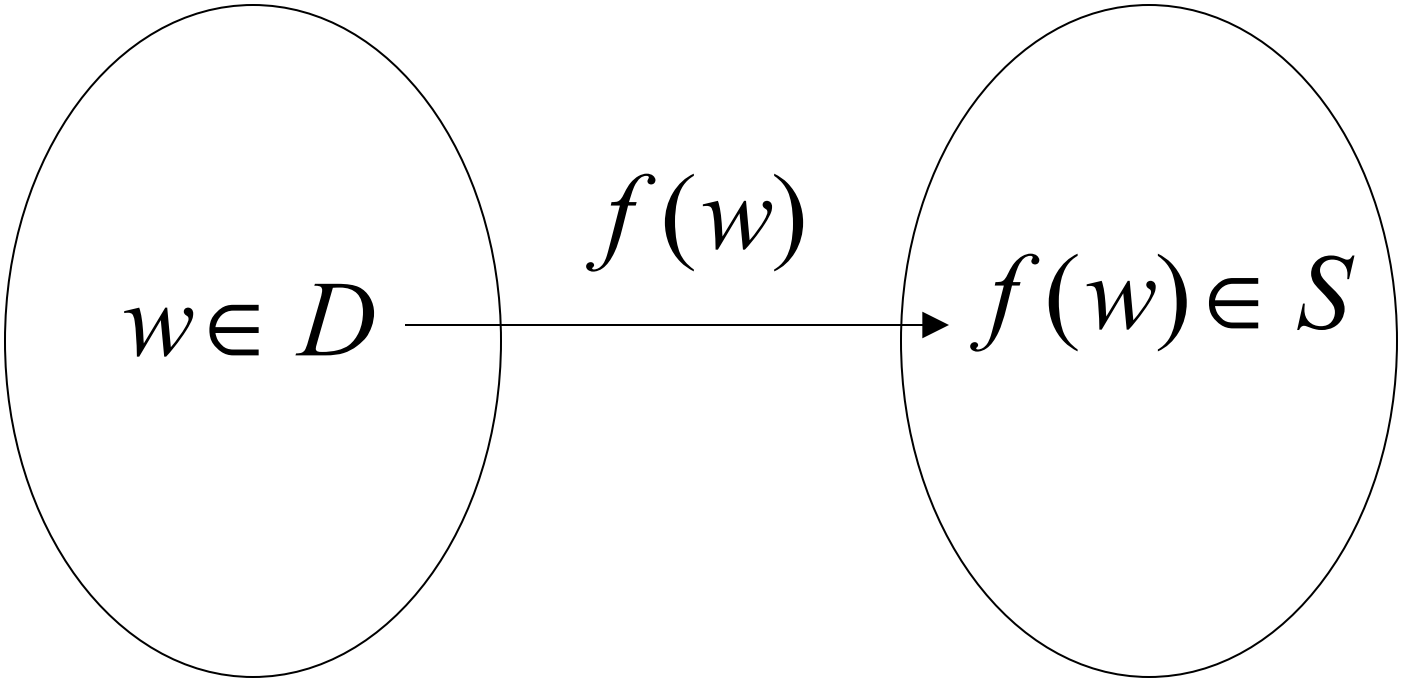
# Computing Functions with Turing Machines

A function $f(w)$ has:

Domain: $D$

Result Region: $S$



A function may have many parameters:

Example: Addition function

$$f(x, y) = x + y$$

# Integer Domain

Decimal: 5

Binary: 101

Unary: 11111

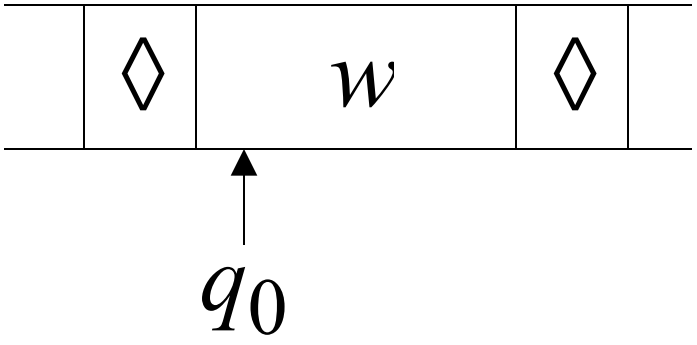
We prefer **unary** representation:

easier to manipulate with Turing machines

# Definition:

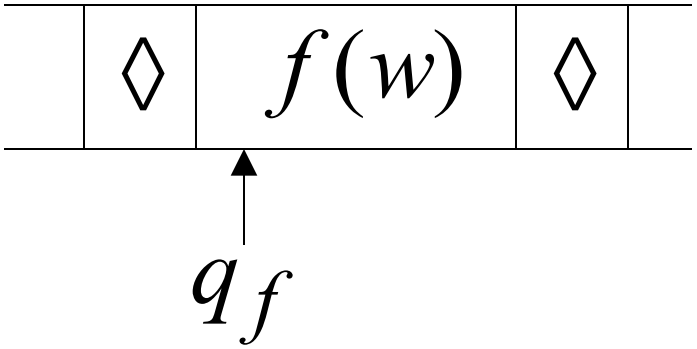
A function $f$ is computable if there is a Turing Machine $M$ such that:

Initial configuration




initial state

Final configuration




accept state

For all $w \in D$ Domain

In other words:

A function $f$ is computable if there is a Turing Machine $M$ such that:

$$q_0\, w \overset{*}{\succ} q_f\, f(w)$$

Initial Configuration

Final Configuration

For all $w \in D$ Domain

# Example

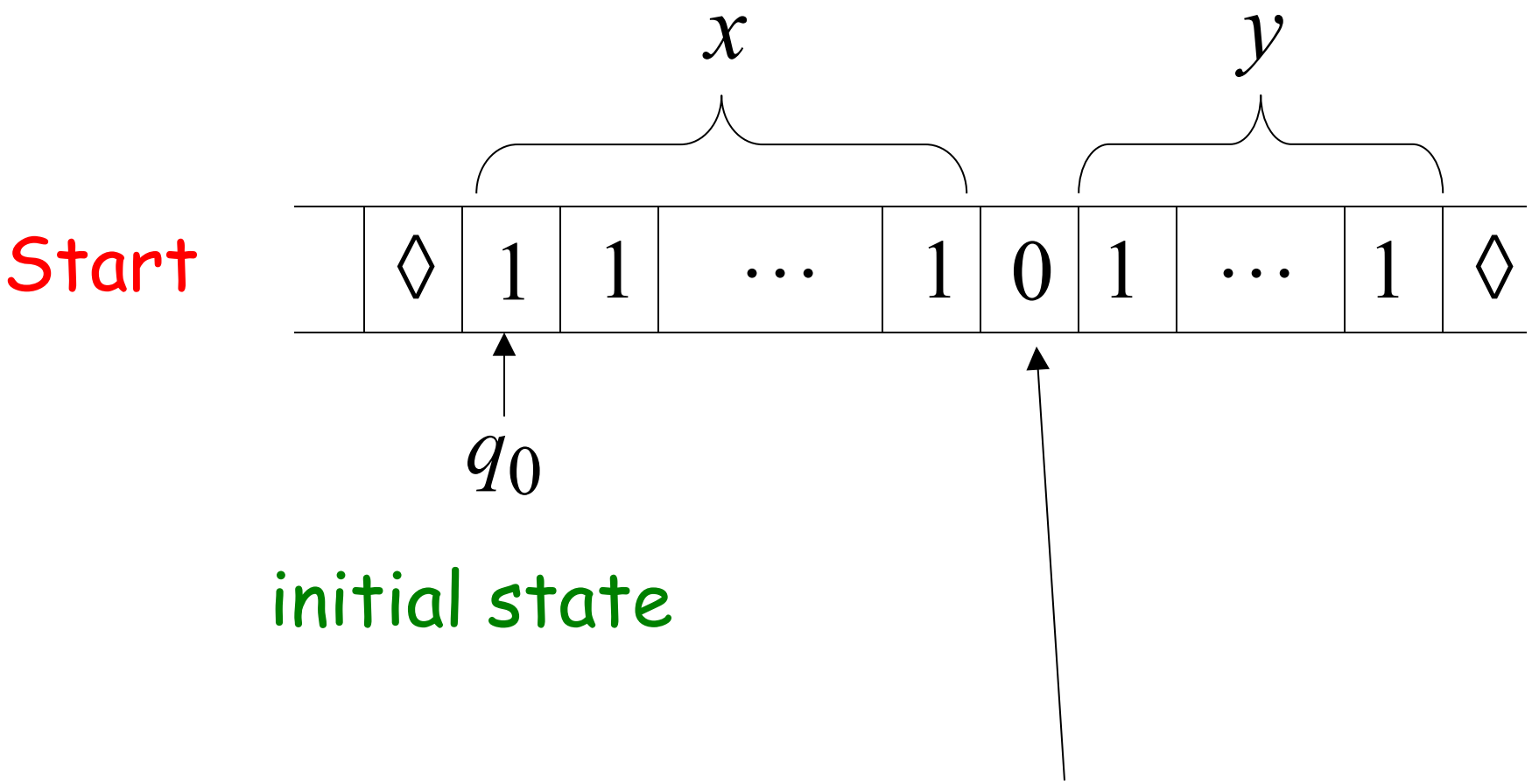
The function $f(x,y) = x + y$ is computable

$x, y$ are integers

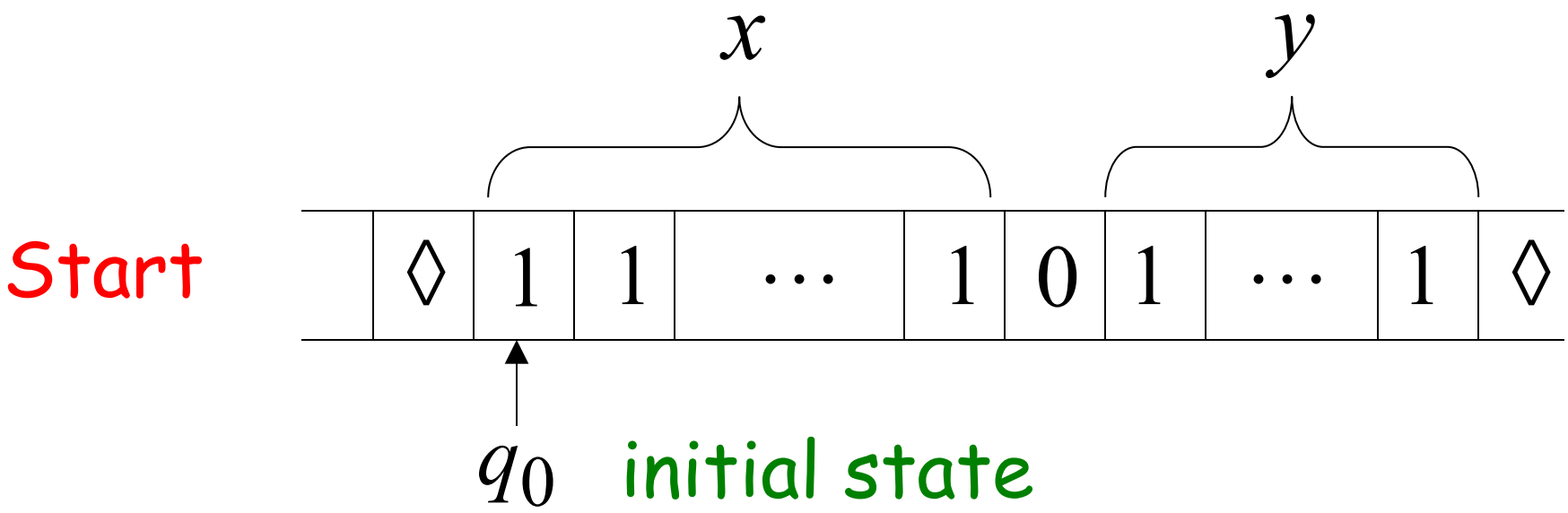
Turing Machine:

