Lecture 9

Turing Machines

COSE215: Theory of Computation

Seunghoon Woo

Fall 2023

Contents

- **Turing machines as transducers (변환기)**
- **TMs for complicated tasks**

Turing machine

- **Practice:** Design a TM for $L = \{a^n b^n c^n : n \geq 0\}$

Turing machine

- **Practice:** Design a TM for $L = \{a^n b^n c^n : n \geq 0\}$
  - Hint: basic idea
    - ❖ **While** there are a's **do**
    - ❖ **Find** and **Replace** a with **A**
    - ❖ **Find** and **Replace** b with **B**
    - ❖ **Find** and **Replace** c with **C**
    - ❖ **end**
    - ❖ If no symbol a, b, c exist => accept

Turing machine

- **Turing machines as transducers (변환기)**
  - Previous automata => language accepters
  - TM can be used a transducer
- **Definition**
  - A function f with domain D is said to be **Turing-computable** if there exists some Turing machine $M = (Q, \Sigma, \Gamma, \delta, q_0, \square, F)$ such that

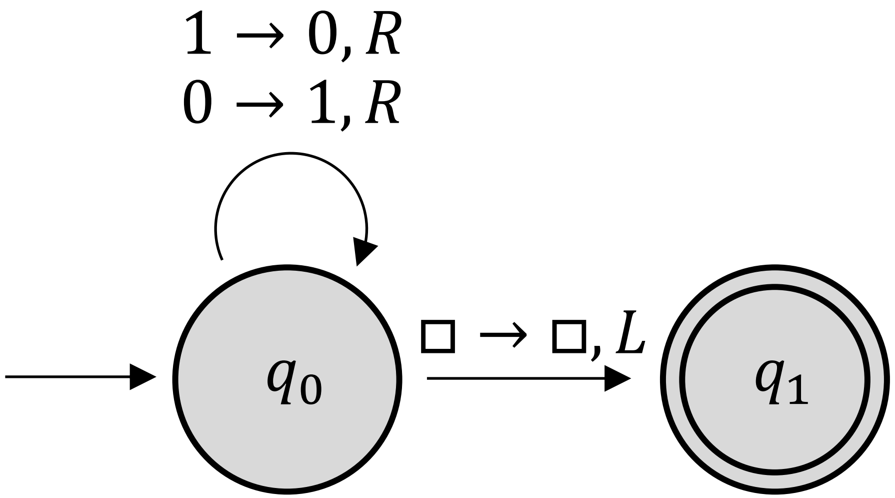
$$q_0 w \vdash^* q_f f(w), \text{ where } q_f \in F \text{ and all } w \in D$$

Turing machine

- **Turing machines as transducers: example**
  - $f(w)$ = (the flip of each bit in w)
    - E.g., 0110 => 1001

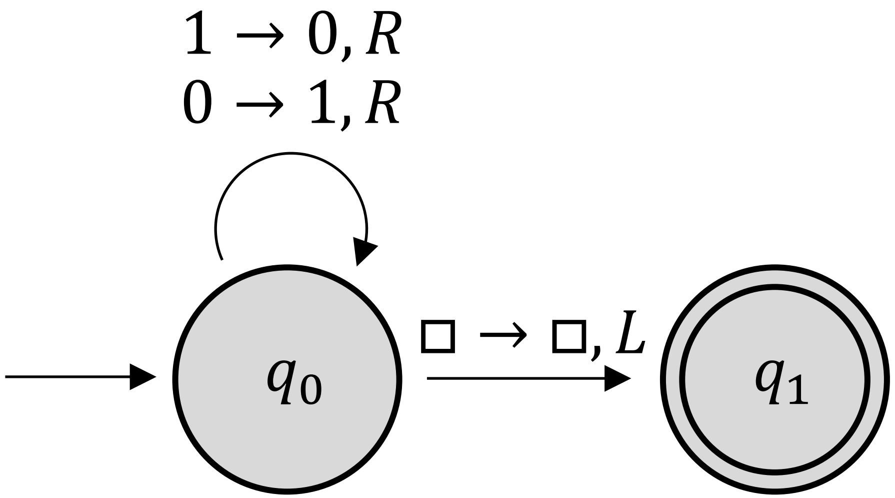
Turing machine

- **Turing machines as transducers: example**
  - $f(w)$ = (the flip of each bit in w)
    - E.g., 0110 => 1001



