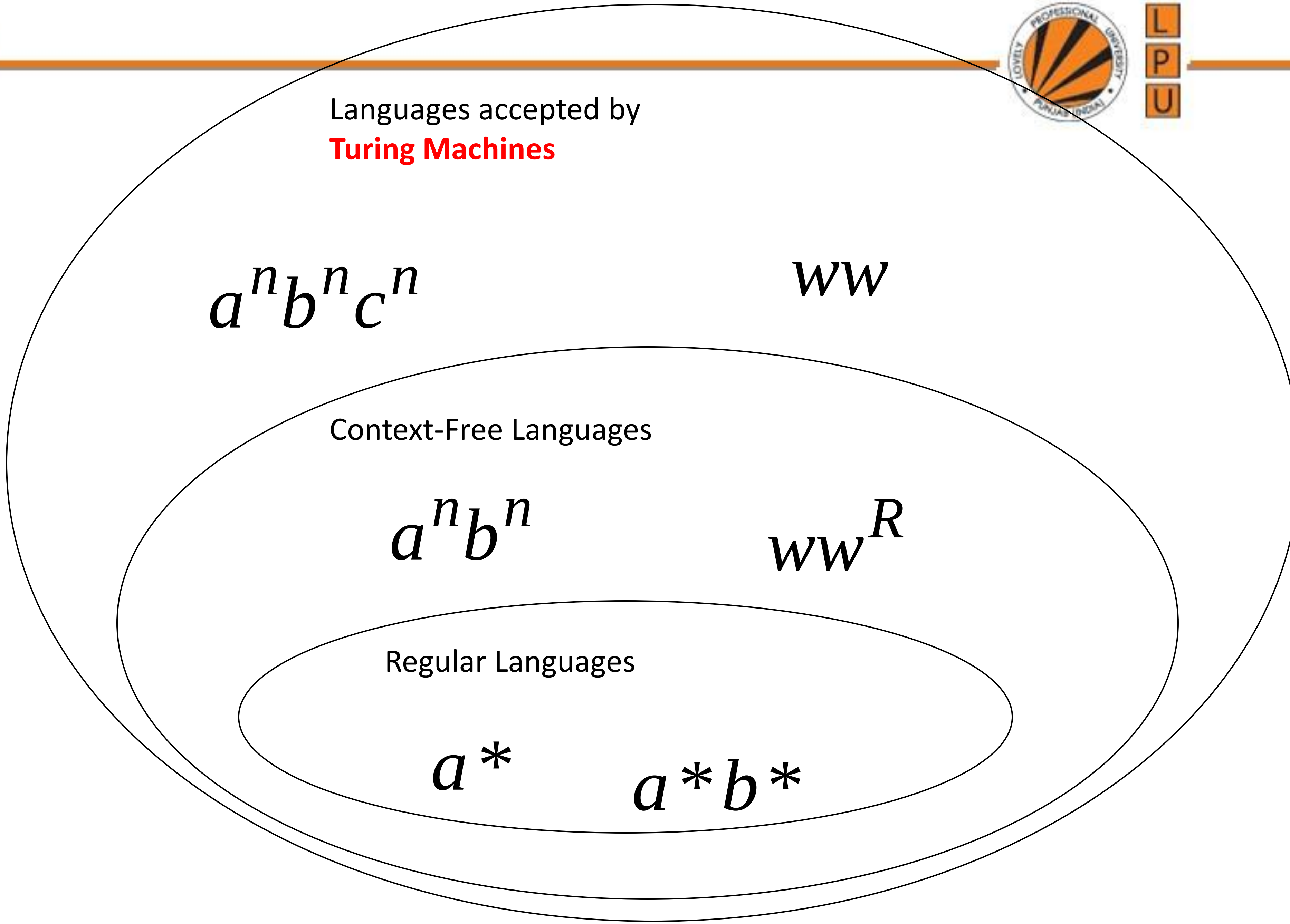
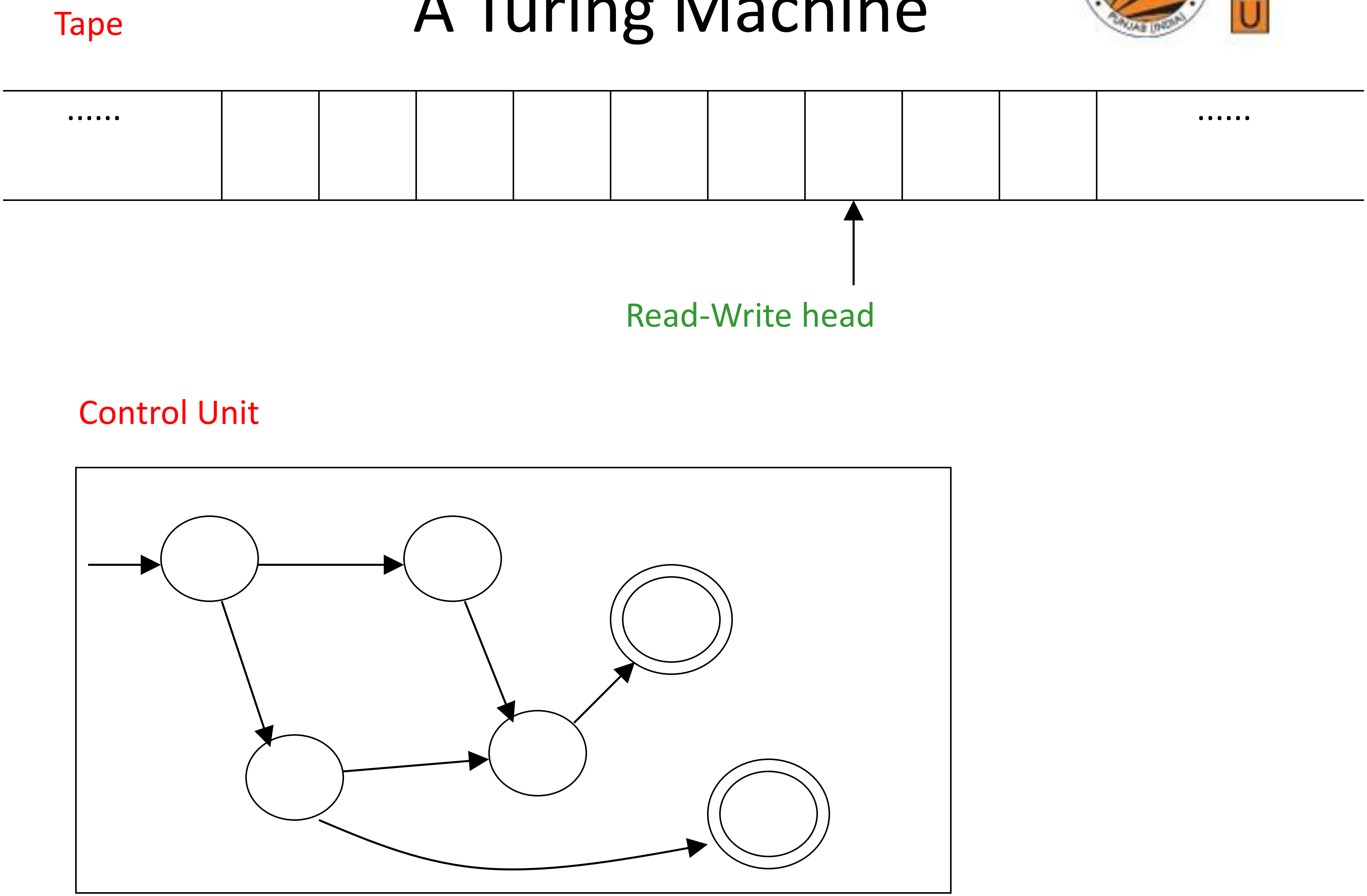
CSE322

Turing
Machine Model
&
Representation and Design of Turing Machines

Lecture #36

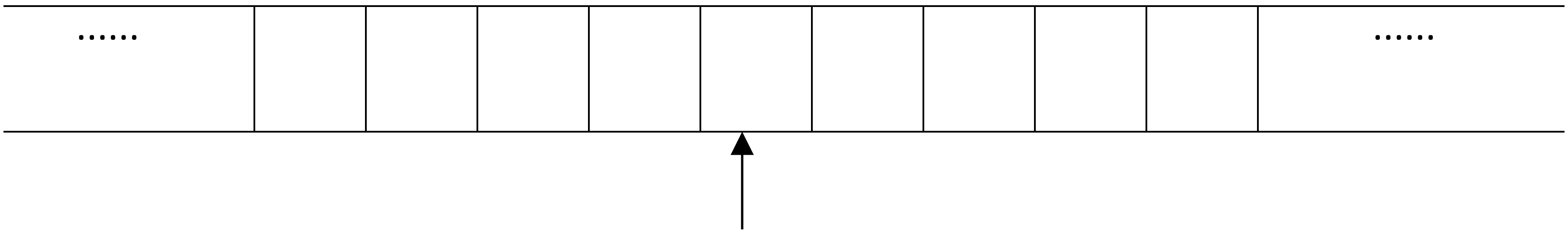

Languages accepted by
Turing Machines
$a^n b^n c^n$
ww
Context-Free Languages
$a^n b^n$
ww^R
Regular Languages
a^*
$a^* b^*$