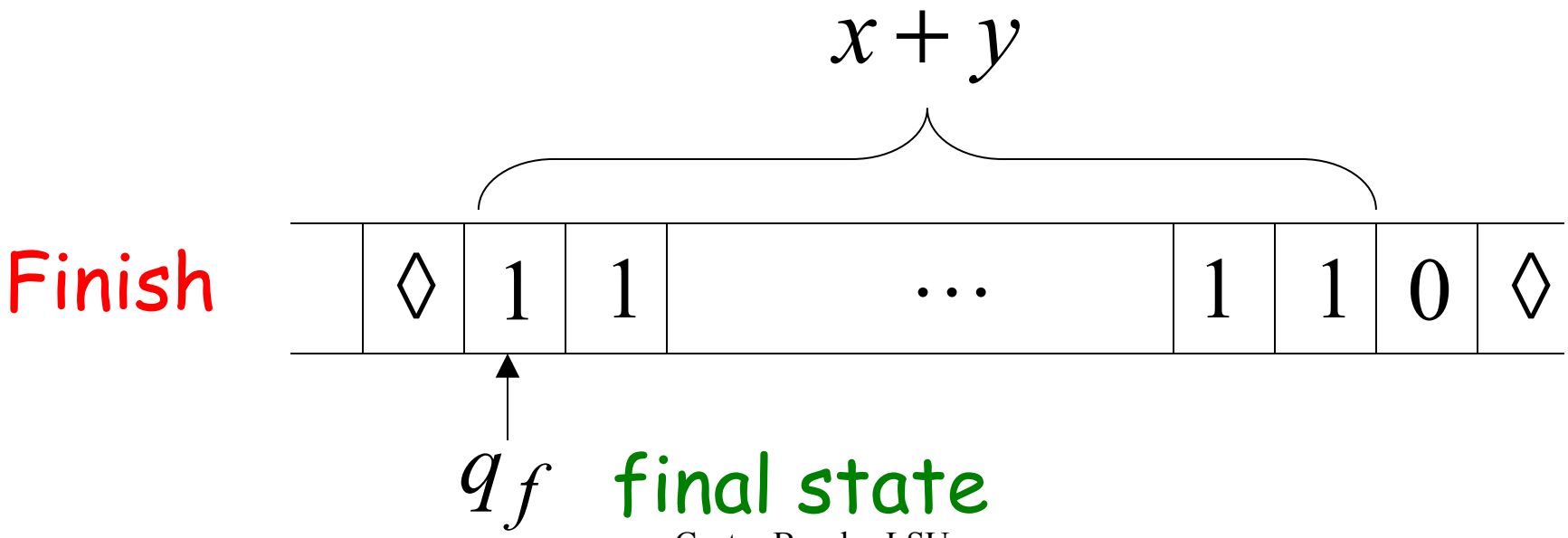
Input string: $x0y$ unary

Output string: $xy0$ unary

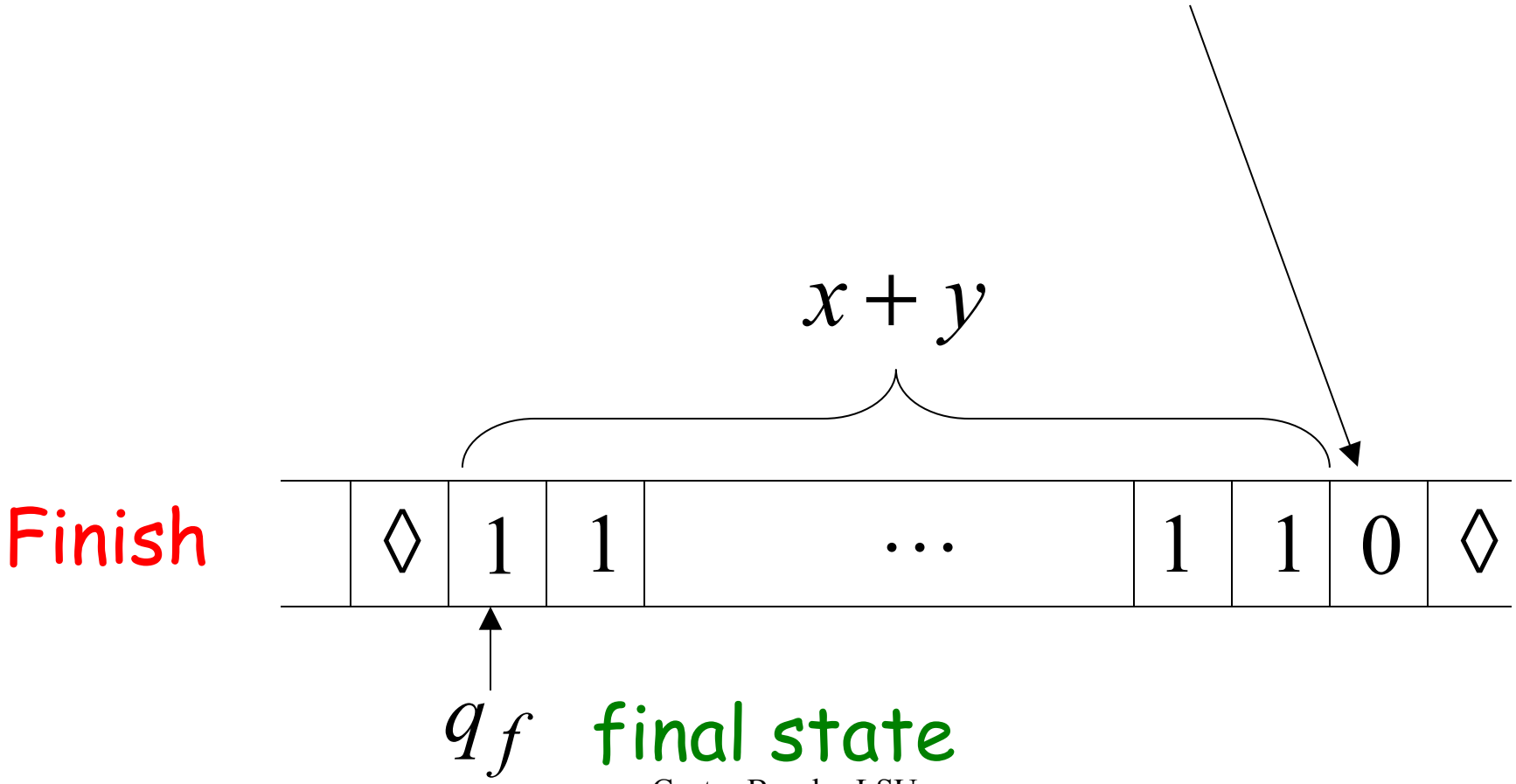



The 0 is the delimiter that separates the two numbers

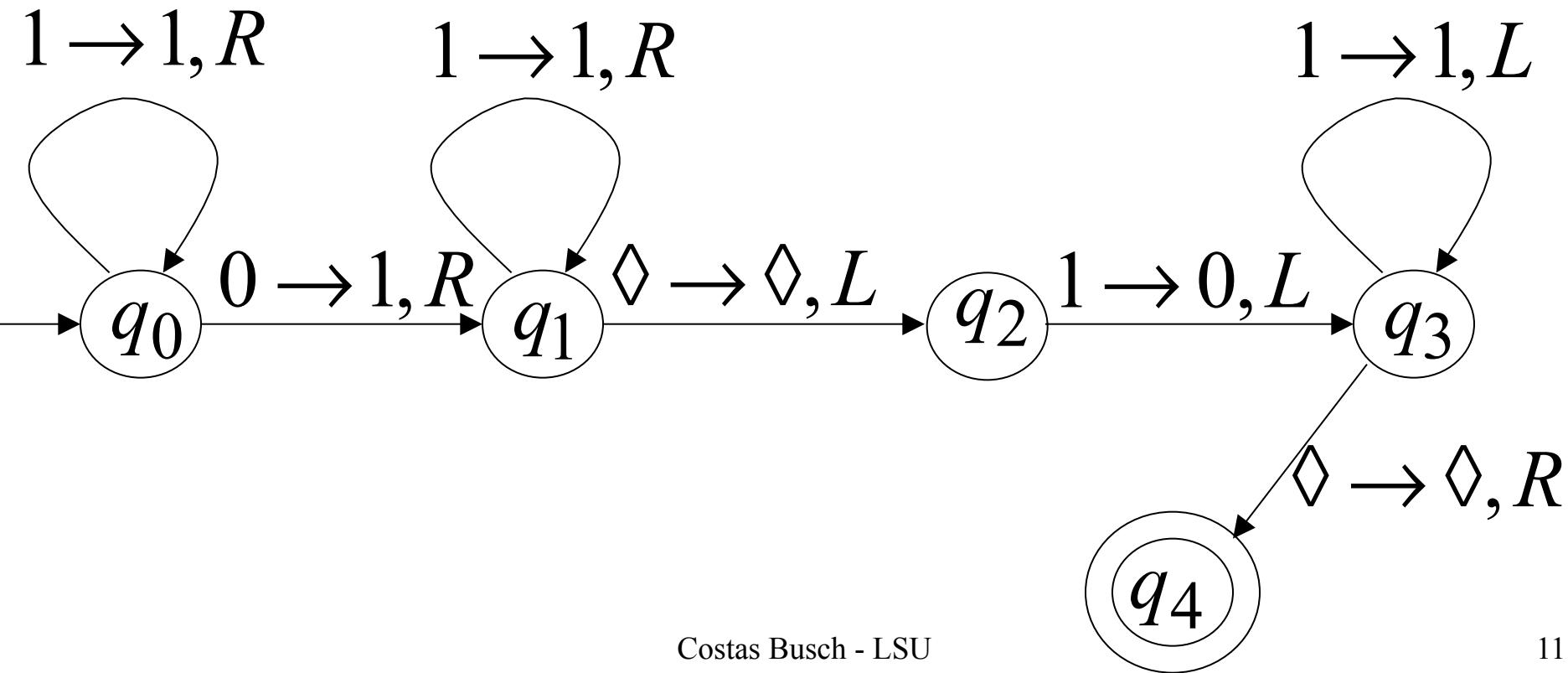

x
y
Start
◊ 1 1 ⋯ 1 0 1 ⋯ 1 ◊
$q_0$ initial state


$x + y$
Finish
◊ 1 1 ⋯ 1 1 0 ◊
$q_f$ final state

The 0 here helps when we use the result for other operations



# Turing machine for function $f(x,y)=x+y$

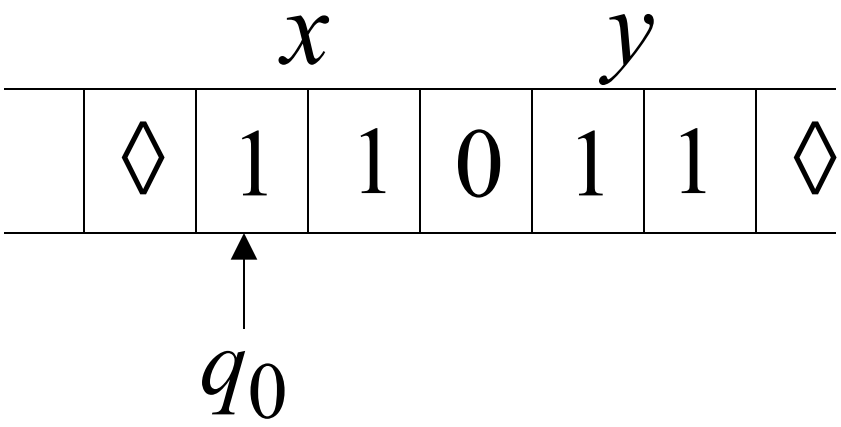


# Execution Example:

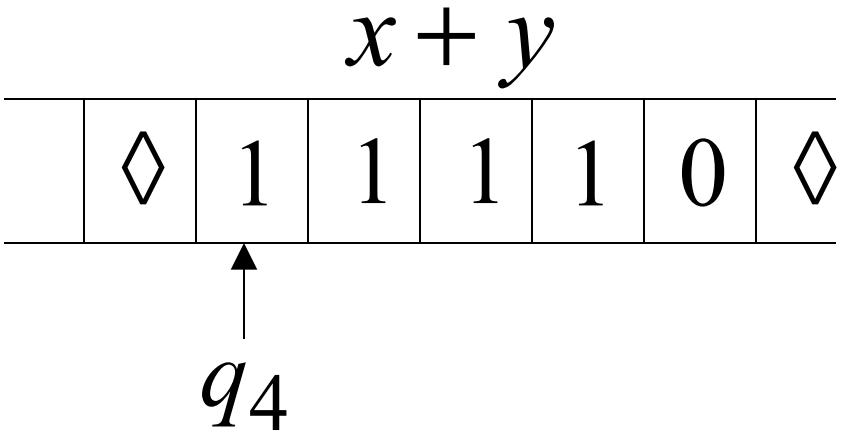
Time 0

$x = 11$ (=2)