Turing machine

- **Turing machines as transducers: example**
  - $f(w)$ = (the flip of each bit in w)
    - E.g., 0110 => 1001



ID for the input string 0110

$$q_0 0110$$
$$\vdash 1 q_0 110$$
$$\vdash 10 q_0 10$$
$$\vdash 100 q_0 0$$
$$\vdash 1001 q_0 \square$$
$$\vdash 100 q_1 1$$

Turing machine

- **Turing machines as transducers: example**
  - Given two positive integers x and y, design a TM that computes $x + y$

Turing machine

- **Turing machines as transducers: example**
  - Given two positive integers x and y, design a TM that computes $x + y$
  - Basic idea
    - Choose a convention for representing positive integers
      - Unary notation: any positive integer x is represented by $w(x) \in \{1\}^+$, such that $|w(x)| = x$
        - e.g., $1111 = 4$

Turing machine

- **Turing machines as transducers: example**
  - Given two positive integers x and y, design a TM that computes $x + y$
  - Basic idea
    - Choose a convention for representing positive integers
      - Unary notation: any positive integer x is represented by $w(x) \in \{1\}^+$, such that $|w(x)| = x$
        - e.g., $1111 = 4$
    - Decide how x and y are placed on the tape initially and how their sum is reported
      - We assume that $w(x)$ and $w(y)$ are on the tape separated by a single 0
      - After the computation, $w(x + y)$ will be on the tape followed by a single 0 (head => leftmost)
      - i.e., $q_0 w(x) 0 w(y) \vdash^* q_f w(x + y) 0$

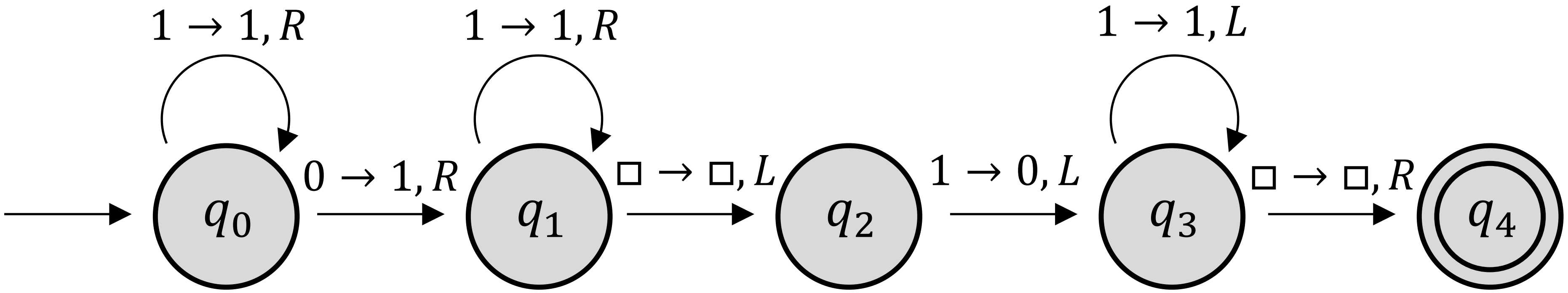
Turing machine

- **Turing machines as transducers: example**
  - Given two positive integers x and y, design a TM that computes $x + y$
  - $M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \{0,1,\square\}, \delta, q_0, \square, \{q_4\})$

The problem of sending the 0 between w(x) and w(y) to the end!