A Turing Machine



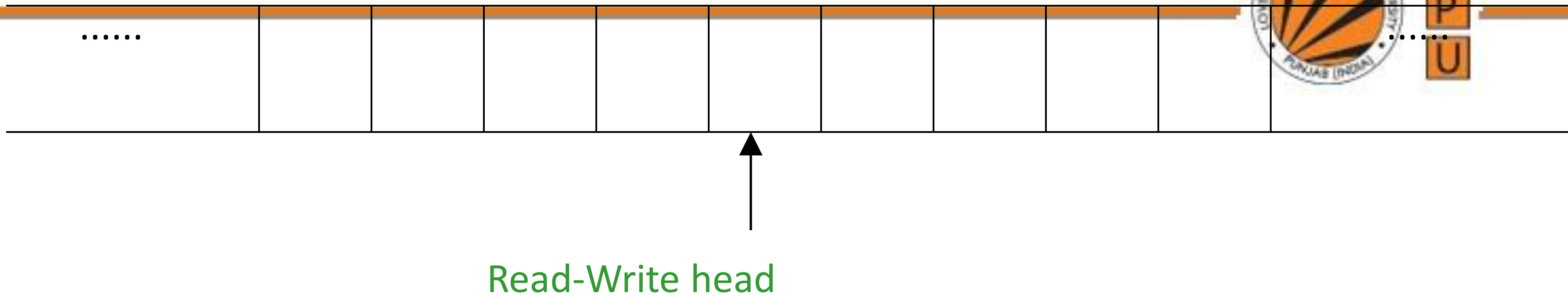
The Tape

No boundaries -- infinite length



Read-Write head

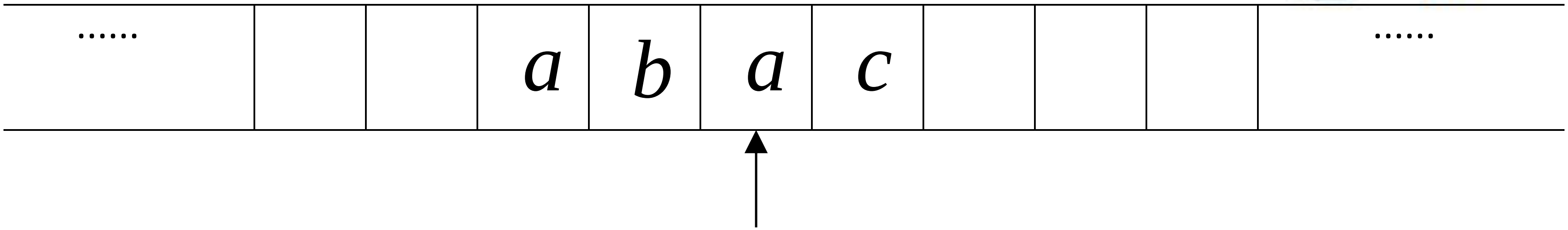
The head moves Left or Right



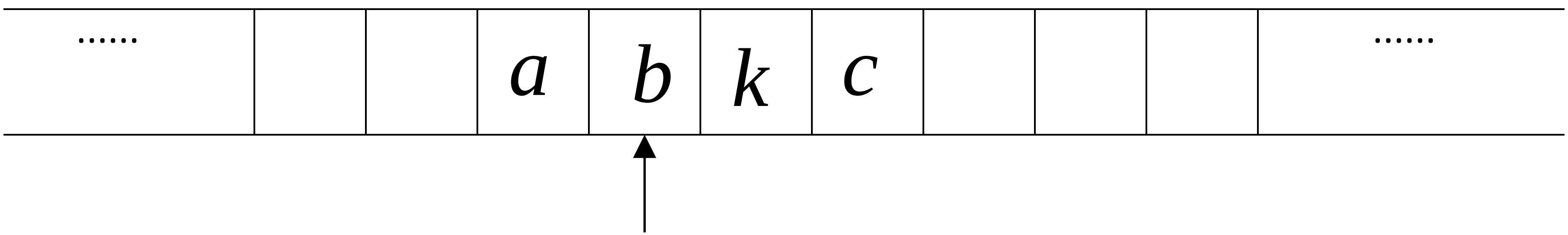
The head at each transition (time step):

