(q0,1) = (q5,B,R)$
$\partial(q5,0) = (q5,B,R)$
$\partial(q5,1) = (q5,B,R)$
$\partial(q5,B) = (q6,B,R)$

Subtraction m - n

	symbol		
state	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	-
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	-
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	-	-	-