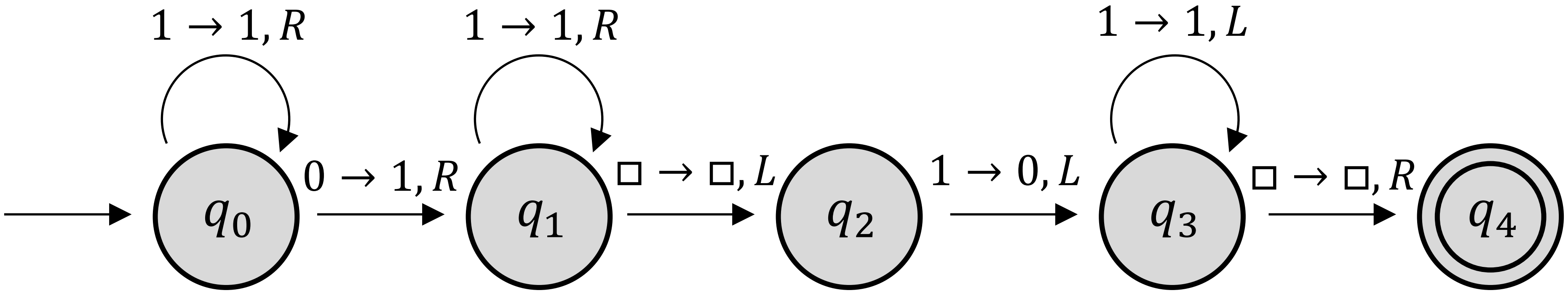
Turing machine

- **Turing machines as transducers: example**
  - Given two positive integers x and y, design a TM that computes $x + y$
  - $M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \{0,1, \square\}, \delta, q_0, \square, \{q_4\})$



Turing machine

- **Turing machines as transducers: example**
  - Given two positive integers x and y, design a TM that computes $x + y$
  - $M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \{0,1, \square\}, \delta, q_0, \square, \{q_4\})$



ID for the input string $3 + 2$

$$
\begin{aligned}
& q_0 111011 \\
\vdash\ & 1 q_0 11011 \\
\vdash\ & 11 q_0 1011 \\
\vdash\ & 111 q_0 011 \\
\vdash\ & 1111 q_1 11 \\
\vdash\ & 11111 q_1 1 \\
\vdash\ & 111111 q_1 \square \\
\vdash\ & 11111 q_2 1 \\
\vdash\ & 1111 q_3 10 \\
\vdash^*\ & q_3 \square 111110 \\
\vdash\ & q_4 111110
\end{aligned}
$$

Turing machine

- **Turing machines as transducers: practice**
  - Design a Turing machine that copies strings of 1's

$$q_0 w \vdash^* q_f ww$$

Turing machine

- **Turing machines as transducers: practice**
  - Design a Turing machine that copies strings of 1's