1. Reads a symbol
2. Writes a symbol
3. Moves Left or Right

Time 0

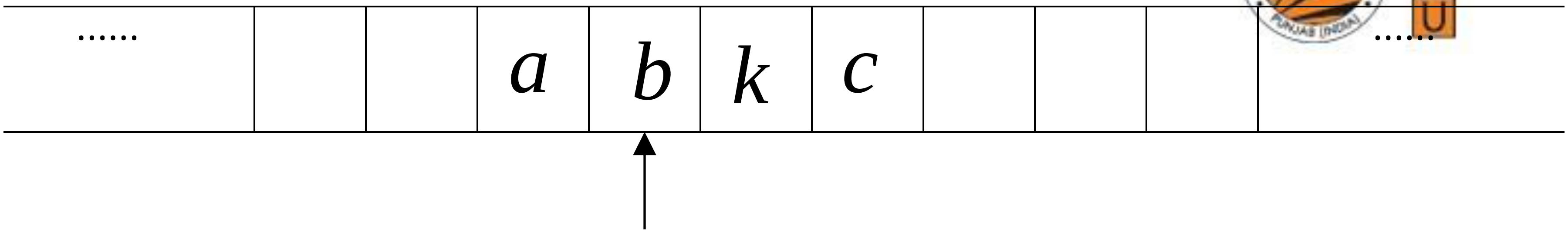


Time 1

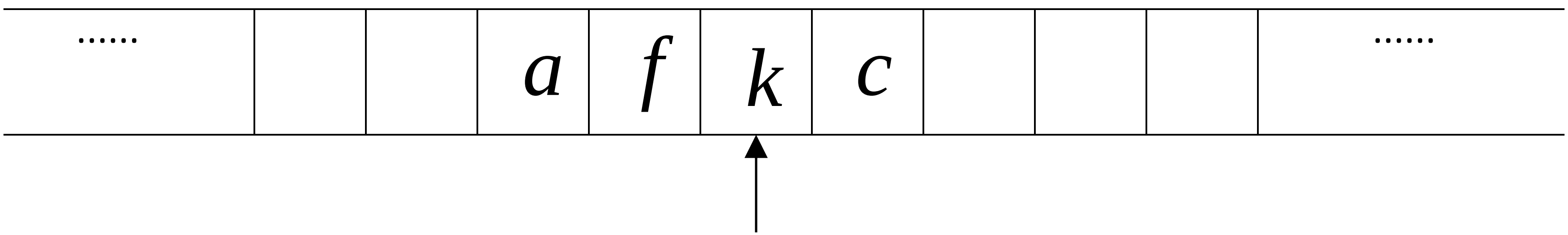


1. Reads a
2. Writes k
3. Moves Left

Time 1

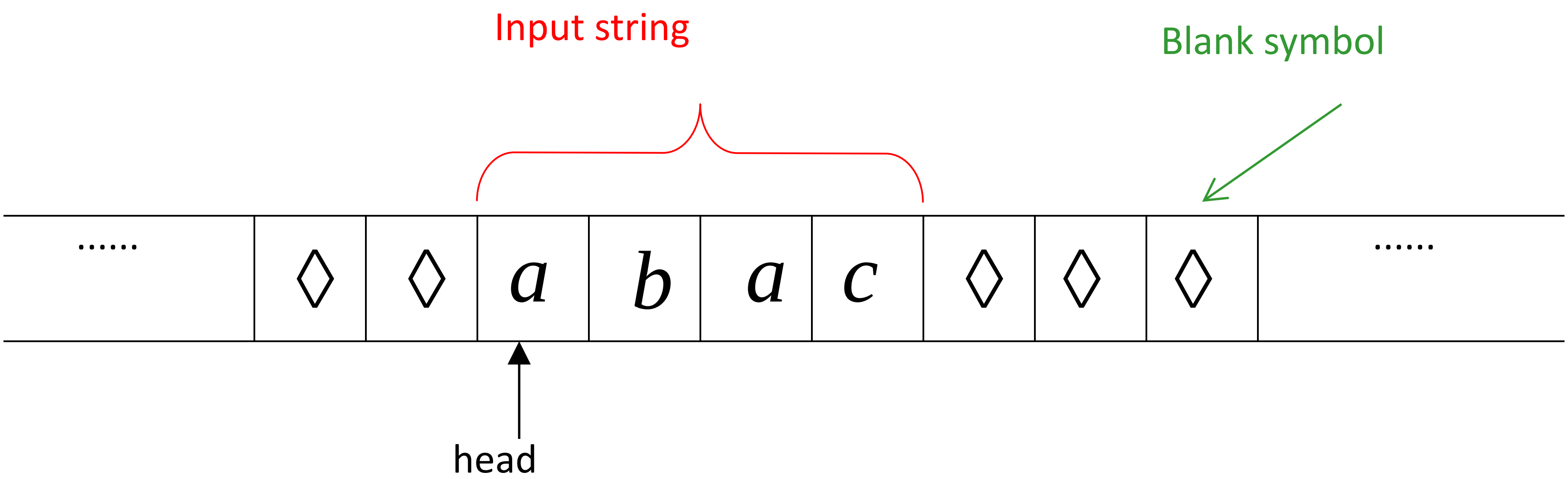


Time 2



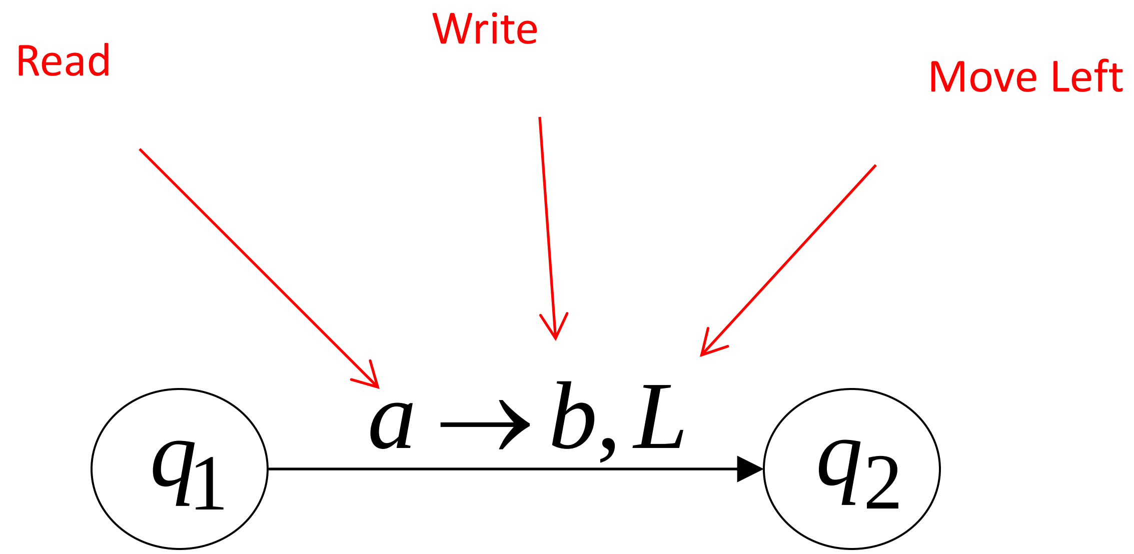
1. Reads b
2. Writes f
3. Moves Right

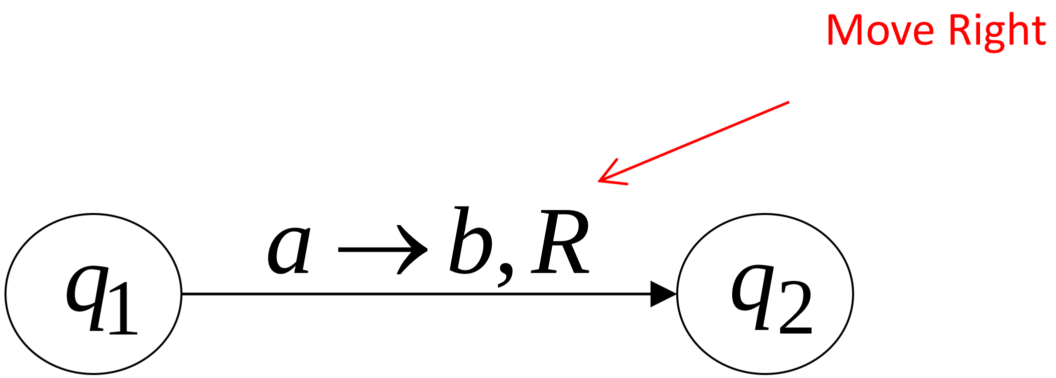
The Input String



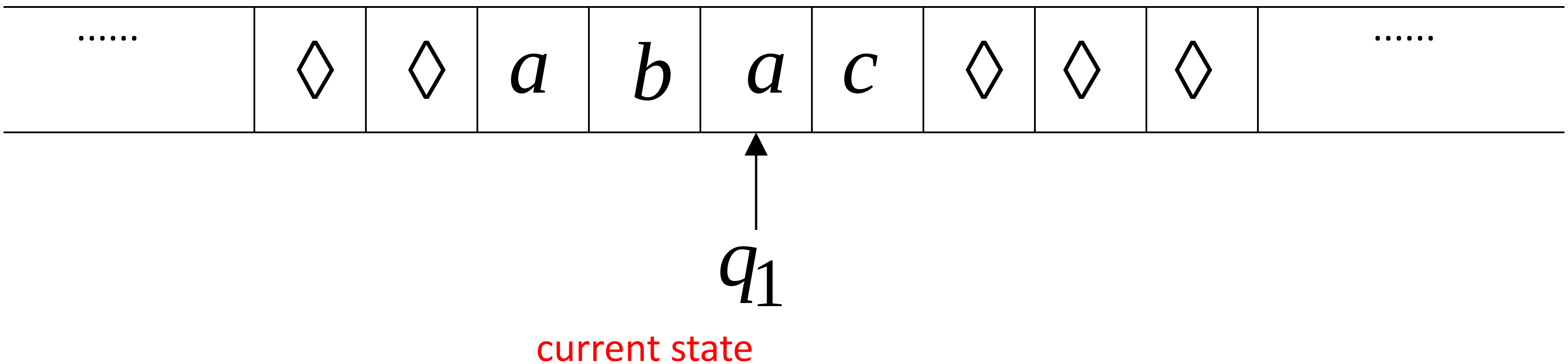
Head starts at the leftmost position
of the input string

