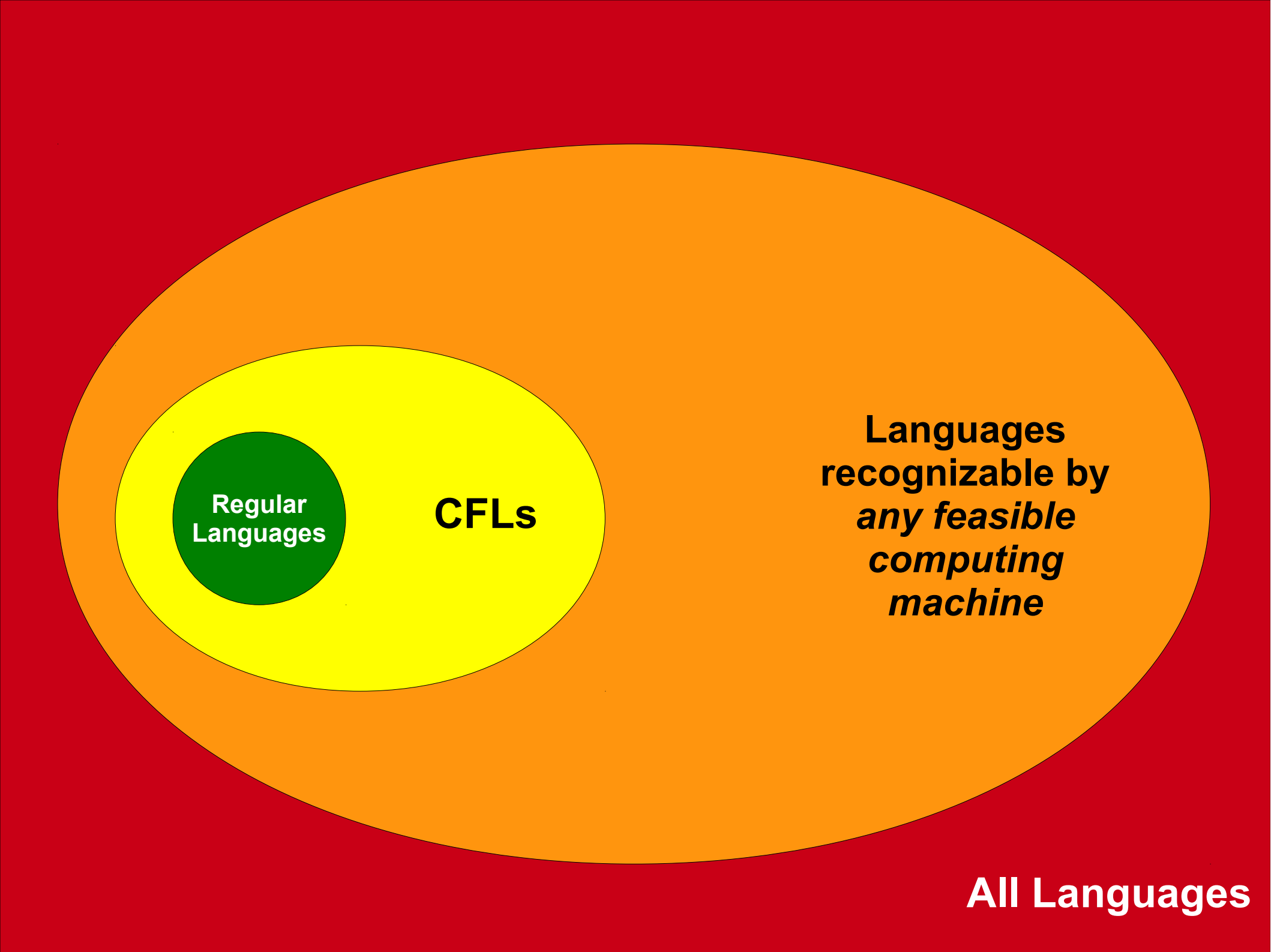
Turing Machines

Part One

Are some problems inherently harder than others?


Regular Languages
CFLs
Languages recognizable by *any feasible computing machine*
All Languages

That same drawing, to scale.