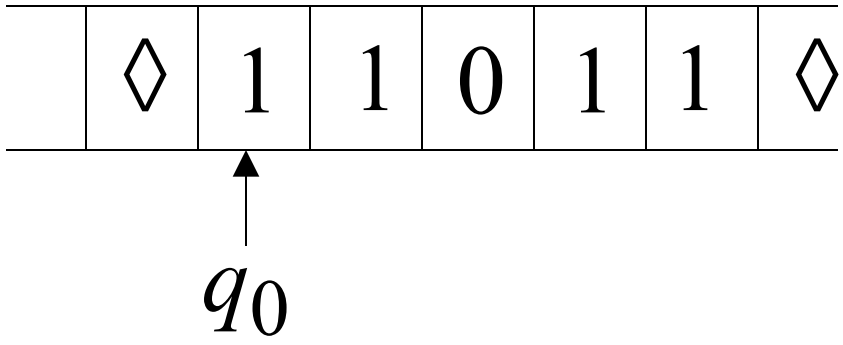
$y = 11$ (=2)

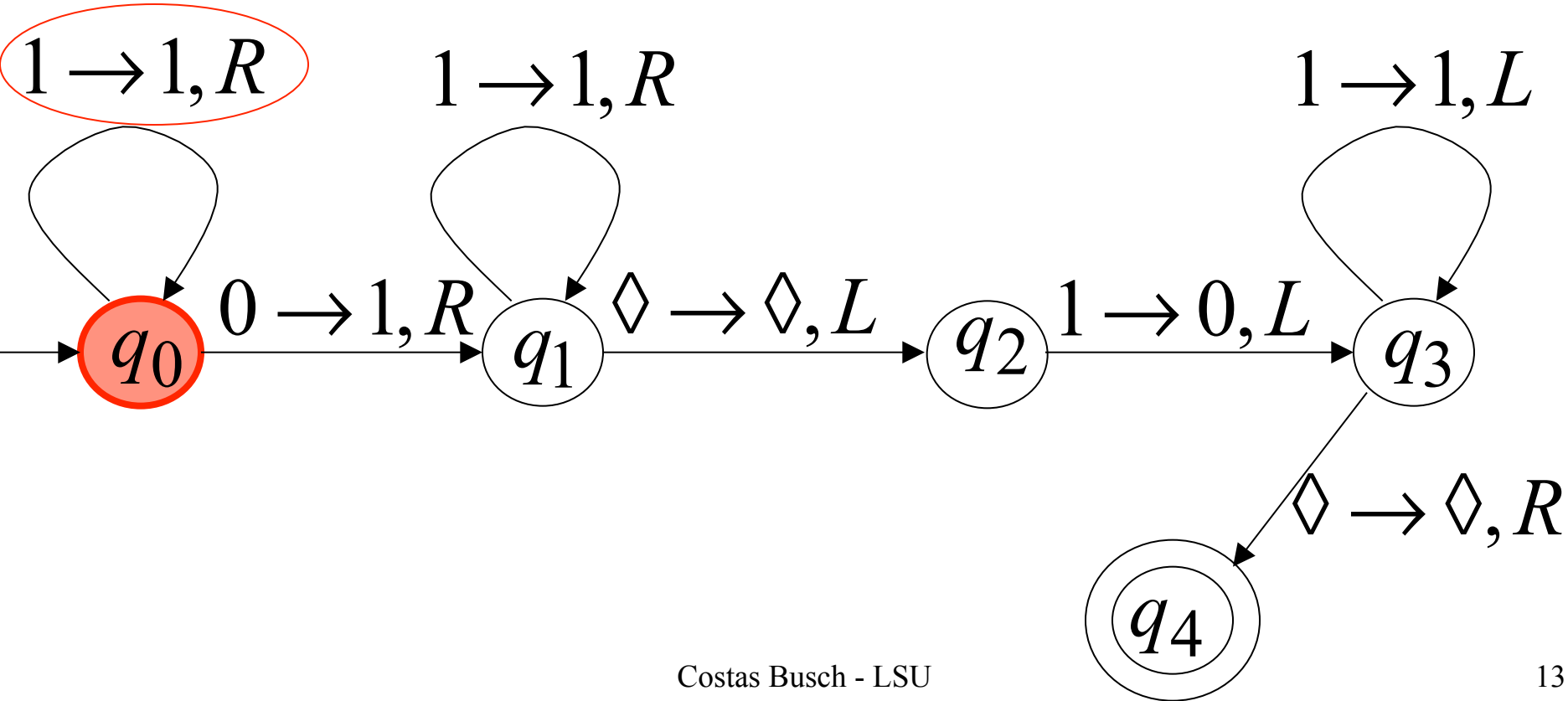



Final Result

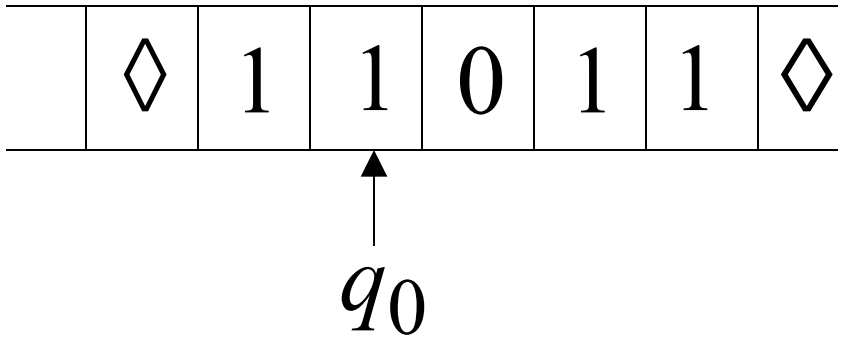


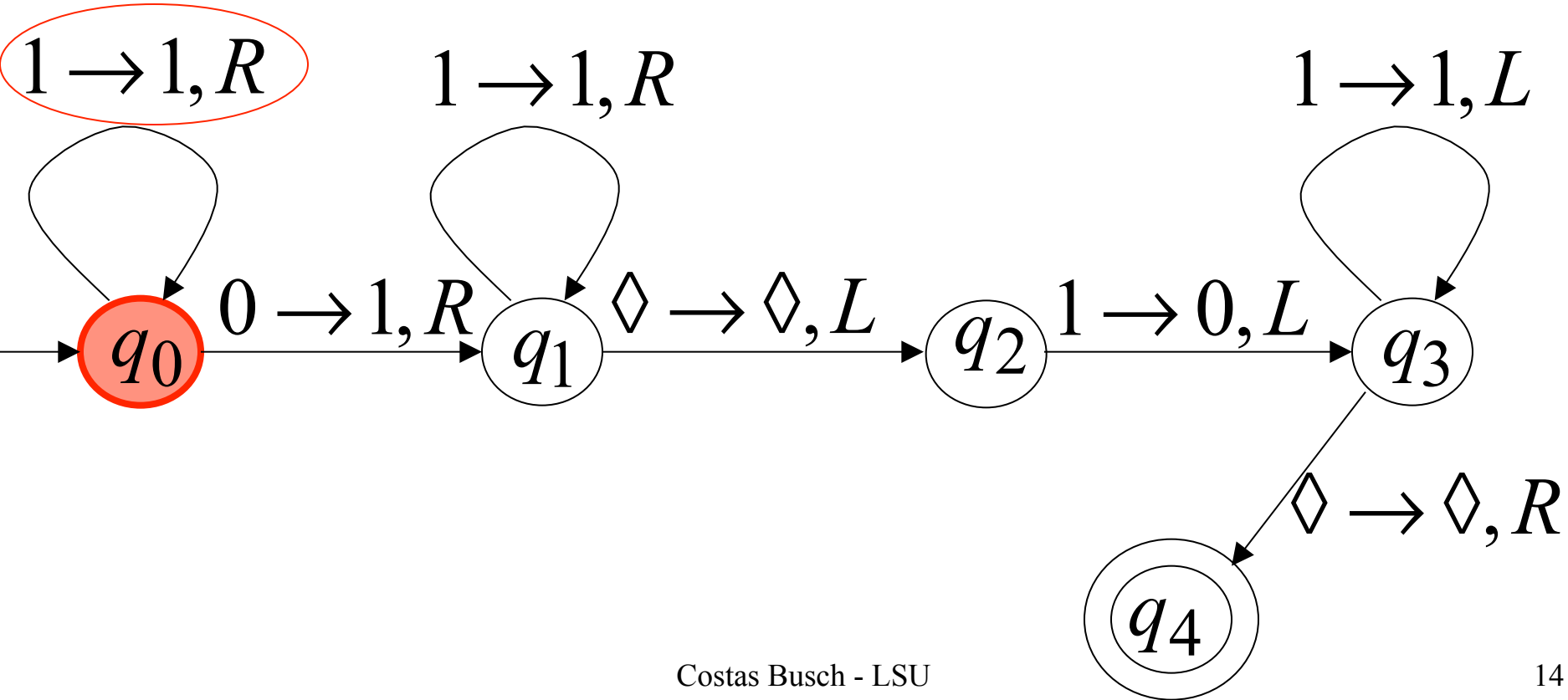
Time 0


◊ 1 1 0 1 1 ◊
$q_0$

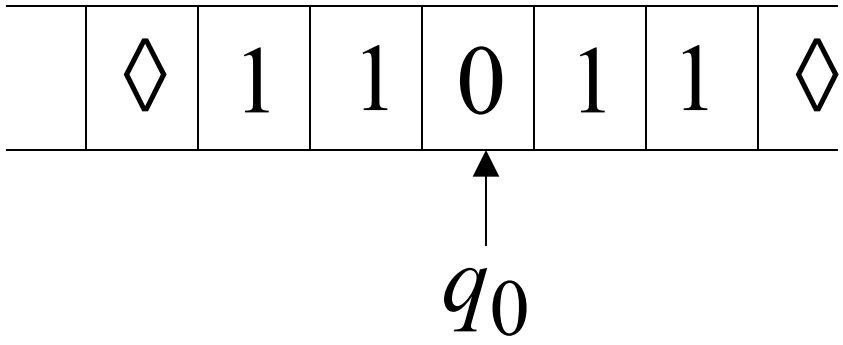

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

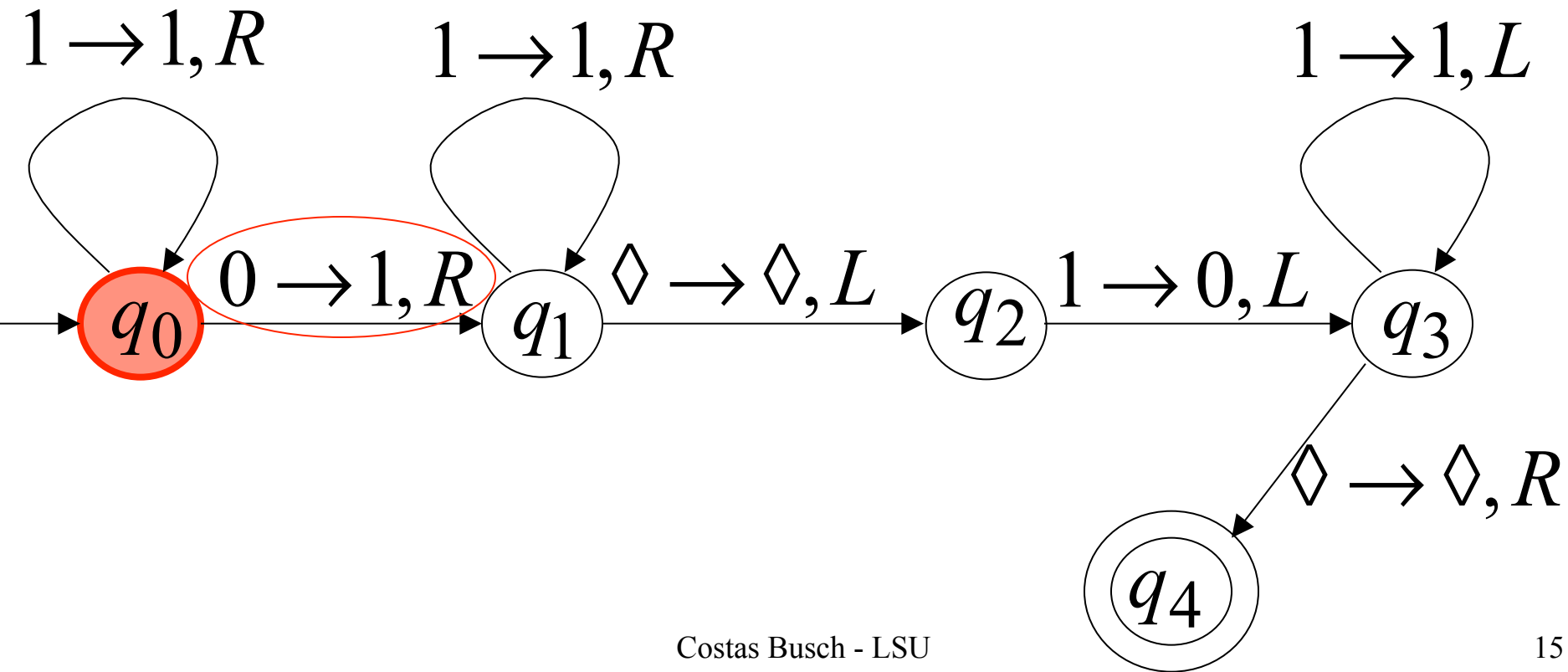
Time 1


◊ 1 1 0 1 1 ◊
$q_0$

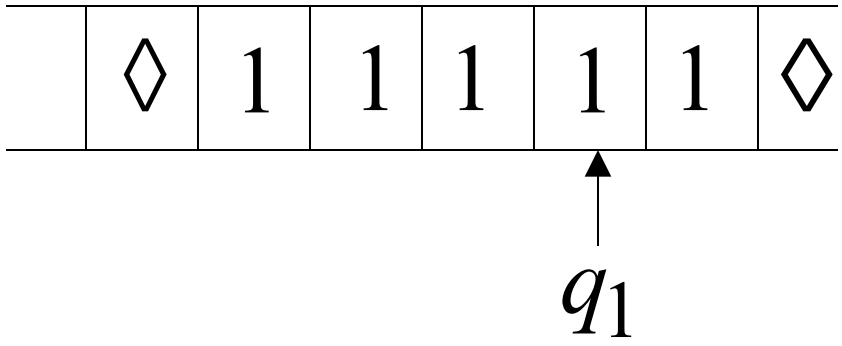

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