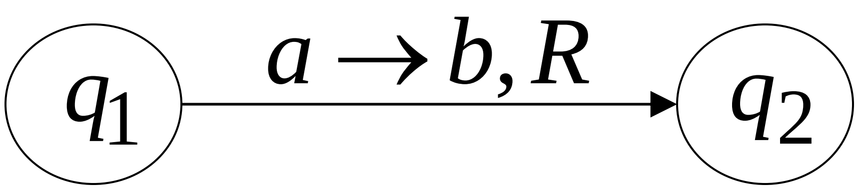
States & Transitions



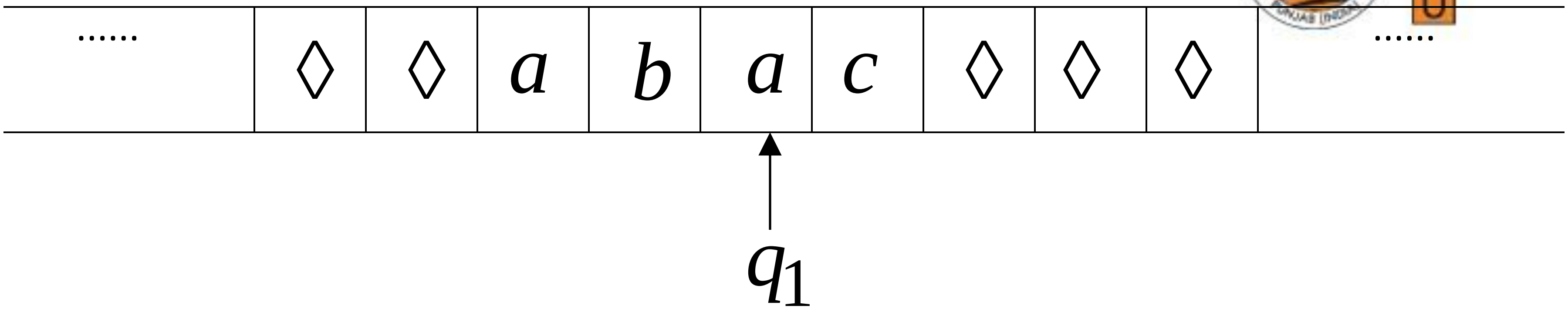


Time 1

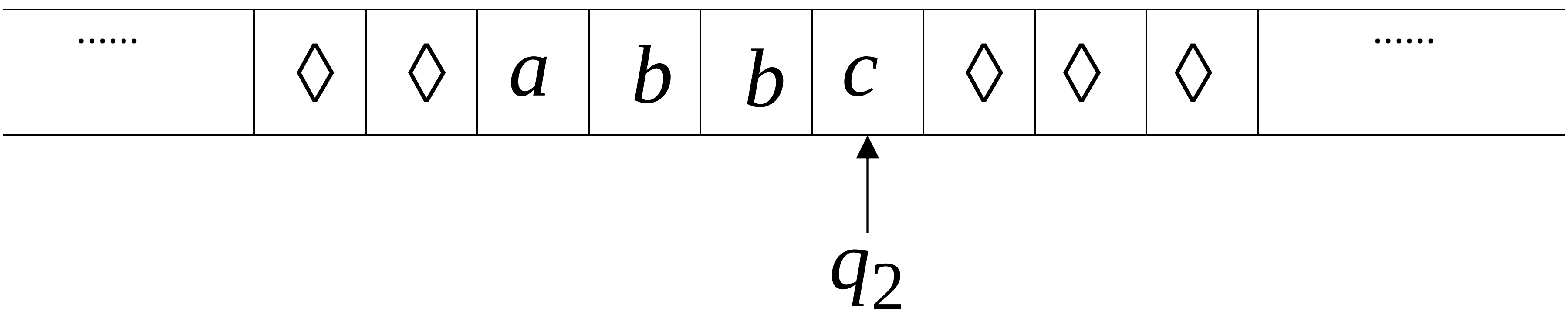


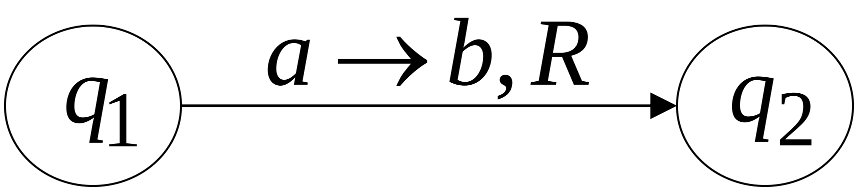


Time 1

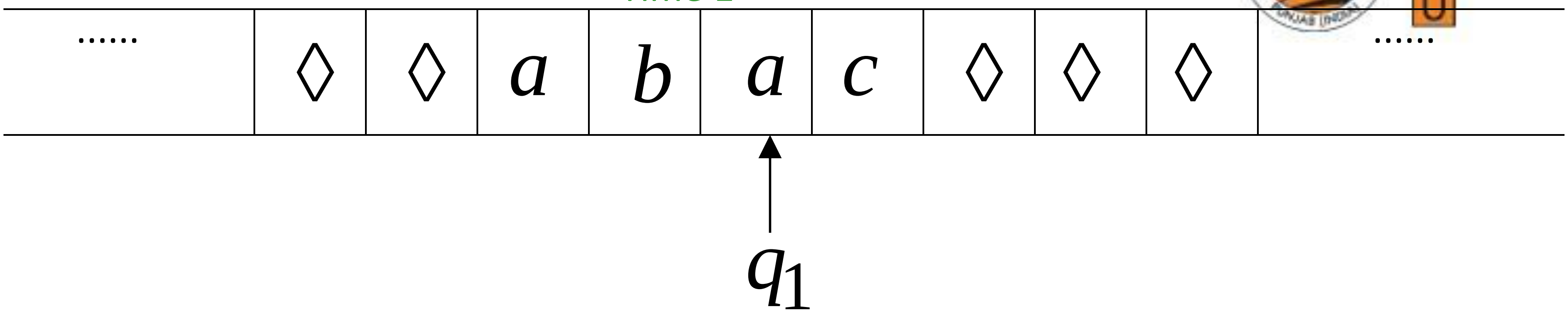


Time 2

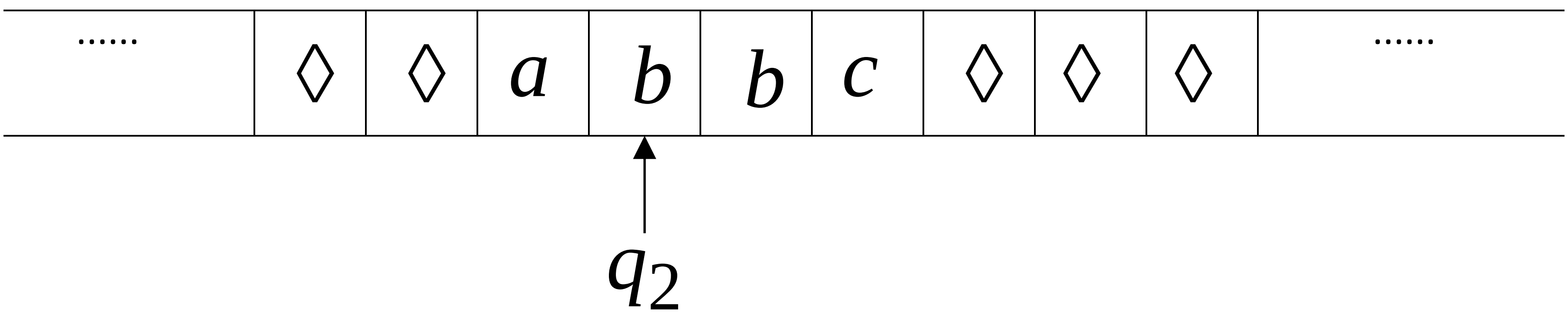


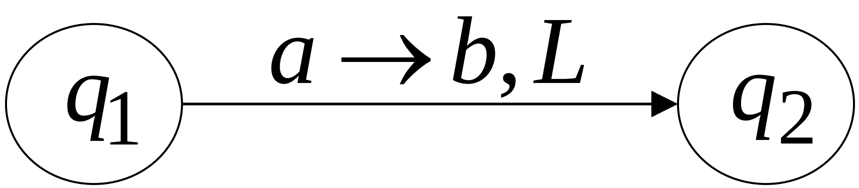


Time 1

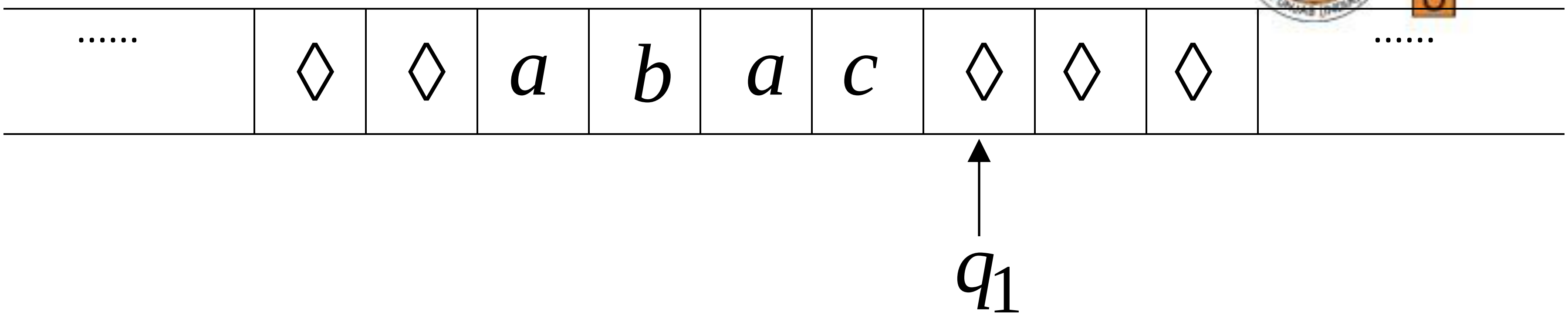


Time 2

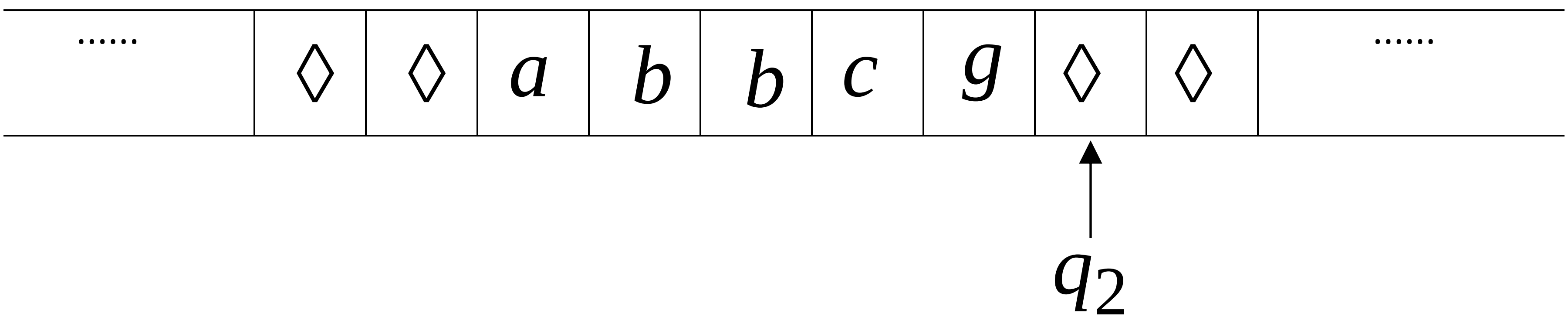


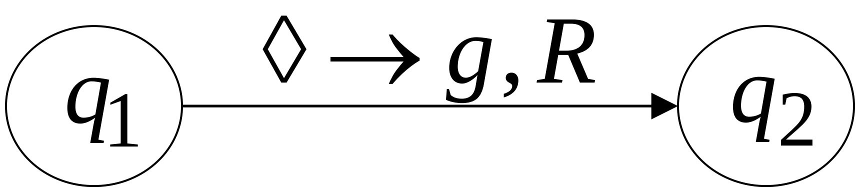


Time 1



Time 2

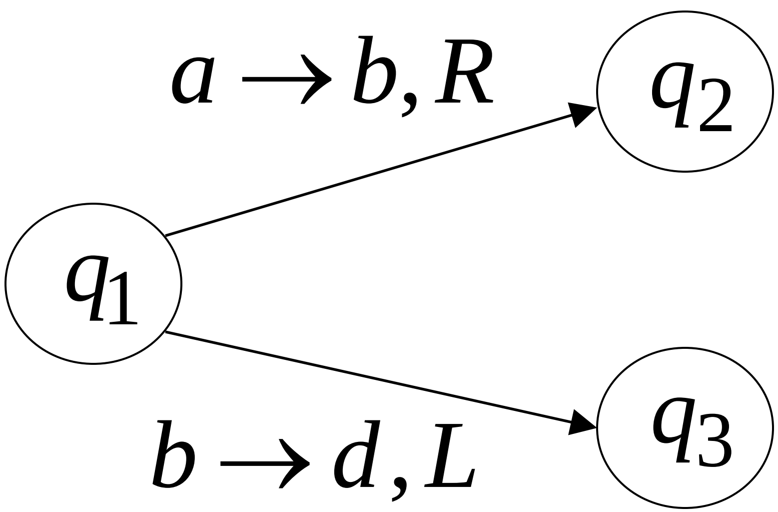




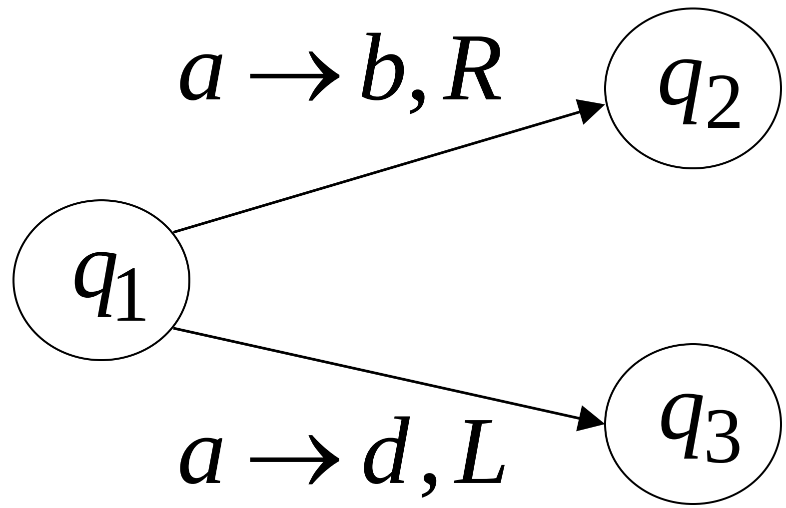
Determinism

Turing Machines are deterministic

Allowed

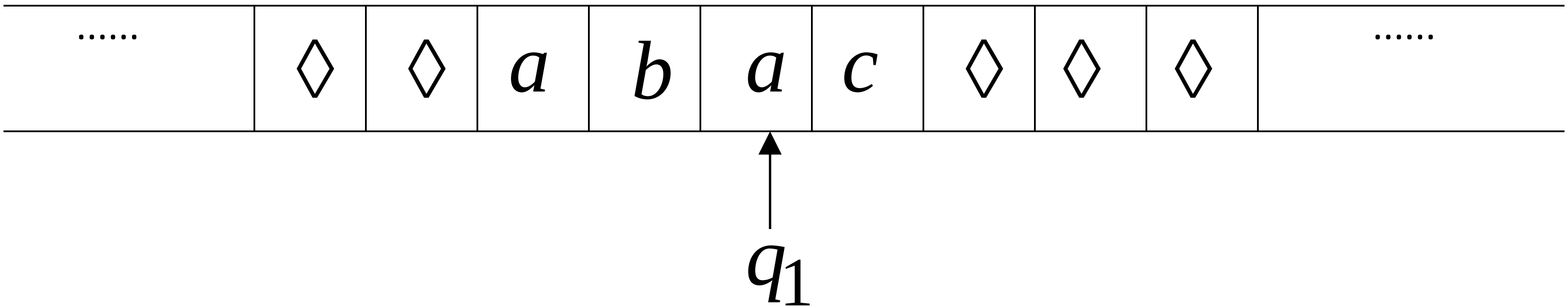


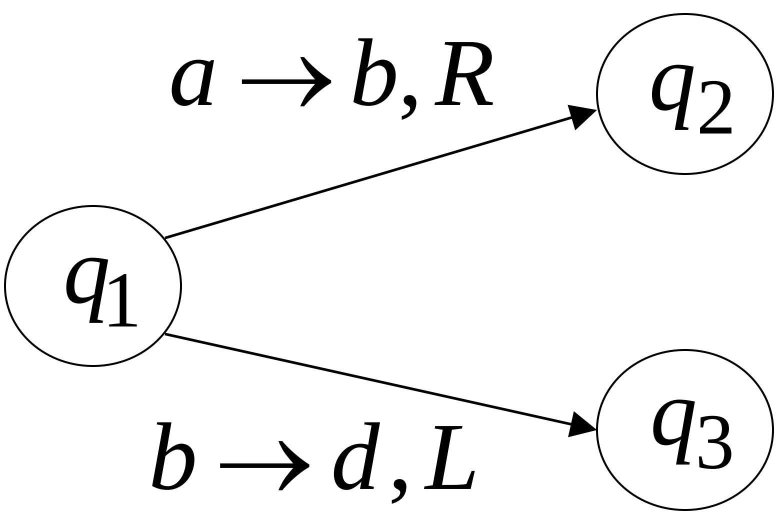
Not Allowed



No lambda transitions allowed

Partial Transition Function





Allowed:

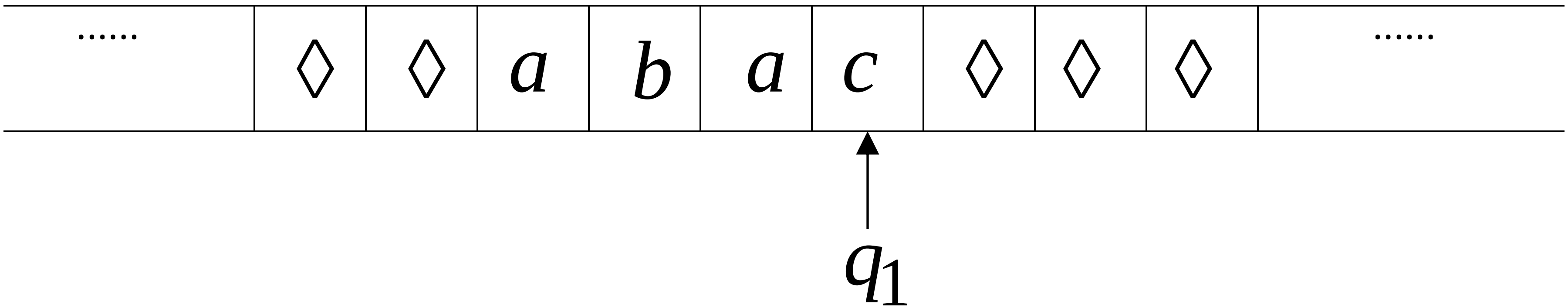
No transition
for input symbol

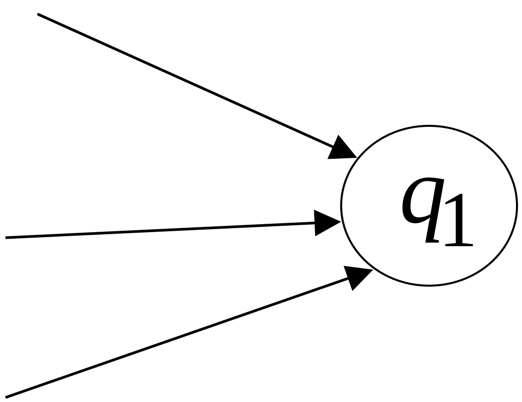
c

Halting

The machine ***halts*** in a state if there is no transition to follow

Halting Example 1:

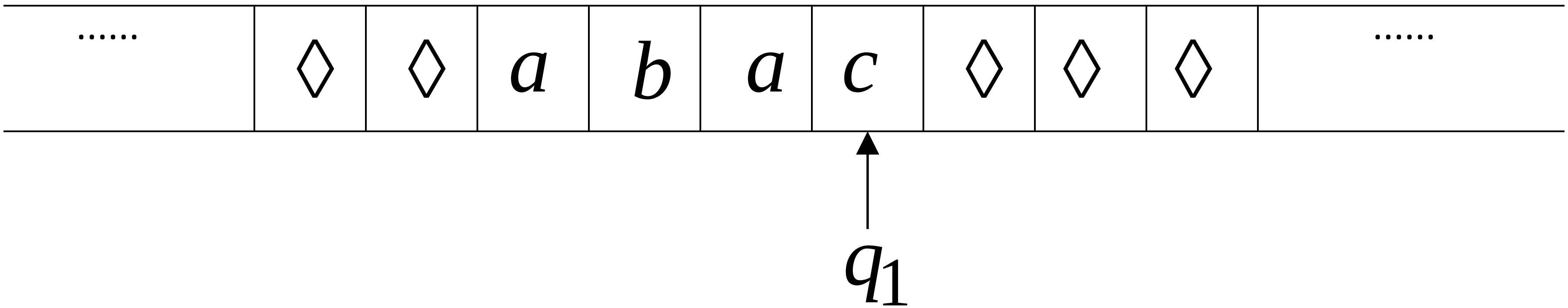


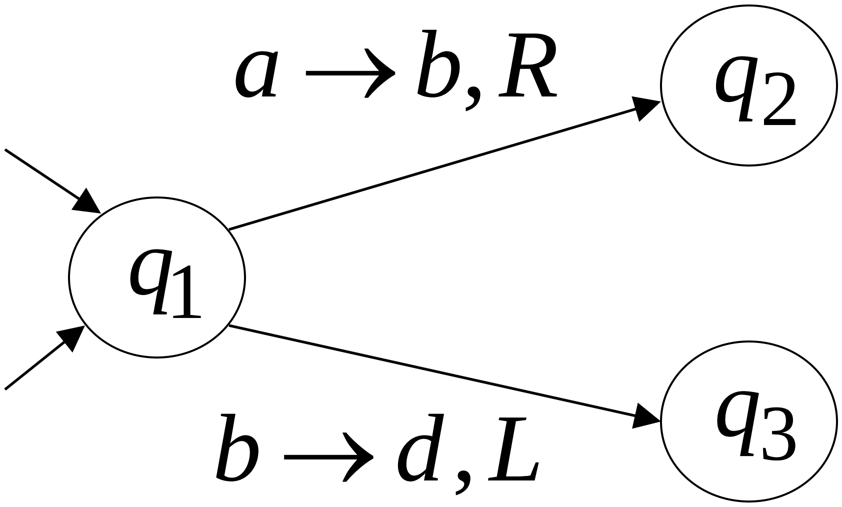


No transition from q_1

HALT!!!

Halting Example 2:

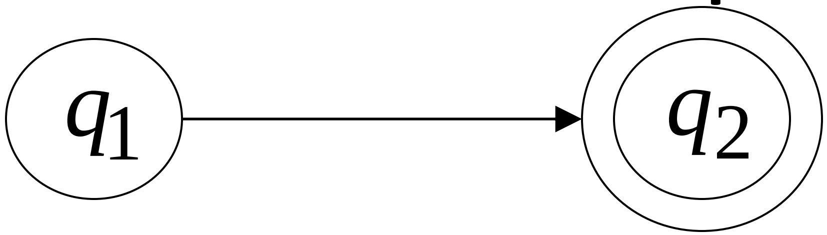




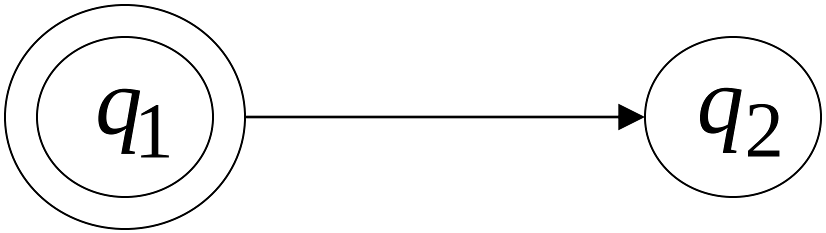
No possible transition from q_1 and symbol c

HALT!!!

Accepting States



Allowed



Not Allowed

- Accepting states have no outgoing transitions
- The machine halts and accepts

Acceptance

Accept Input

string

If machine halts
in an accept state

Reject Input

string

If machine halts
in a non-accept state
or
If machine enters
an *infinite loop*

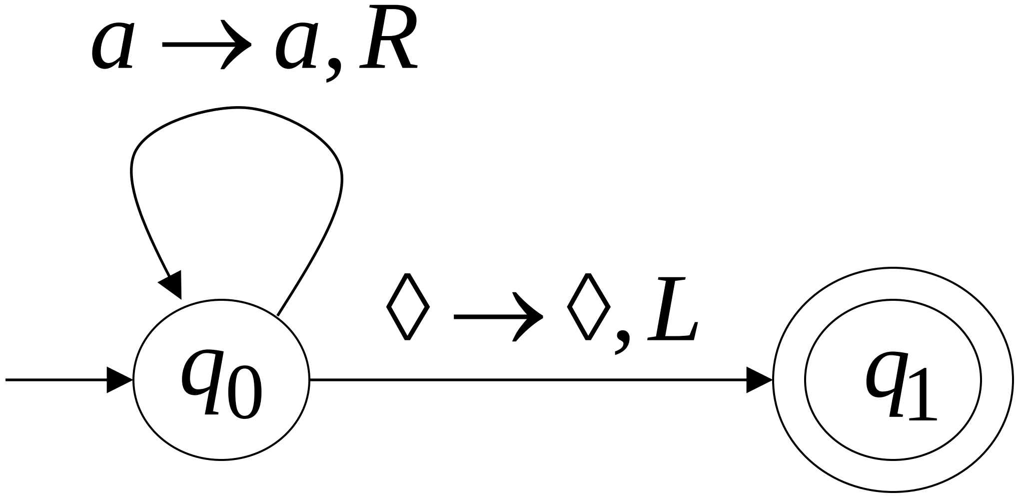
Observation:

In order to accept an input string, it is not necessary to scan all the symbols in the string