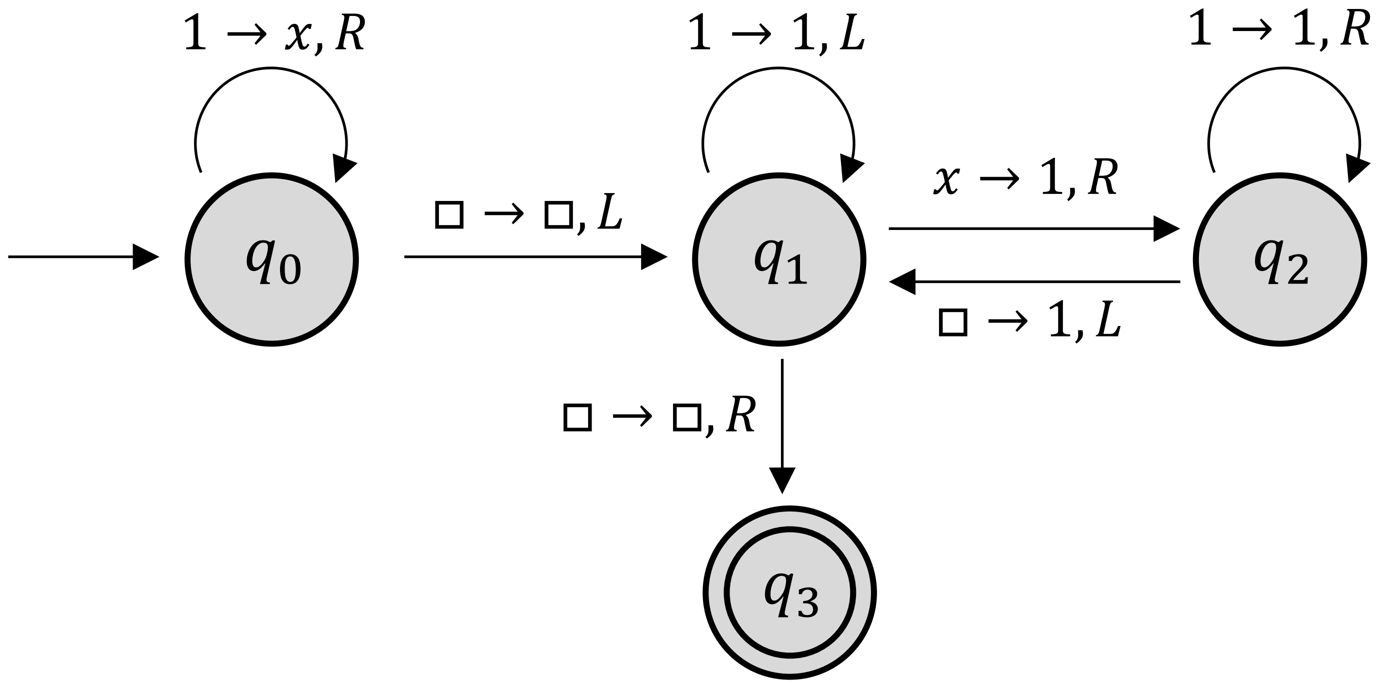
$$q_0 w \vdash^* q_f ww$$

  - Hint: basic idea
    1. Replace every 1 by a symbol (let x)
    2. Find the rightmost x and replace it with 1
    3. Travel to the right end of the nonblack region and create another 1 there (for copying)
    4. Repeat Steps 2 and 3 until there are no more x's

Turing machine

- **Turing machines as transducers: practice**
  - Design a Turing machine that copies strings of 1's

$$q_0 w \vdash^* q_f ww$$



Combining TMs for complicated tasks

- **High-level description of Turing machines**
  - Block diagrams
    - Encapsulate computation in boxes, details are not shown
  - Pseudocode
    - Outline a computation with descriptive phrases

Combining TMs for complicated tasks

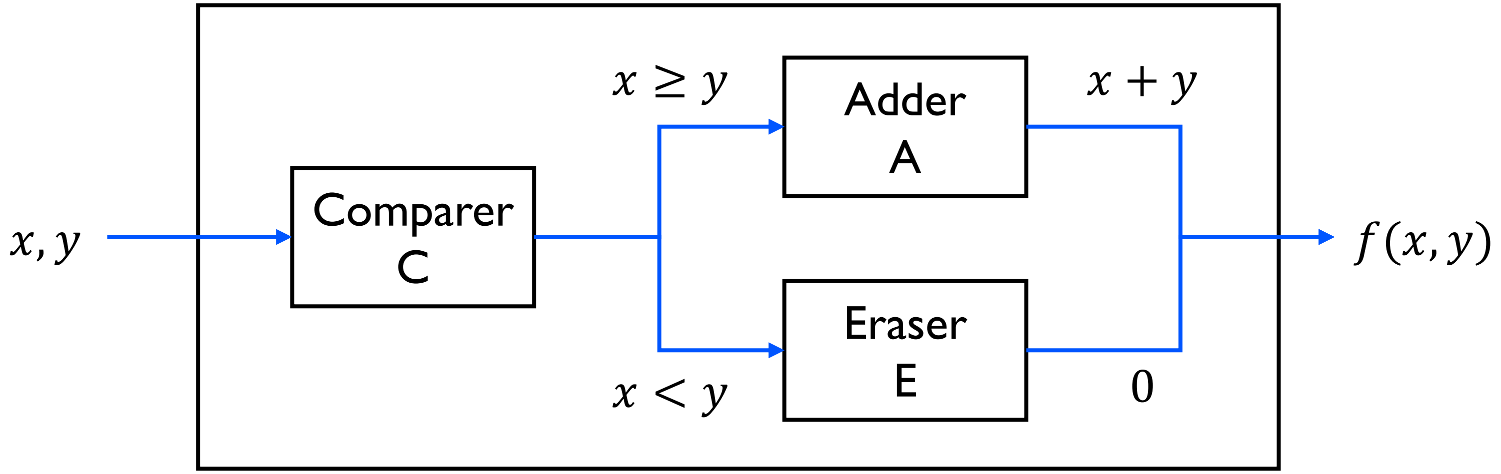
- **High-level description of Turing machines: Block diagrams**
  - Design a Turing machine that computes the function