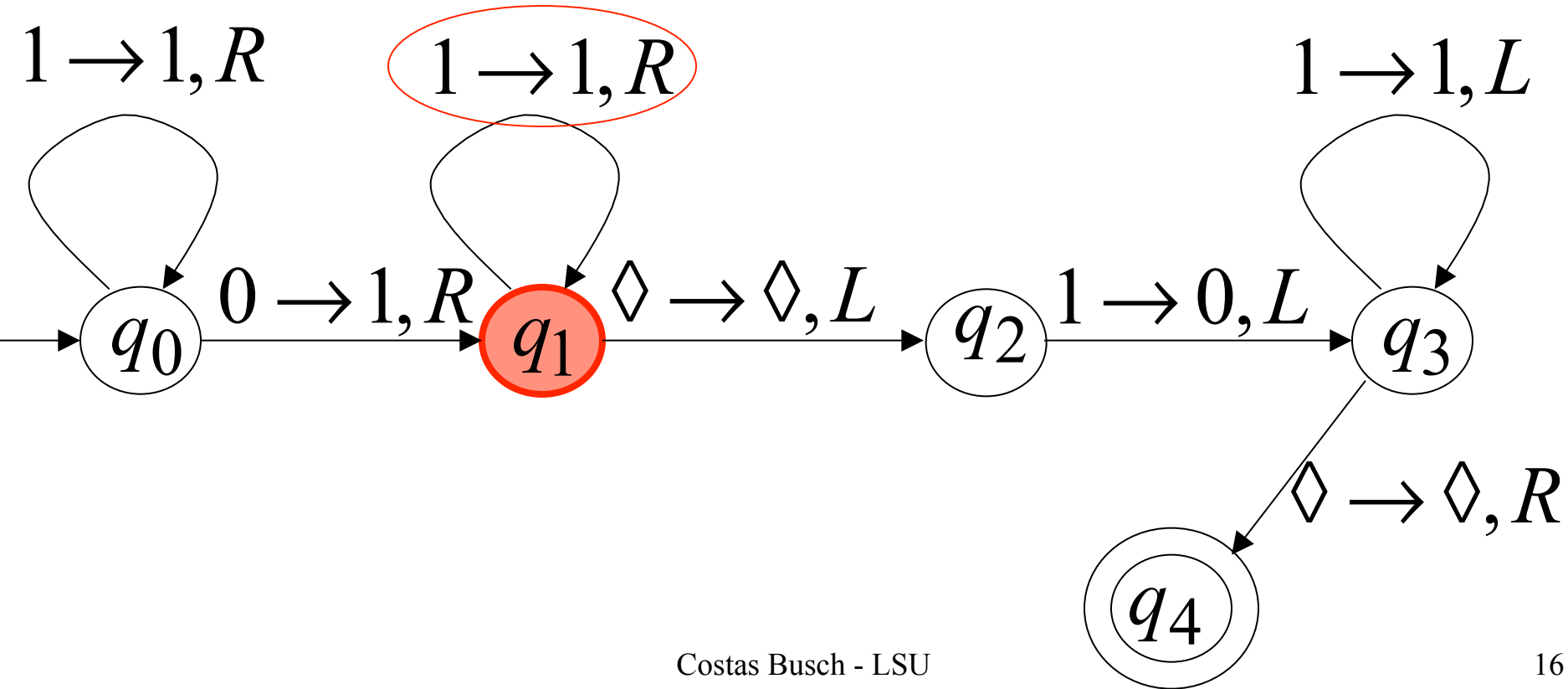
Time 2


◊ 1 1 0 1 1 ◊
$q_0$

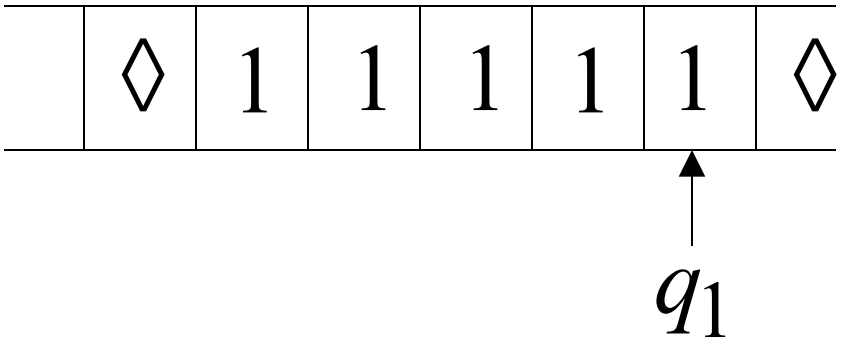

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

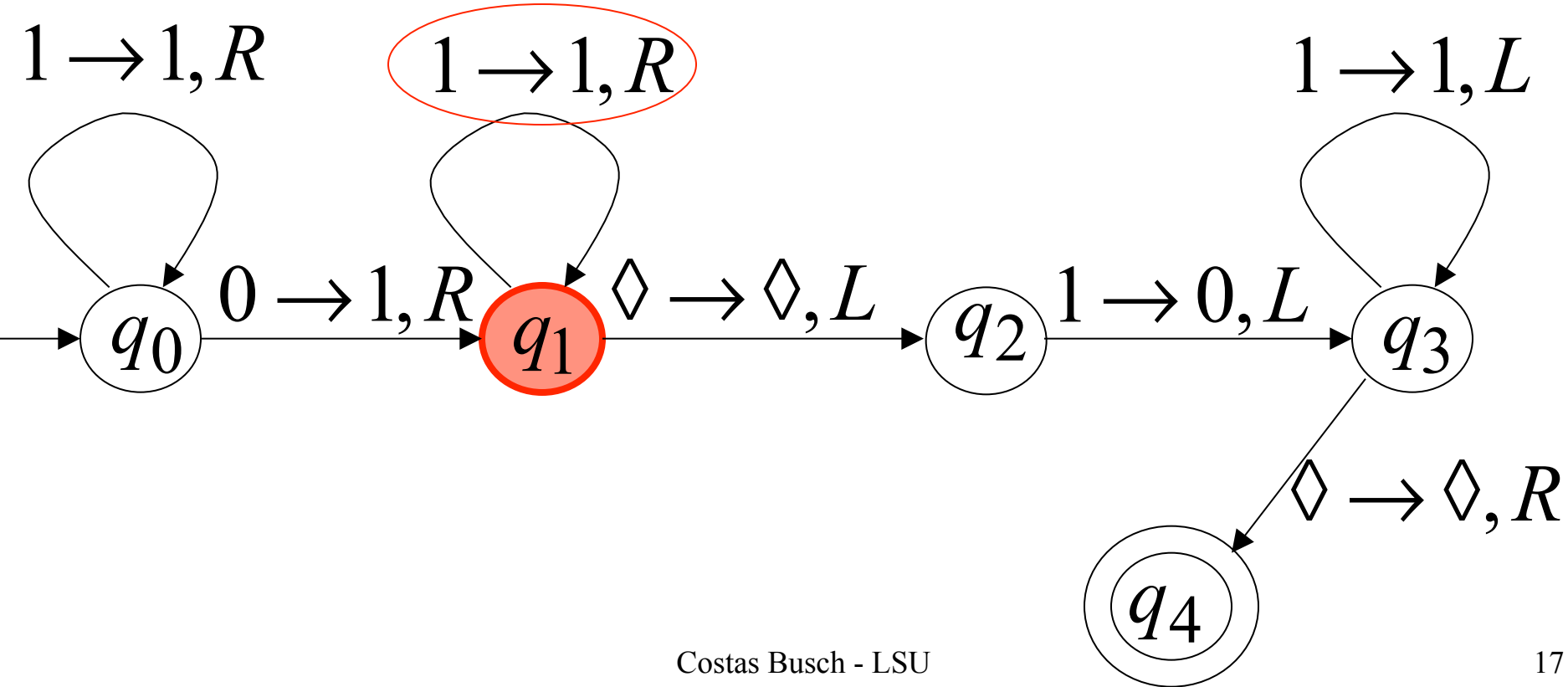
Time 3


◊ 1 1 1 1 1 ◊
$q_1$

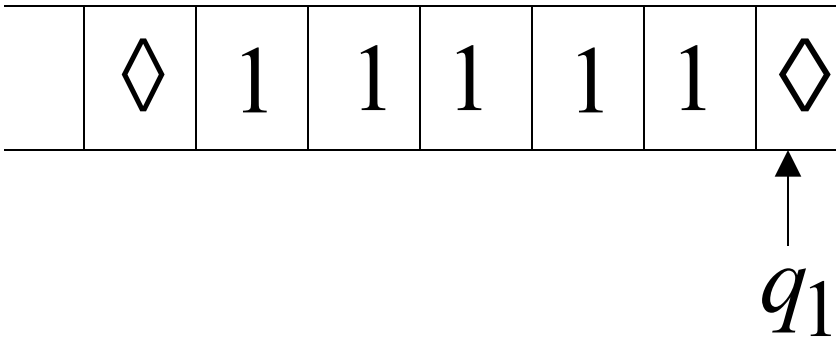

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

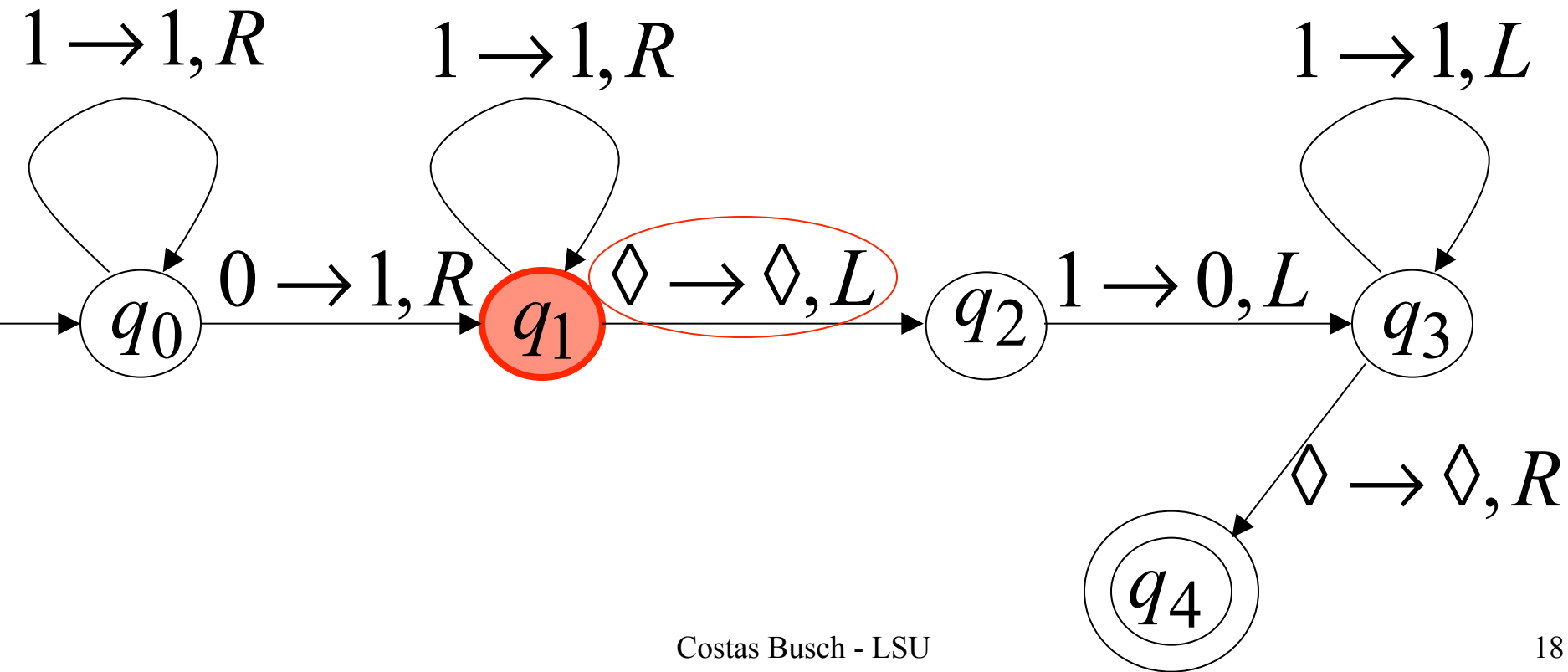
Time 4


◊ 1 1 1 1 1 ◊
$q_1$

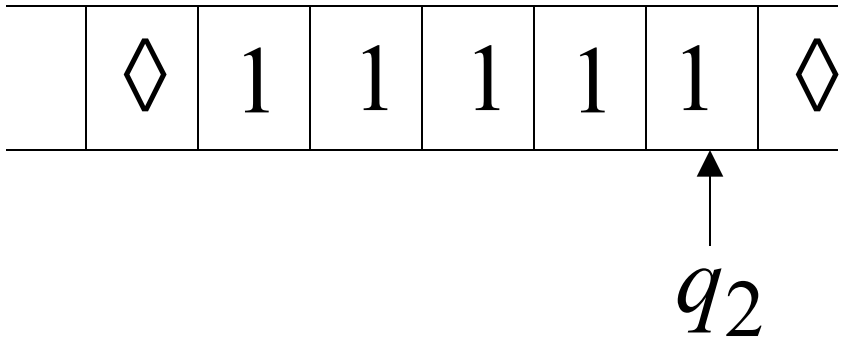