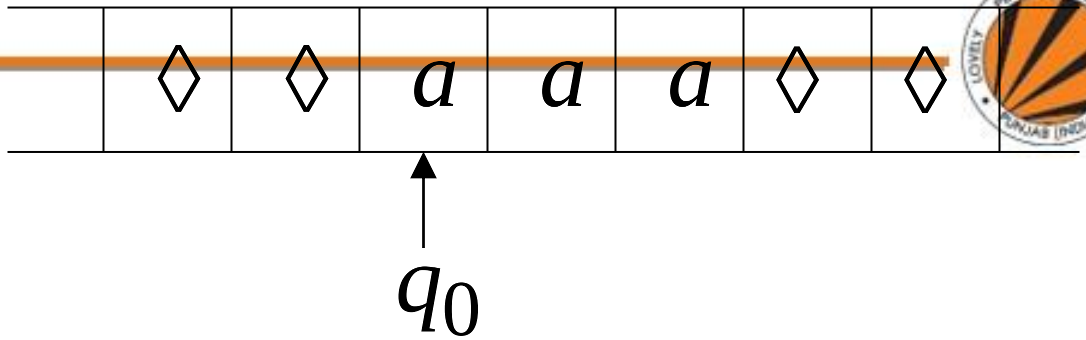
Turing Machine Example

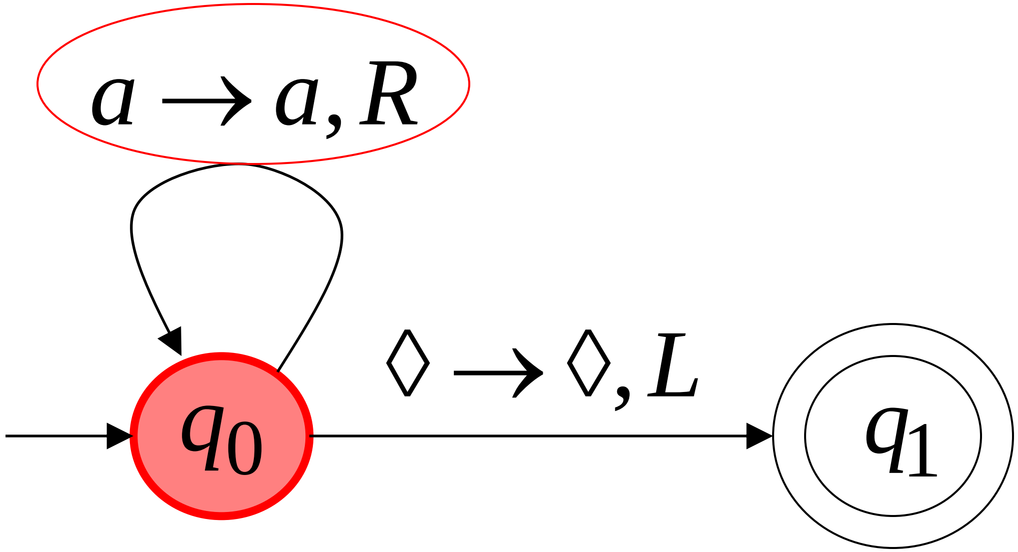
Input alphabet $\Sigma = \{a, b\}$

Accepts the language: a^*

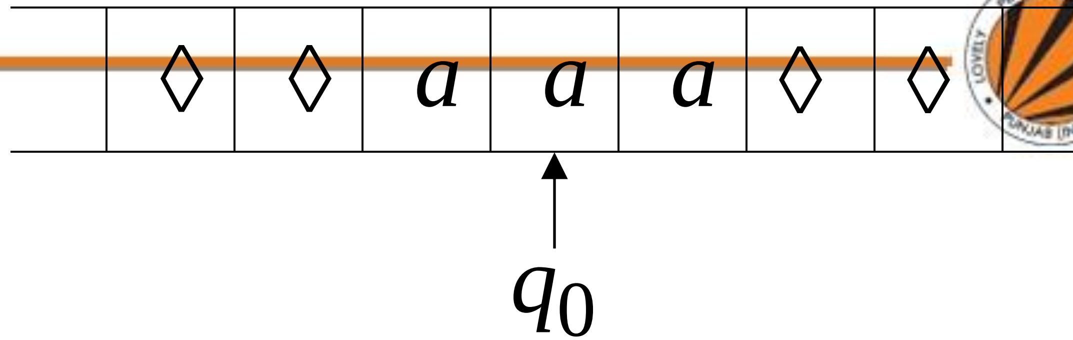


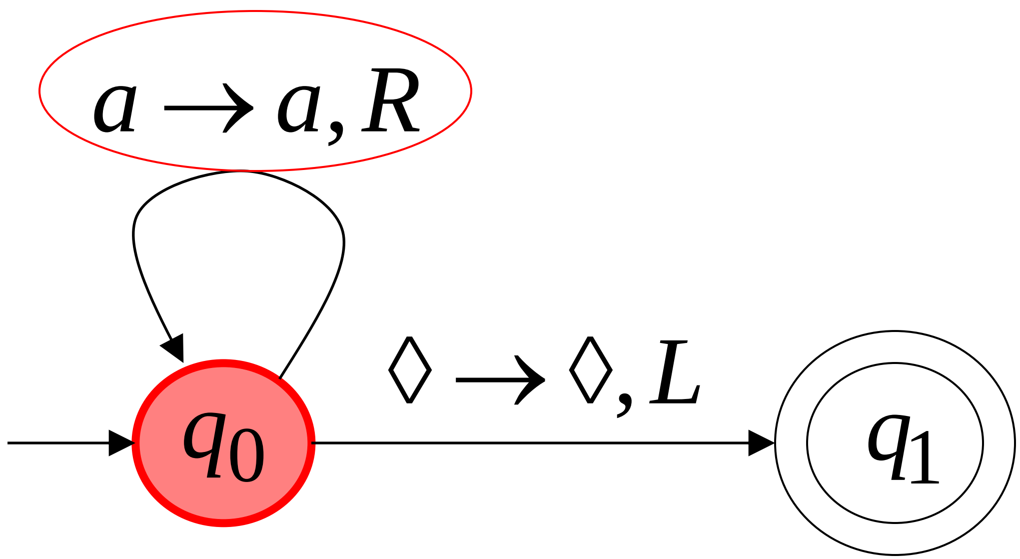
Time 0



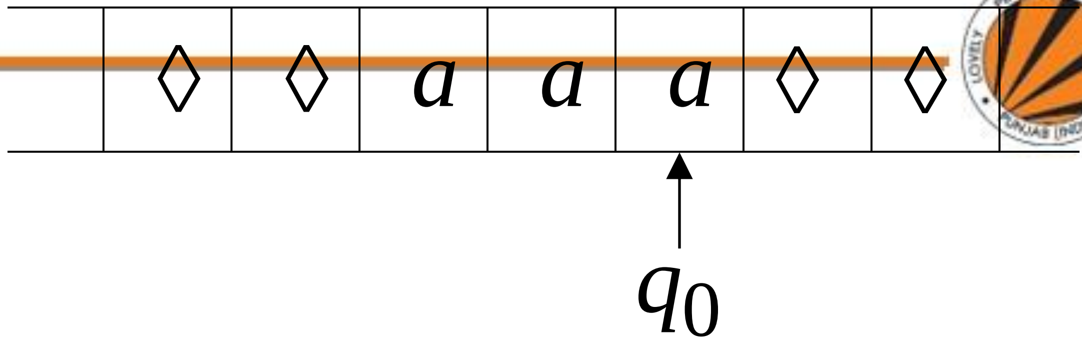


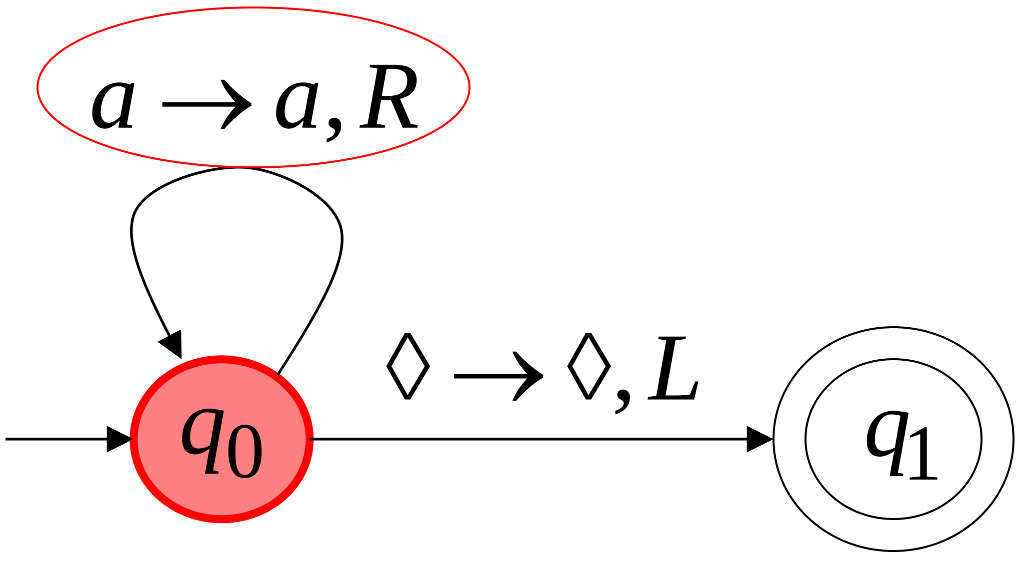
Time 1



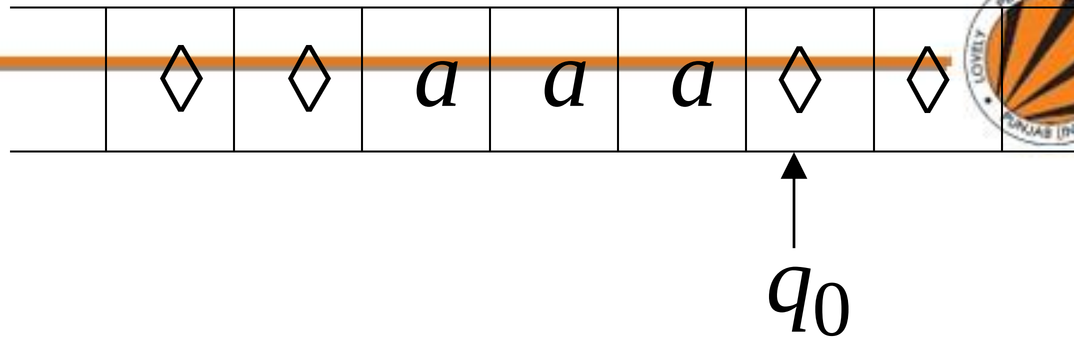


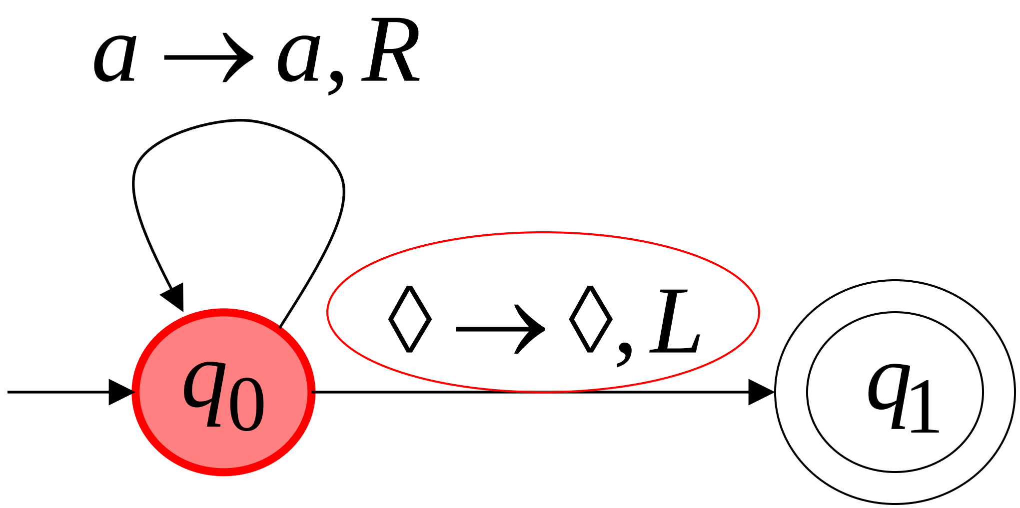
Time 2



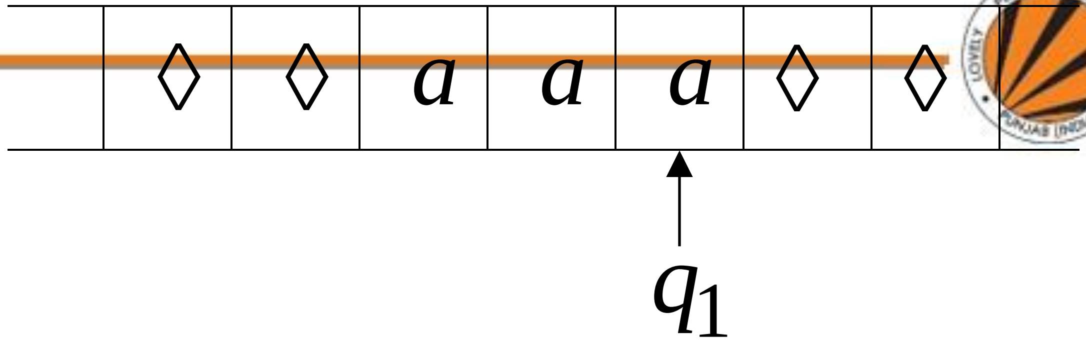


Time 3

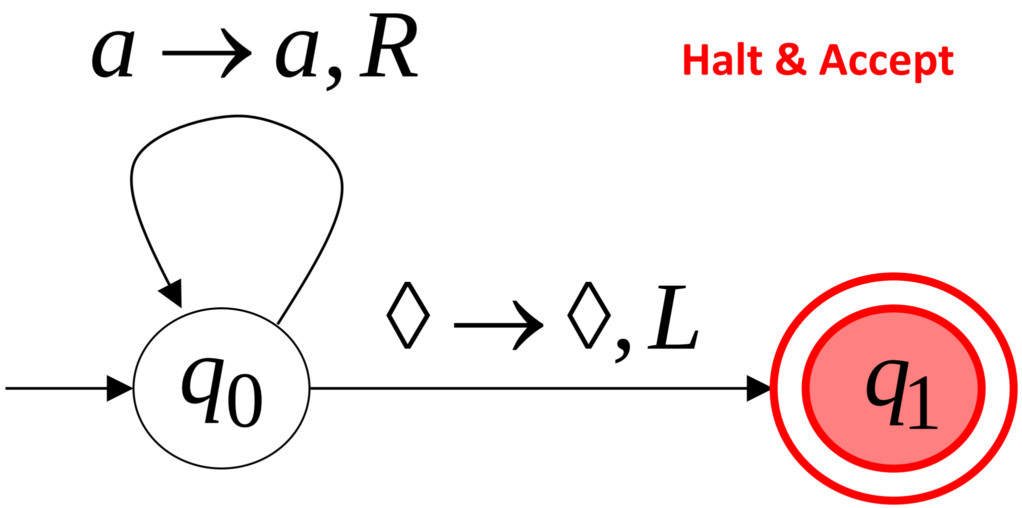




Time 4



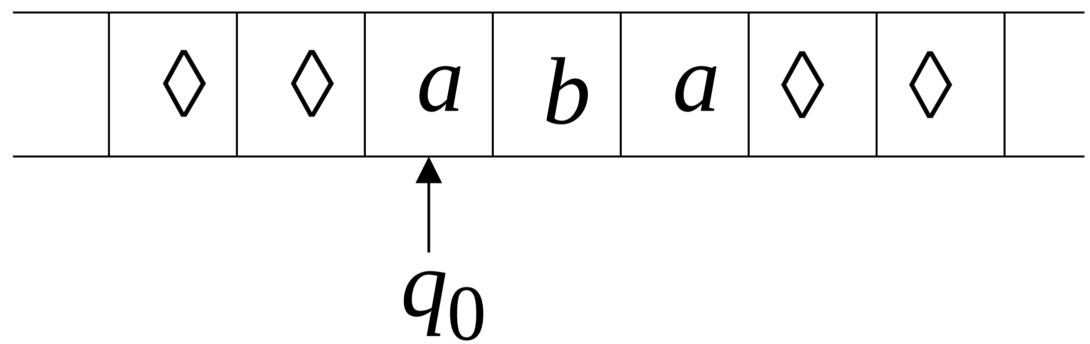
◊
◊
a
a
a
◊
◊
q_1

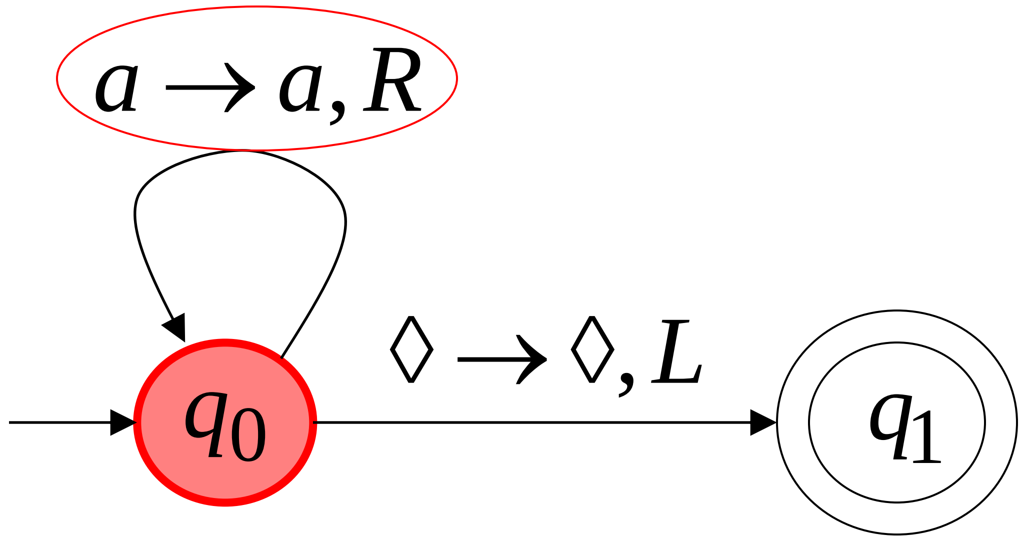


$a \rightarrow a, R$
Halt & Accept
$\Diamond \rightarrow \Diamond, L$
q_0
q_1

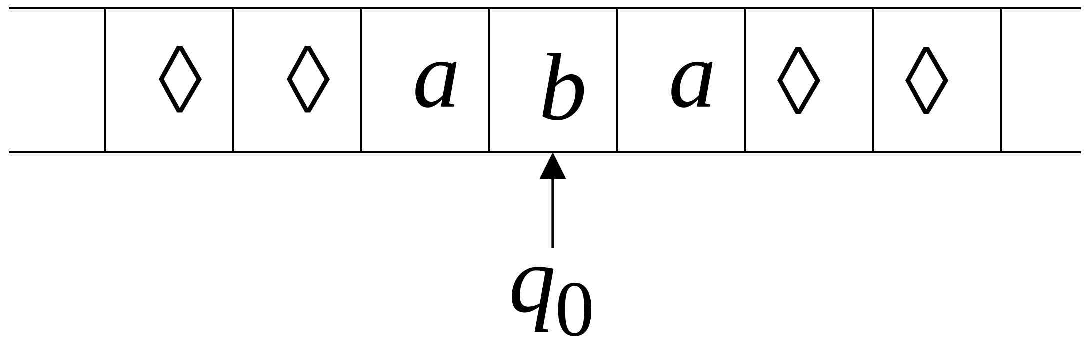
Rejection Example

Time 0