$$f(x, y) = \begin{cases} x + y & if\ x \geq y \\ 0 & if\ x < y \end{cases}$$

Combining TMs for complicated tasks

- **High-level description of Turing machines: Block diagrams**
  - Design a Turing machine that computes the function

$$f(x,y) = \begin{cases} x + y & if\ x \geq y \\ 0 & if\ x < y \end{cases}$$

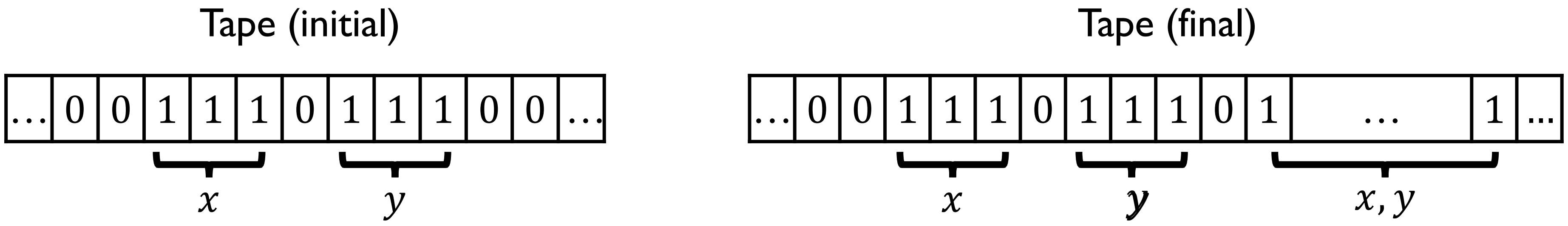


Combining TMs for complicated tasks

- **High-level description of Turing machines: Pseudocode**
  - Design a TM that multiplies two positive integers (in unary notation)

Combining TMs for complicated tasks

- **High-level description of Turing machines: Pseudocode**
  - Design a TM that multiplies two positive integers (in unary notation)
  - Let us assume that the initial and final tape contents are as follows.



Combining TMs for complicated tasks

- **High-level description of Turing machines: Pseudocode**
  - Design a TM that multiplies two positive integers (in unary notation)
  - Let us assume that the initial and final tape contents are as follows.

Tape (initial)

$$\boxed{\ldots} \boxed{0} \boxed{0} \underbrace{\boxed{1} \boxed{1} \boxed{1}}_{x} \boxed{0} \underbrace{\boxed{1} \boxed{1} \boxed{1}}_{y} \boxed{0} \boxed{0} \boxed{\ldots}$$

Tape (final)

$$\boxed{\ldots} \boxed{0} \boxed{0} \underbrace{\boxed{1} \boxed{1} \boxed{1}}_{x} \boxed{0} \underbrace{\boxed{1} \boxed{1} \boxed{1}}_{y} \boxed{0} \underbrace{\boxed{1} \boxed{\quad \ldots \quad} \boxed{1}}_{x,y} \boxed{\ldots}$$

1. Repeat the following steps until x contains no more 1's
   Find a 1 in x and replace it with another symbol (let a)
   Replace the rightmost 0 by $y0$
2. Replace all a's with 1's

Turing Thesis

- **Any computation being carried out by mechanical means can be performed by some Turing machine**
  - Anything that can be done on any existing digital computer can also be done by a Turing machine
  - No one has yet been able to suggest a problem, solvable by what we intuitively consider an algorithm, for which a Turing machine program cannot be written
  - Alternative models have been proposed for mechanical computation, but non of them is more powerful than the Turing machine model

Next Lecture

- **Other models of Turing machines**