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

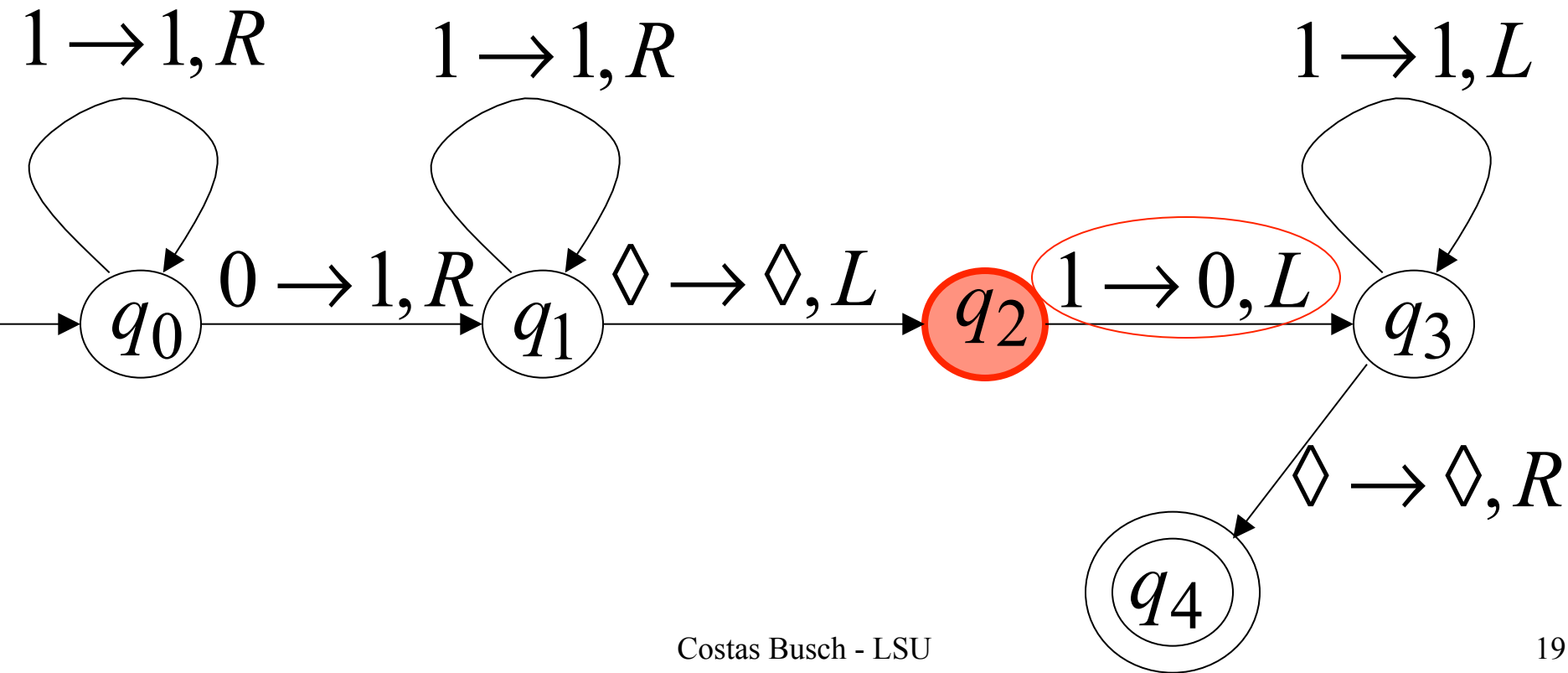
Time 5


◊ 1 1 1 1 1 ◊
$q_1$

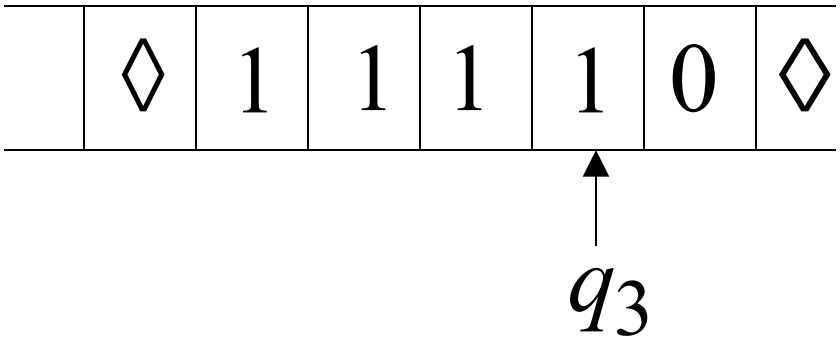

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

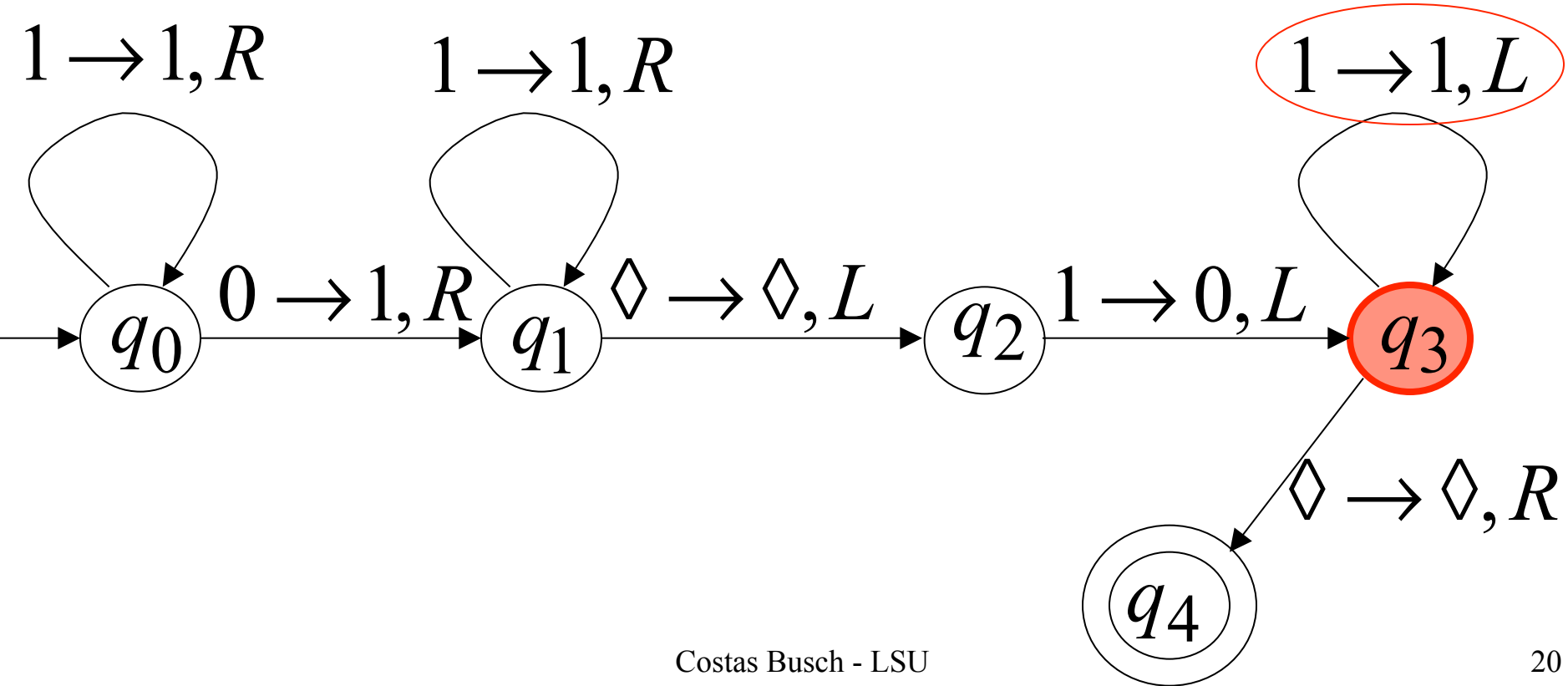
Time 6


◊ 1 1 1 1 1 ◊
$q_2$

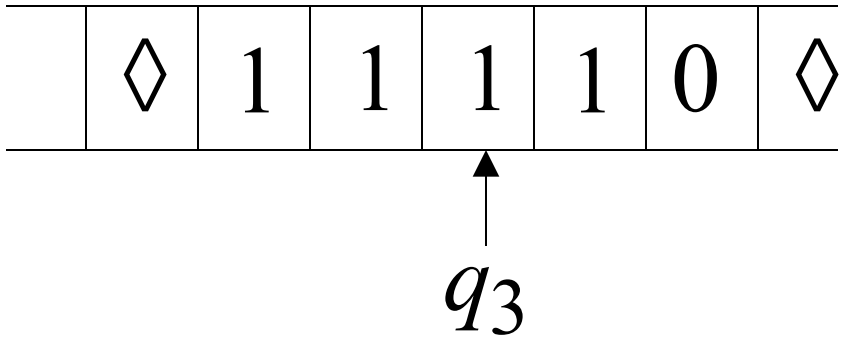

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

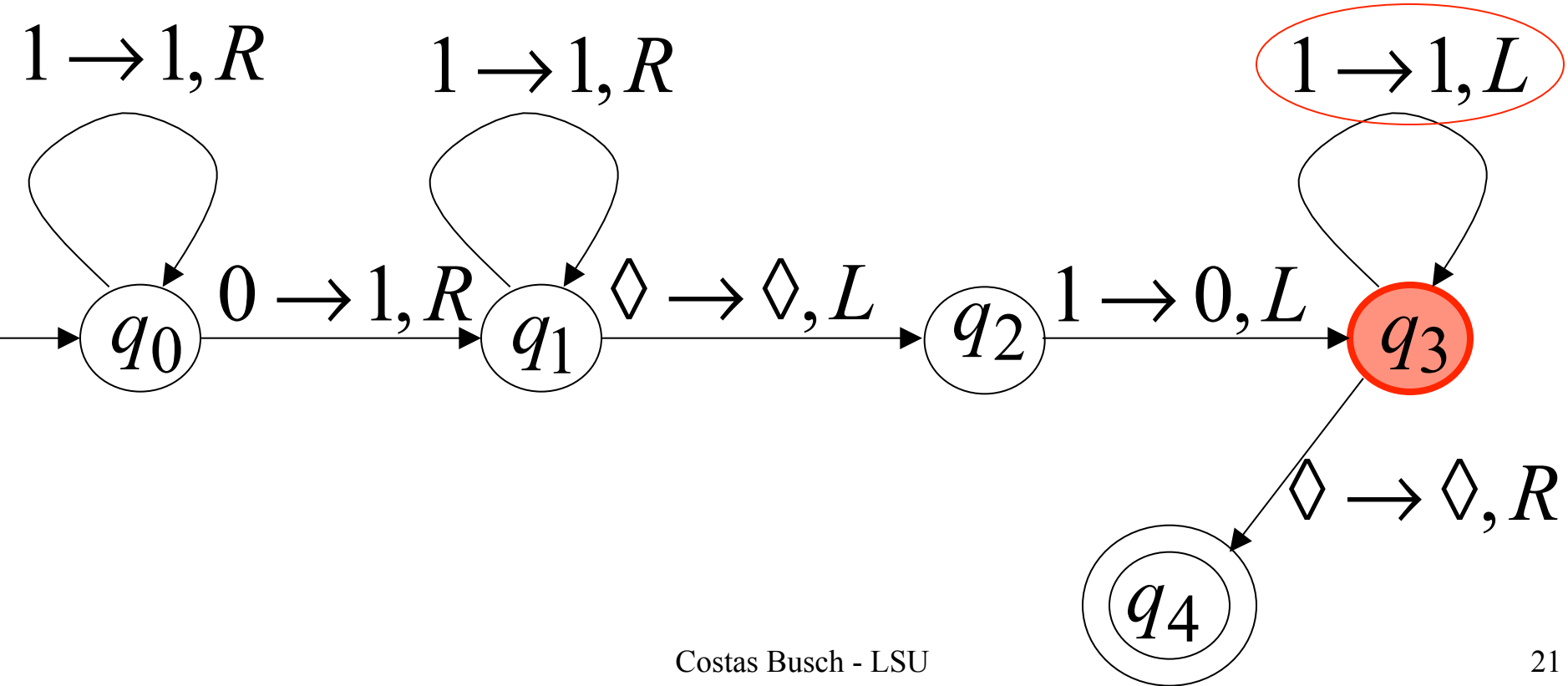
Time 7


◊ 1 1 1 1 0 ◊
$q_3$

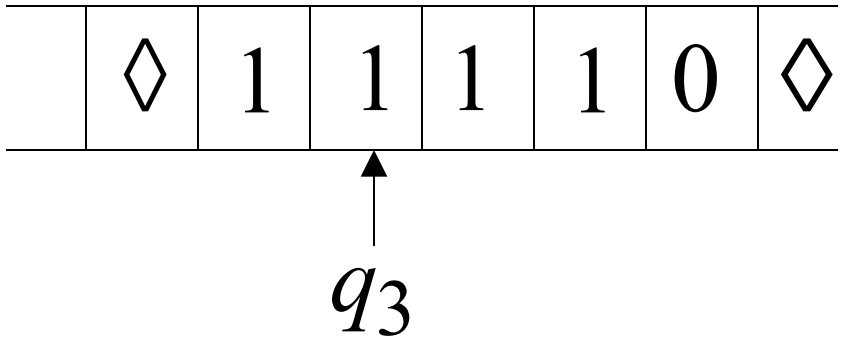