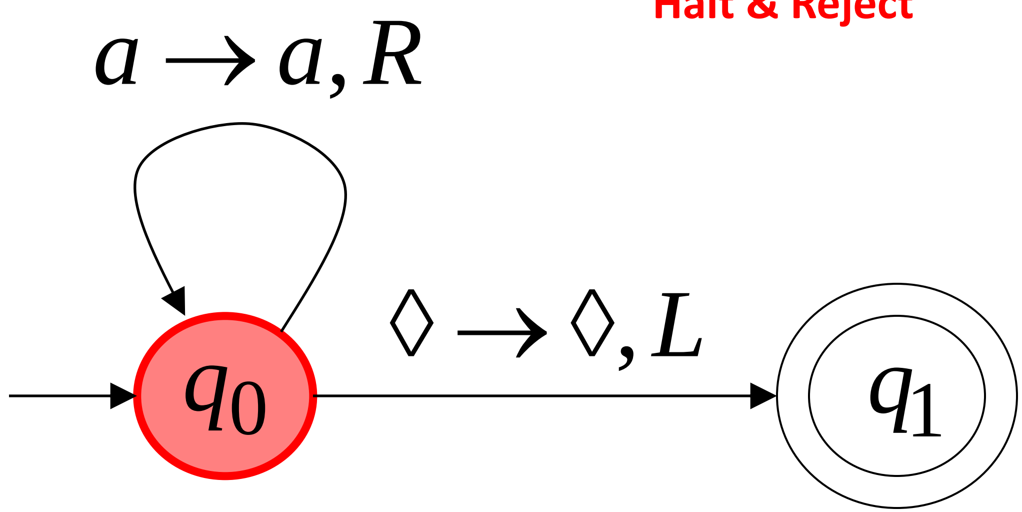


Time 1



No possible Transition

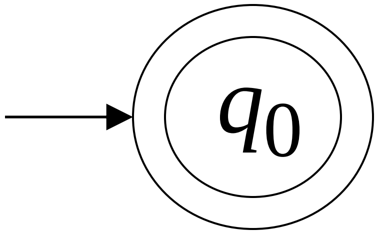
Halt & Reject



A simpler machine for same language

but for input alphabet $\Sigma = \{a\}$

Accepts the language: a^*



Formal Definition

Definition A Turing machine M is a 7-tuple, namely $(Q, \Sigma, \Gamma, \delta, q_0, b, F)$, where

1. Q is a finite nonempty set of states,
2. Γ is a finite nonempty set of tape symbols,
3. $b \in \Gamma$ is the blank,
4. Σ is a nonempty set of input symbols and is a subset of Γ and $b \notin \Sigma$,
5. δ is the transition function mapping (q, x) onto (q', y, D) where D denotes the direction of movement of R/W head; $D = L$ or R according as the movement is to the left or right.
6. $q_0 \in Q$ is the initial state, and
7. $F \subseteq Q$ is the set of final states.