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

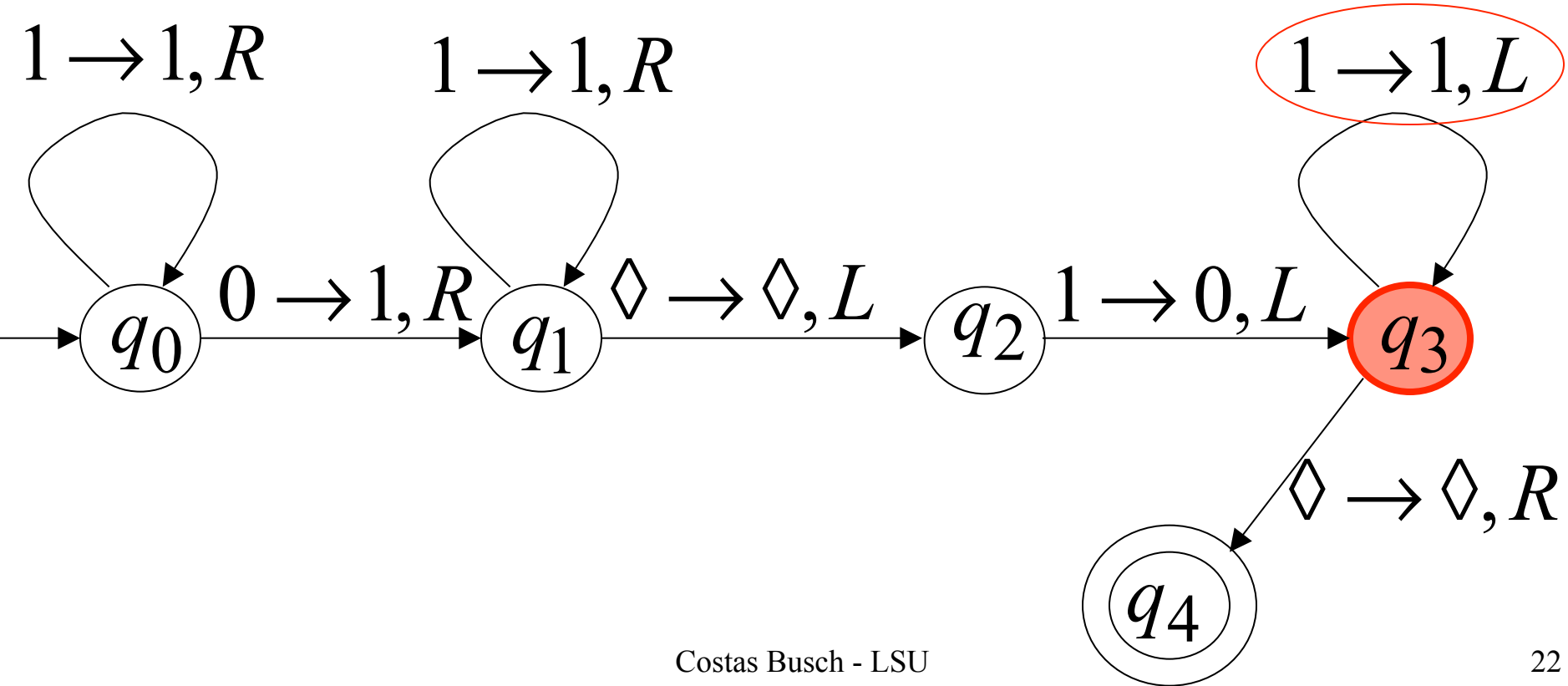
Time 8


◊ 1 1 1 1 0 ◊
$q_3$

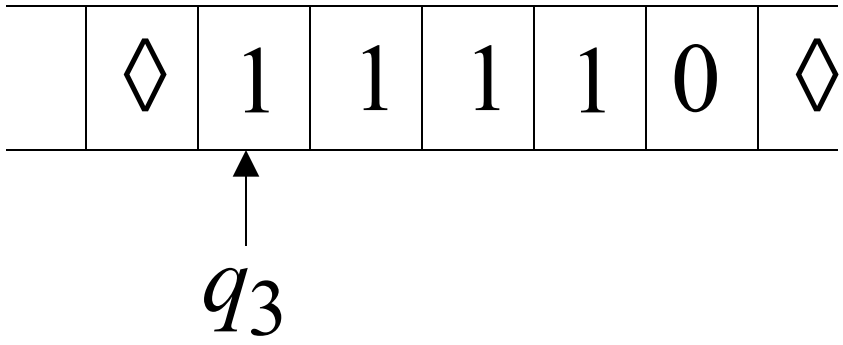

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

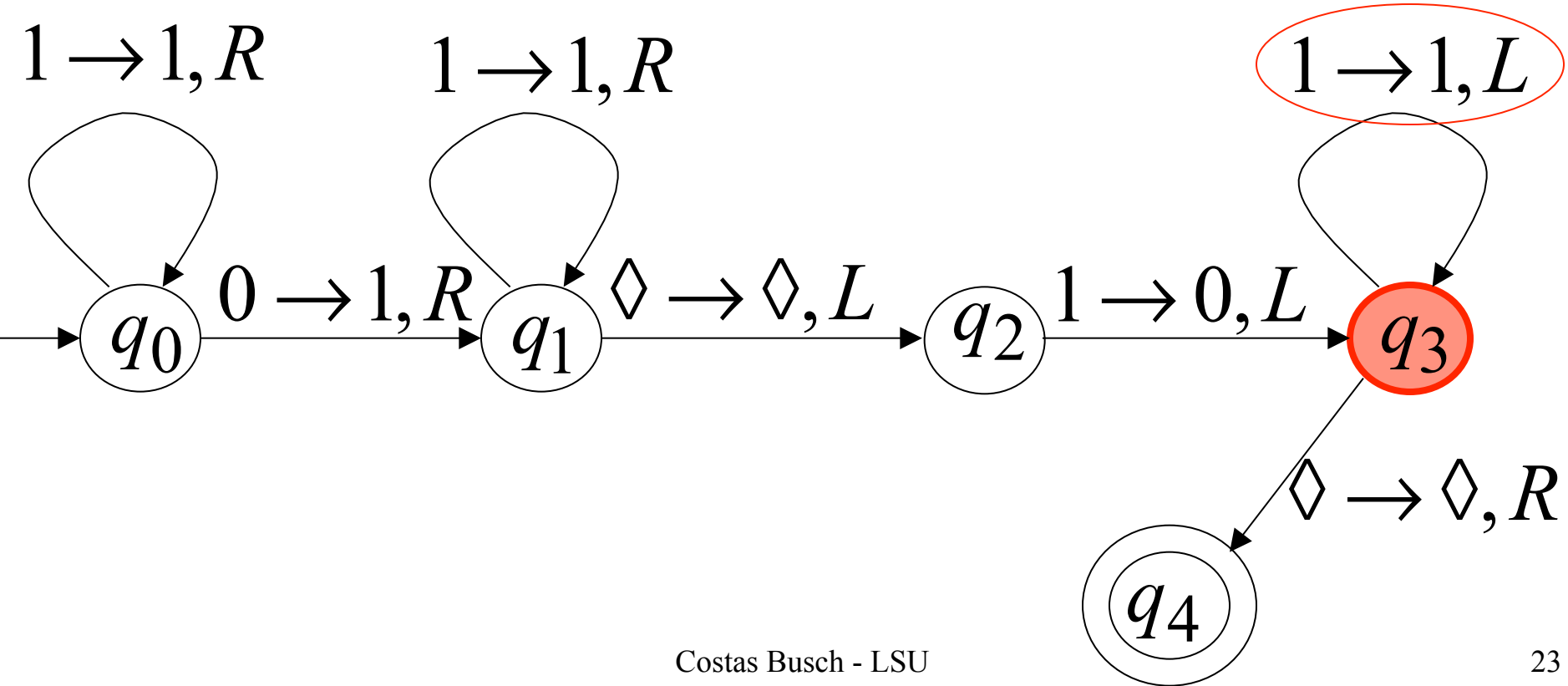
Time 9


◊ 1 1 1 1 0 ◊
$q_3$

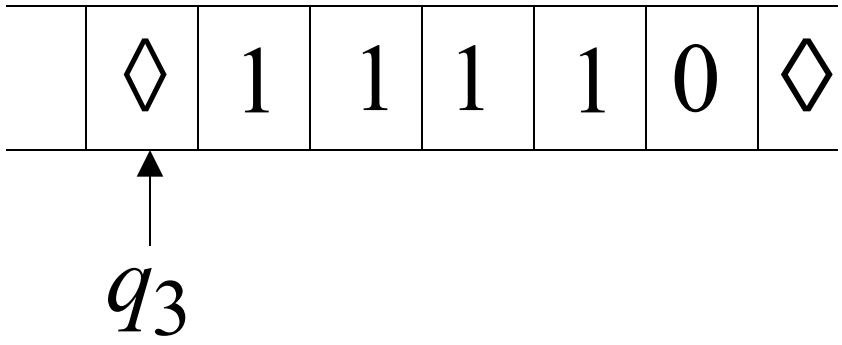

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

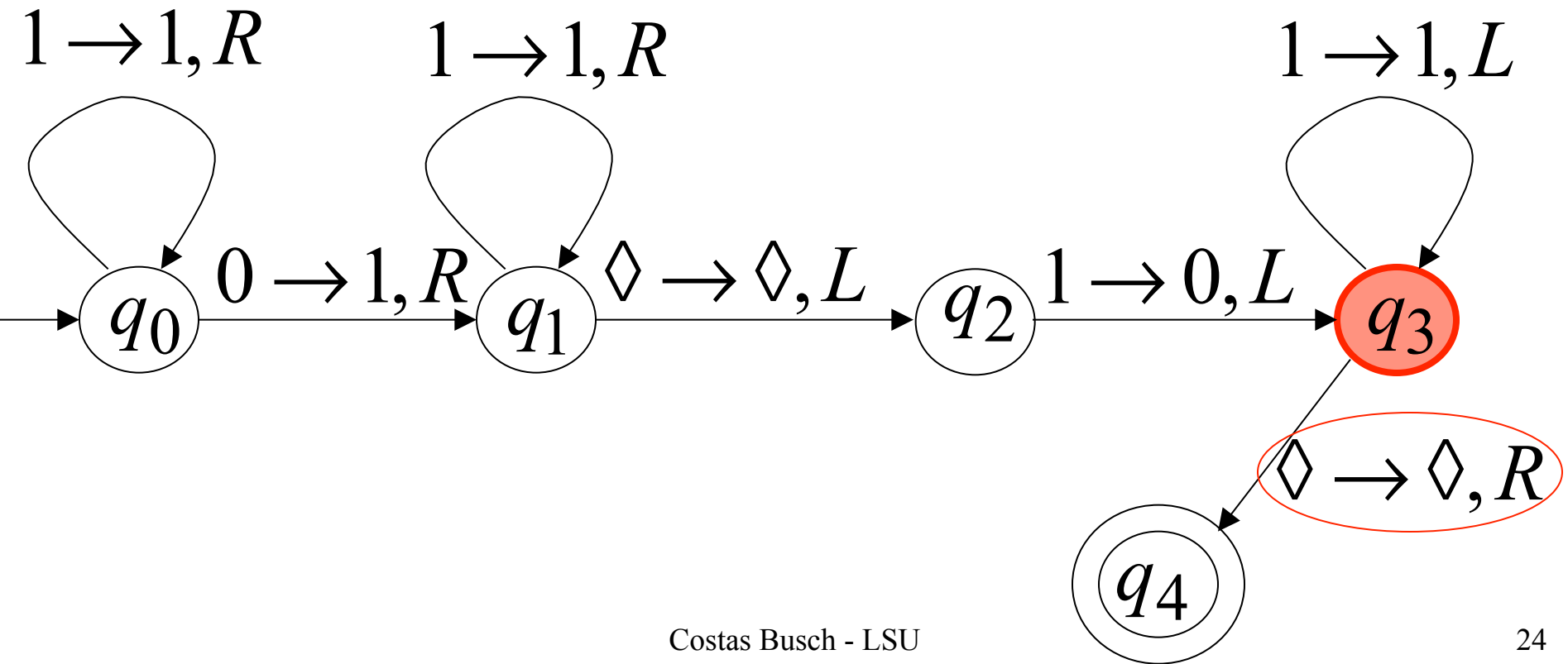
Time 10


◊ 1 1 1 1 0 ◊
$q_3$