Representation of Turing Machine

- By instantaneous description
- By transition table
- By transition diagram

Representation by Instantaneous Description

- ID of PDA has been defined by (q, a, Z)

But the input string to be processed is not sufficient to be defined as the ID of a Turing machine, for the R/W head can move to the left as well. So an ID of a Turing machine is defined in terms of the entire input string and the current state.

Definition An ID of a Turing machine M is a string $\alpha\beta\gamma$, where β is the present state of M, the entire input string is split as $\alpha\gamma$, the first symbol of γ is the current symbol a under the R/W head and γ has all the subsequent symbols of the input string, and the string α is the substring of the input string formed by all the symbols to the left of a.

Moves in a TM

As in the case of pushdown automata, $\delta(q, x)$ induces a change in ID of the Turing machine. We call this change in ID a move.

Suppose $\delta(q, x_i) = (p, y, L)$. The input string to be processed is $x_1x_2 \ldots x_n$, and the present symbol under the R/W head is x_i. So the ID before processing x_i is

$$x_1x_2 \ldots x_{i-1}qx_i \ldots x_n$$

After processing x_i, the resulting ID is

$$x_1 \ldots x_{i-2}\, px_{i-1}yx_{i+1} \ldots x_n$$

This change of ID is represented by

$$x_1x_2 \ldots x_{i-1}\, q\, x_i \ldots x_n \vdash x_i \ldots x_{i-2}\, px_{i-1}\, yx_{i+1} \ldots x_n$$

If $i = 1$, the resulting ID is $p\, y\, x_2x_3 \ldots x_n$.

If $\delta(q, x_i) = (p, y, R)$, then the change of ID is represented by

$$x_1x_2 \ldots x_{i-1}q\, x_i \ldots x_n \vdash x_1x_2 \ldots x_{i-1}\, y\, px_{i+1} \ldots x_n$$

If $i = n$, the resulting ID is $x_1x_2 \ldots x_{n-1}\, y\, p\, b$.

Representation by transition table

Definition of delta is given by transition table. If

$\delta(q, a) = (\gamma, \alpha, \beta)$, we write $\alpha\beta\gamma$ under the α-column and in the q-row.

$\alpha\beta\gamma$ in the table, it means that α is written in the current cell, β gives

the movement of the head (L or R) and γ denotes the new state into which the Turing machine enters.

Consider, for example, a Turing machine with five states $q_1, \ldots, q_5$, where q_1 is the initial state and q_5 is the (only) final state. The tape symbols are 0, 1 and b. The transition table given in Table 9.1 describes δ.

TABLE Transition Table of a Turing Machine

Present state	*Tape symbol*		
	b	0	1
$\rightarrow q_1$	$1Lq_2$	$0Rq_1$	
q_2	bRq_3	$0Lq_2$	$1Lq_2$
q_3		bRq_4	bRq_5
q_4	$0Rq_5$	$0Rq_4$	$1Rq_4$
q_5	$0Lq_2$		

EXAMPLE

Consider the TM description given in Table 9.1. Draw the computation sequence of the input string 00.

Solution

We describe the computation sequence in terms of the contents of the tape and the current state. If the string in the tape is $a_1a_2 \ldots a_j a_{j+1} \ldots a_m$ and the TM in state q is to read a_{j+1}, then we write $a_1a_2 \ldots a_j q\, a_{j+1} \ldots a_m$.

For the input string $00b$, we get the following sequence:

$$q_100b \vdash 0q_10b \vdash 00q_1b \vdash 0q_201 \vdash q_2001$$
$$\vdash q_2b001 \vdash bq_3001 \vdash bbq_401 \vdash bb_0q41 \vdash bb_01q_4b$$
$$\vdash bb010q_5 \vdash bb01q_200 \vdash bb0q_2100 \vdash bbq_20100$$
$$\vdash bq_2b0100 \vdash bbq_30100 \vdash bbbq_4100 \vdash bbb_1q_400$$
$$\vdash bbb10q_40 \vdash bbb100q_4b \vdash bbb1000q_5b$$
$$\vdash bbb100q_200 \vdash bbb10q_2000 \vdash bbb1q_20000$$
$$\vdash bbbq_210000 \vdash bbq_2b10000 \vdash bbbq_310000 \vdash bbbbq_50000$$

Representation by Transition diagram

The states are represented by vertices. Directed edges are used to

represent transition of states. The labels are triples of the form (α, β, γ), where $\alpha, \beta \in \Gamma$ and $\gamma \in \{L, R\}$. When there is a directed edge from q_i to q_j with label (α, β, γ), it means that

$$\delta(q_i, \alpha) = (q_j, \beta, \gamma)$$

During the processing of an input string, suppose the Turing machine enters q_i and the R/W head scans the (present) symbol α. As a result, the symbol β is written in the cell under the R/W head. The R/W head moves to the left or to the right, depending on γ, and the new state is q_j.

EXAMPLE

M is a Turing machine represented by the transition system in Fig.   Obtain the computation sequence of *M* for processing the input string 0011.

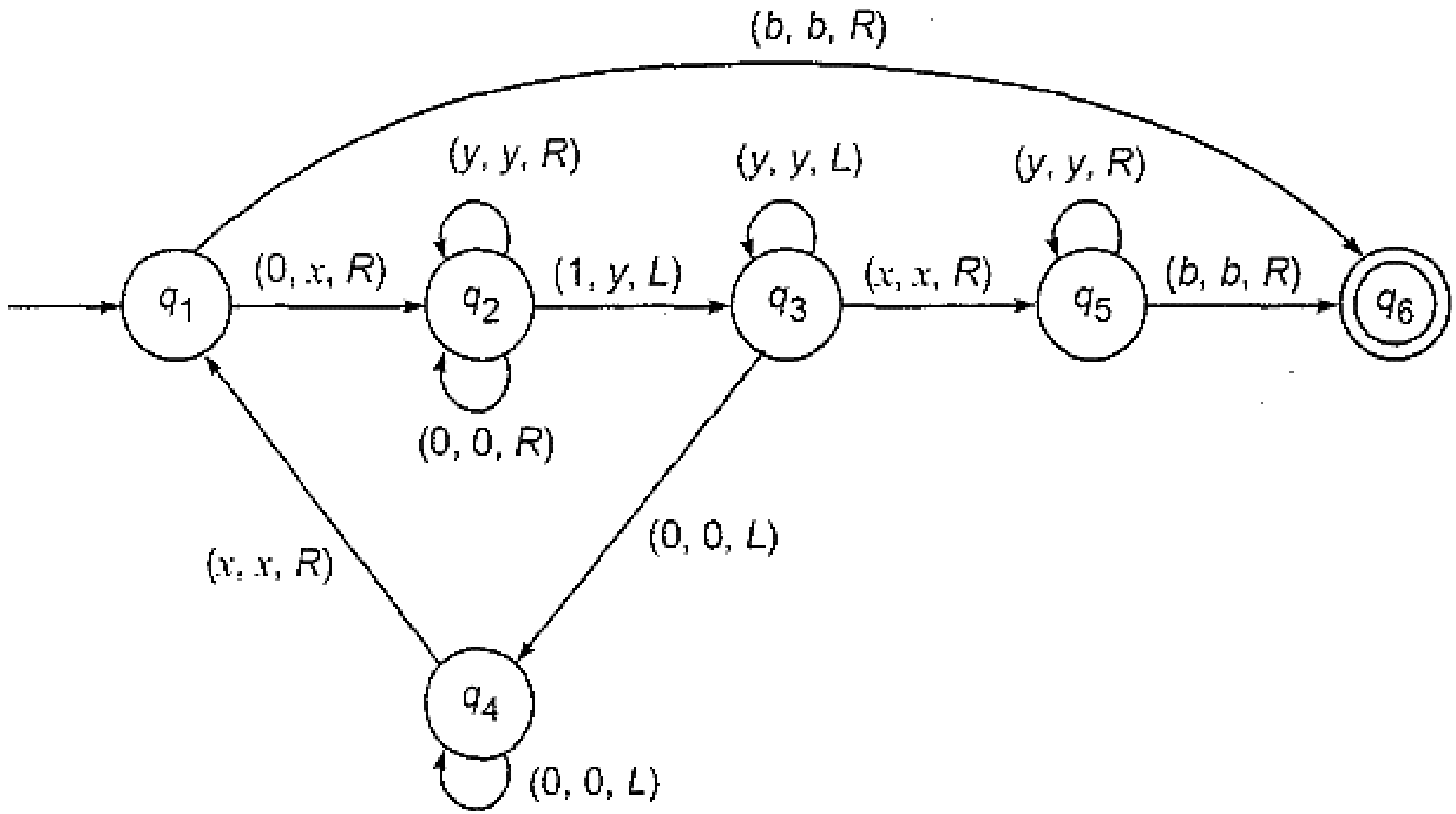


Fig.   Transition system for *M*.

Solution

The entire computation sequence reads as follows:

$$\underset{q_1}{b\overset{\downarrow}{0}011b} \xrightarrow{(0,x,R)} \underset{q_2}{bx\overset{\downarrow}{0}11b} \xrightarrow{(0,0,R)} \underset{q_2}{bx0\overset{\downarrow}{1}1b}$$

$$\xrightarrow{(1,y,L)} \underset{q_3}{bx\overset{\downarrow}{0}y1b} \xrightarrow{(0,0,L)} \underset{q_4}{b\overset{\downarrow}{x}0y1b} \xrightarrow{(x,x,R)} \underset{q_1}{bx\overset{\downarrow}{0}y1b}$$

$$\xrightarrow{(0,x,R)} \underset{q_2}{bxx\overset{\downarrow}{y}1b} \xrightarrow{(y,y,R)} \underset{q_2}{bxxy\overset{\downarrow}{1}b} \xrightarrow{(1,y,L)} \underset{q_3}{bxx\overset{\downarrow}{y}yb}$$

$$\xrightarrow{(y,y,L)} \underset{q_3}{bx\overset{\downarrow}{x}yyb} \xrightarrow{(x,x,R)} \underset{q_5}{bxx\overset{\downarrow}{y}yb} \xrightarrow{(y,y,R)} \underset{q_5}{bxxy\overset{\downarrow}{y}b}$$

$$\xrightarrow{(y,y,R)} \underset{q_5}{bxxyy\overset{\downarrow}{b}} \xrightarrow{(b,b,R)} \underset{q_6}{bxxyyb\overset{\downarrow}{b}}$$