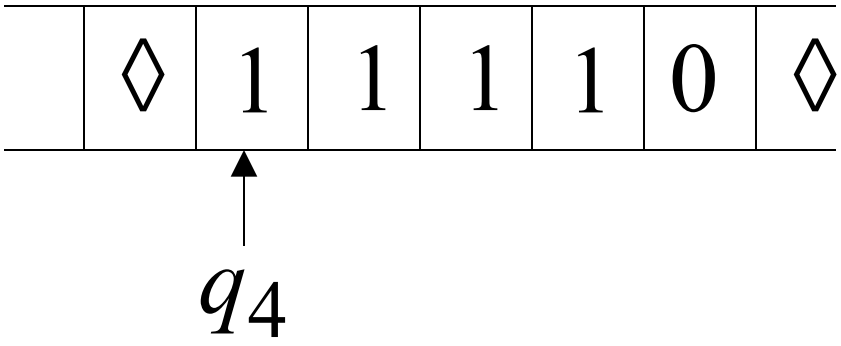

$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

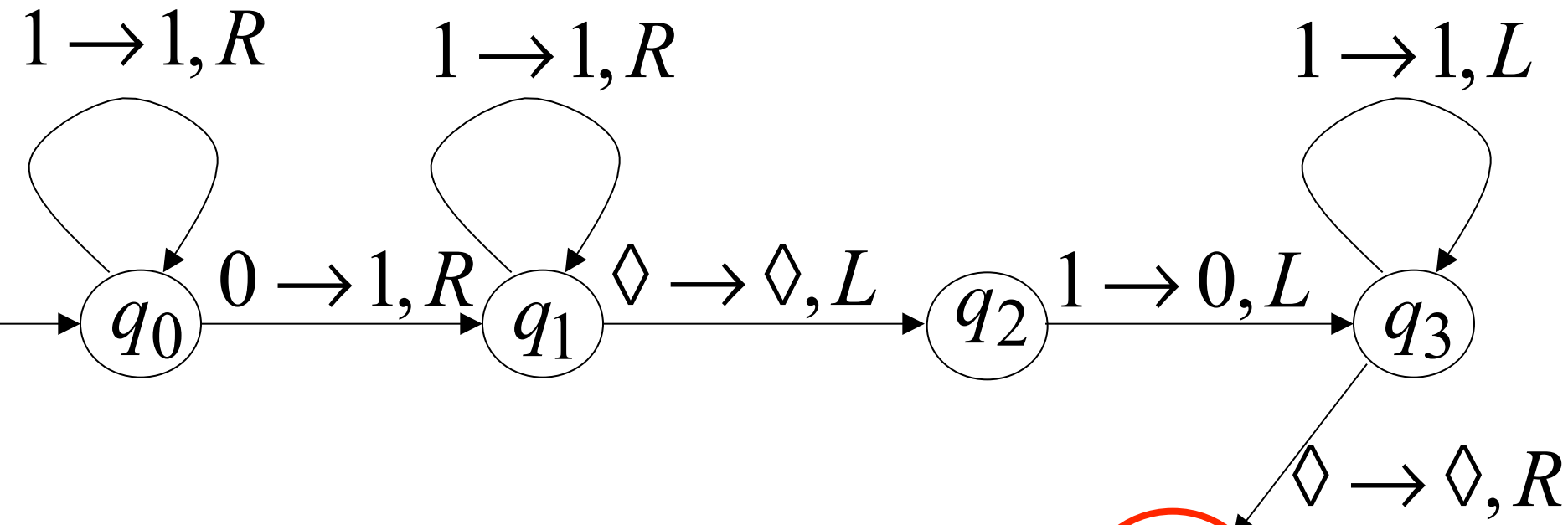
Time 11


◊ 1 1 1 1 0 ◊
$q_3$


$1 \to 1, R$
$1 \to 1, R$
$1 \to 1, L$
$0 \to 1, R$
$\Diamond \to \Diamond, L$
$1 \to 0, L$
$\Diamond \to \Diamond, R$
$q_0$
$q_1$
$q_2$
$q_3$
$q_4$

Time 12




HALT & accept

# Another Example

The function $f(x) = 2x$ is computable

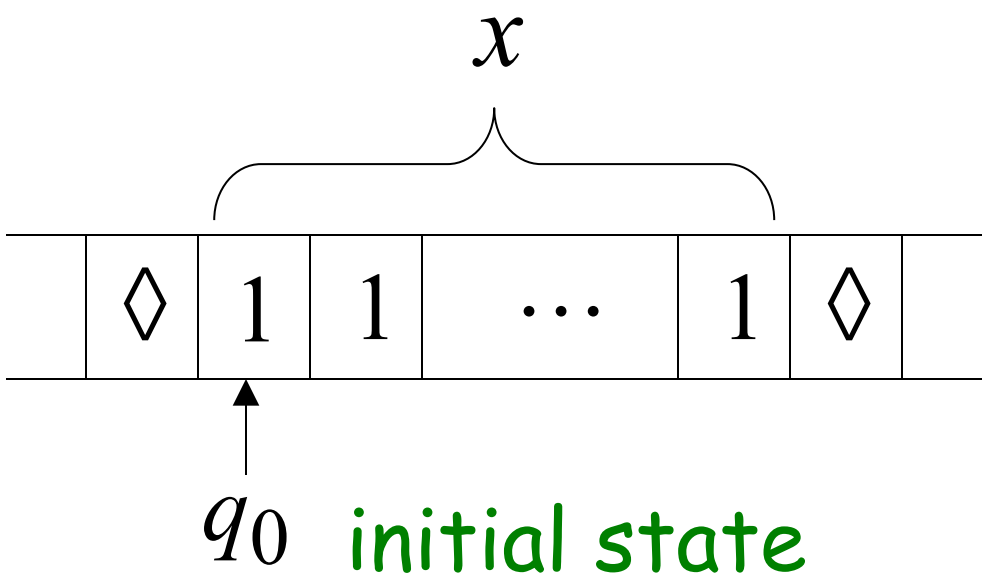
$x$ is integer

Turing Machine:

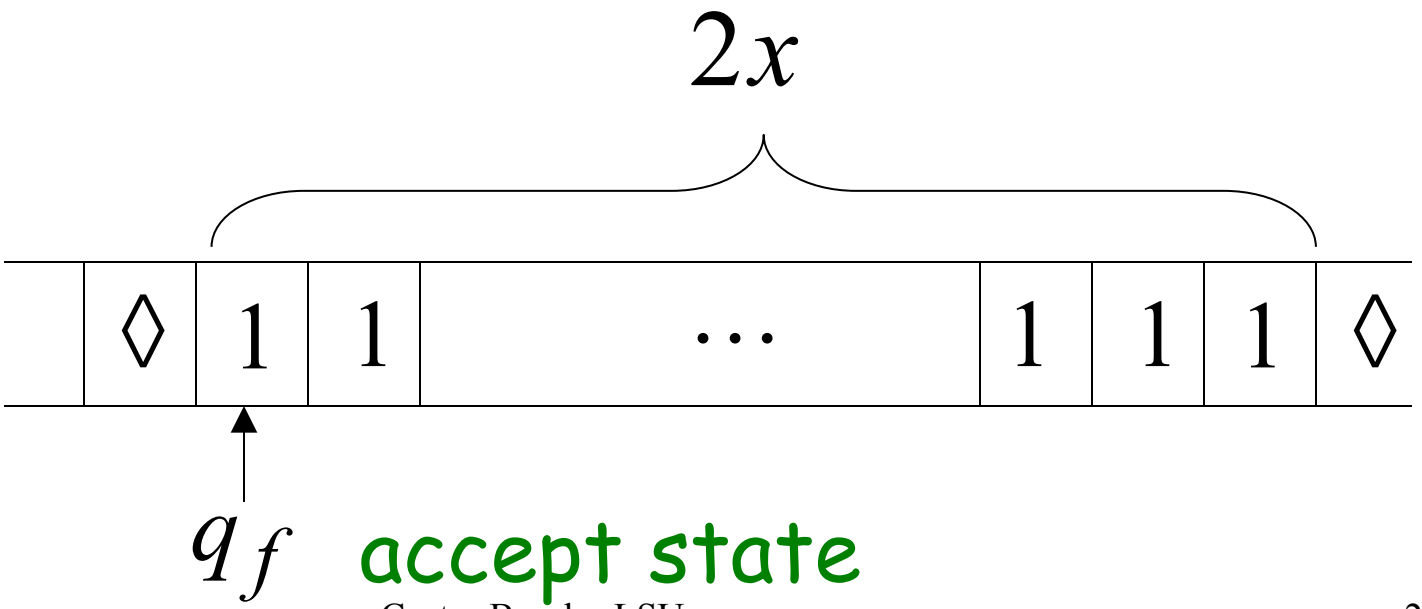
Input string: $x$ unary

Output string: $xx$ unary

Start




Finish

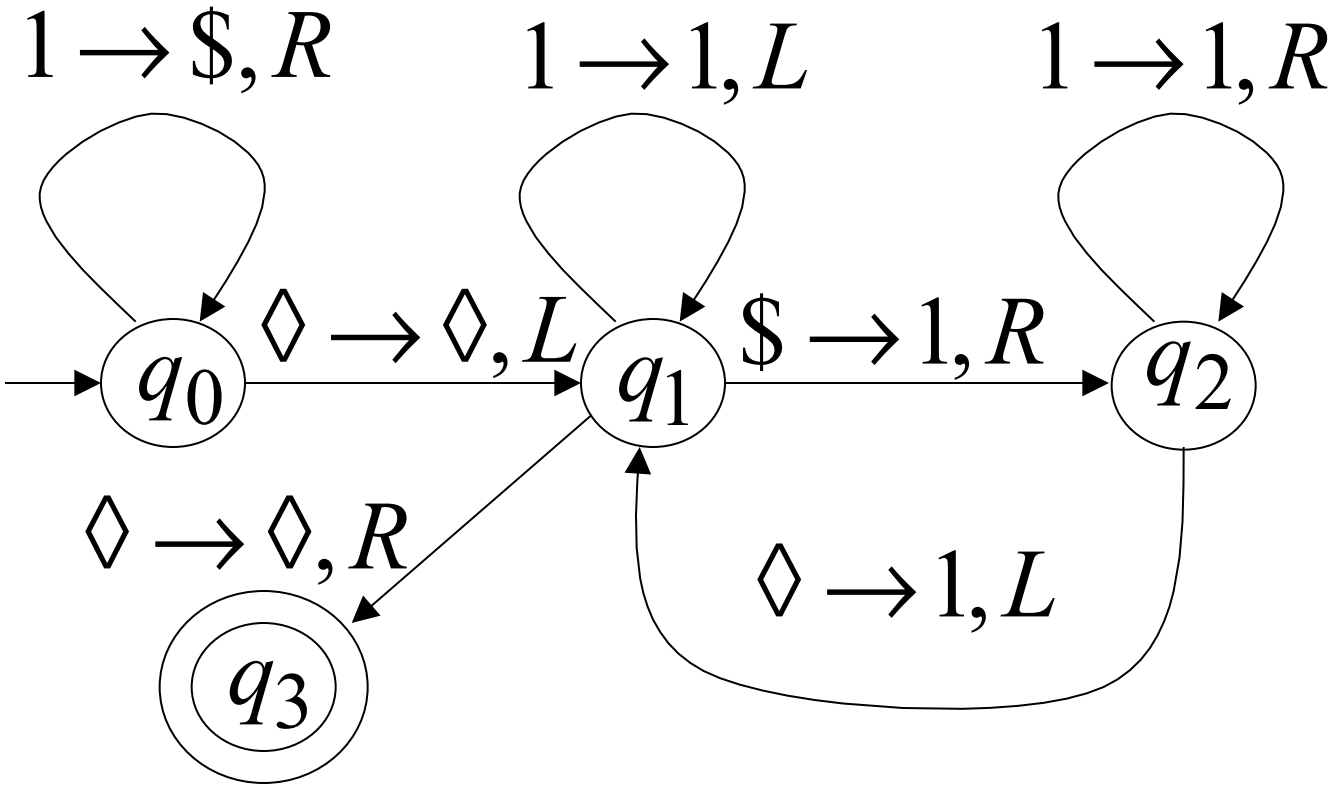


# Turing Machine Pseudocode for $f(x) = 2x$

- Replace every 1 with $
- Repeat:
    - Find rightmost $, replace it with 1
    - Go to right end, insert 1

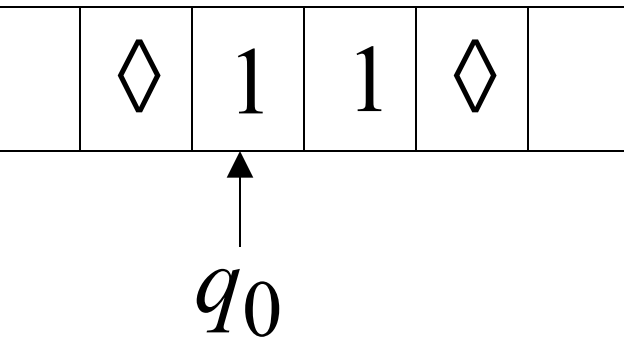
  Until no more $ remain

# Turing Machine for $f(x) = 2x$

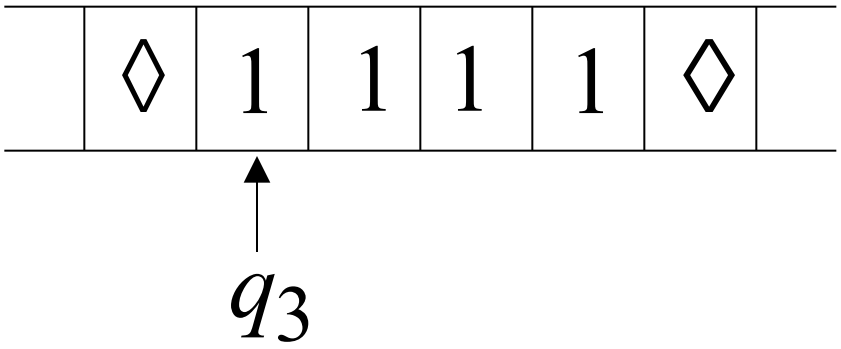


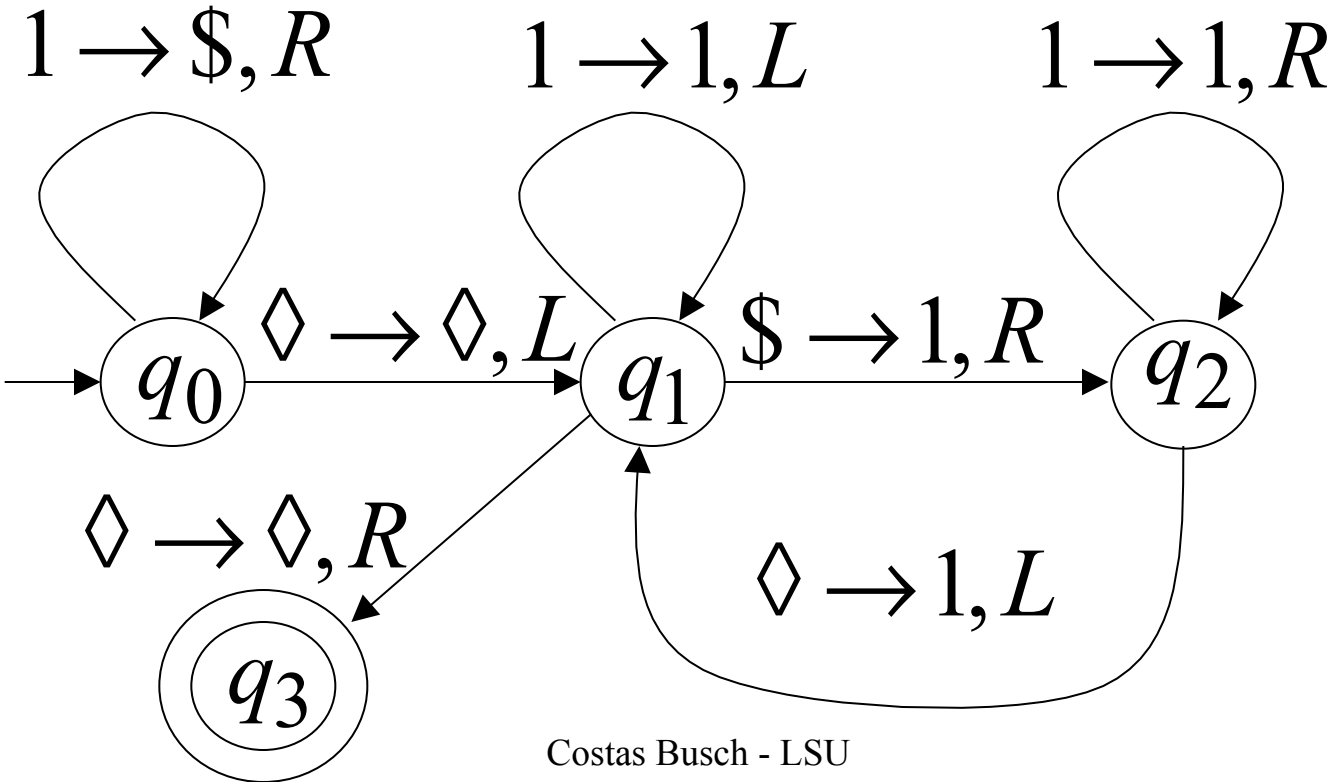
# Example

Start




Finish



# Another Example

The function is computable

$$f(x,y)=\begin{cases}1 & \text{if } x>y\\ 0 & \text{if } x\le y\end{cases}$$

Input: $x0y$

Output: $1$ or $0$

# Turing Machine Pseudocode:

- Repeat

  Match a 1 from $x$ with a 1 from $y$

  Until all of $x$ or $y$ is matched

- If a 1 from $x$ is not matched

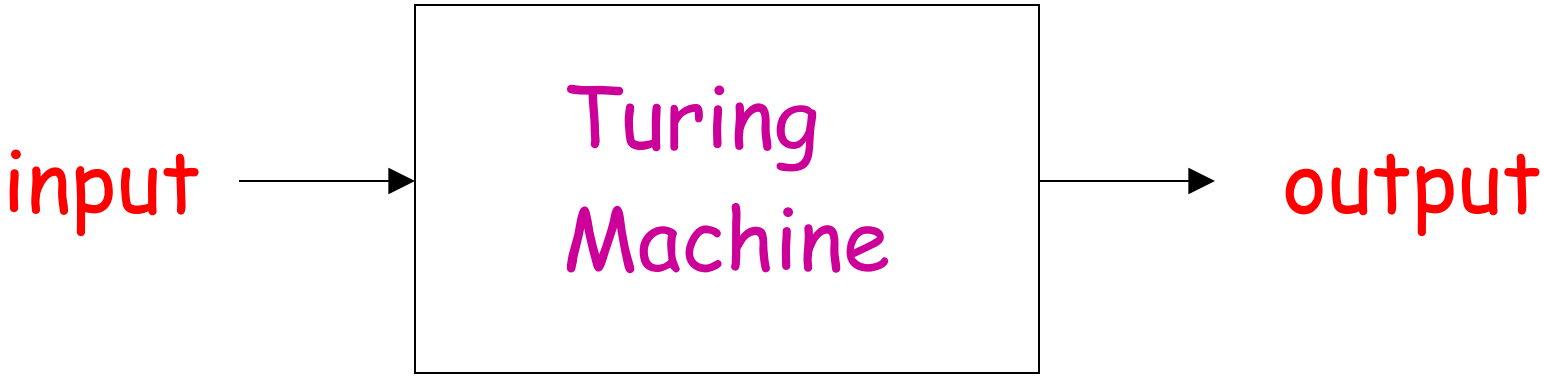
  erase tape, write 1 $(x > y)$

  else

  erase tape, write 0 $(x \leq y)$

# Combining Turing Machines

# Block Diagram



Example:

$$f(x,y) = \begin{cases} x+y & \text{if } x > y \\ 0 & \text{if } x \le y \end{cases}$$