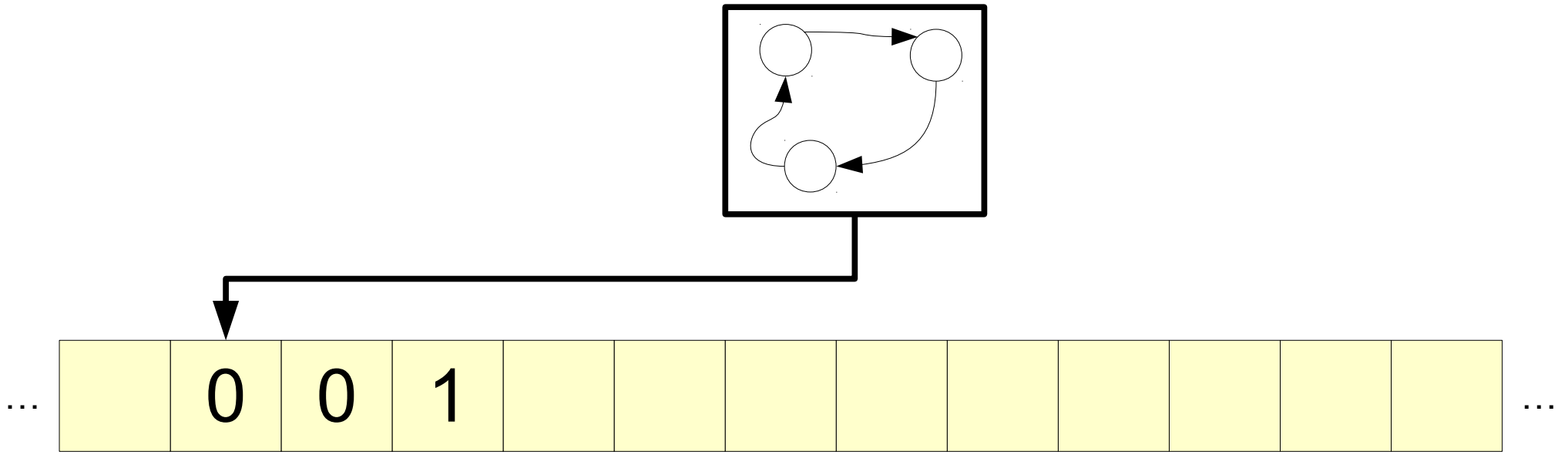
All Languages

The Problem

- Finite automata accept precisely the regular languages.
- We may need unbounded memory to recognize context-free languages.
  - e.g. $\{ \texttt{a}^n\texttt{b}^n \mid n \in \mathbb{N} \}$ requires unbounded counting.
- How do we build an automaton with finitely many states but unbounded memory?

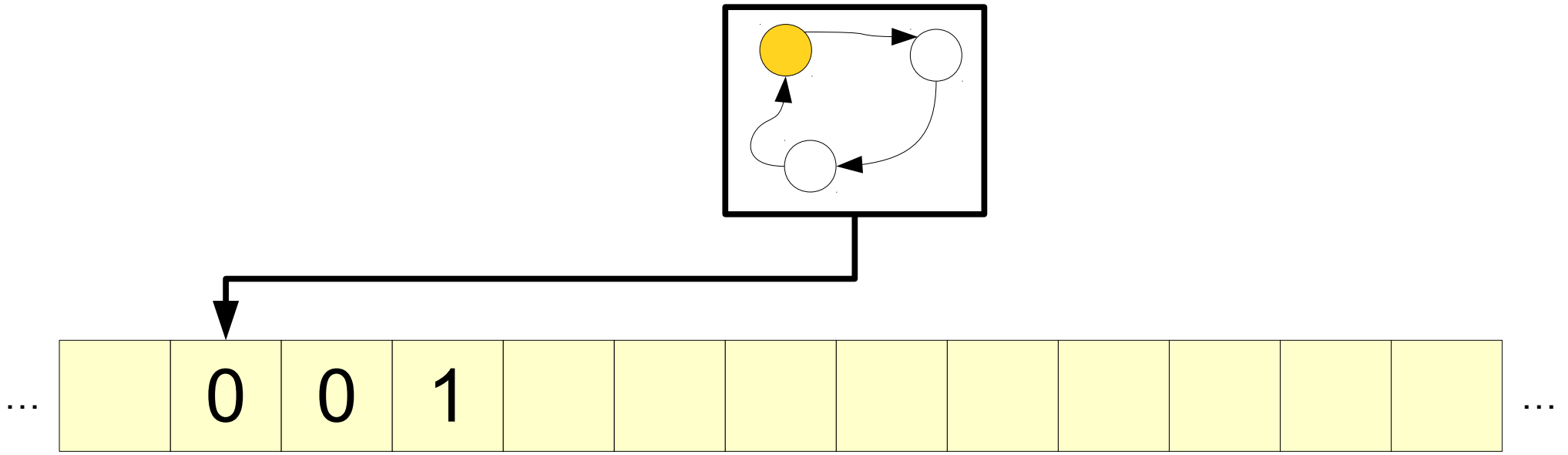
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



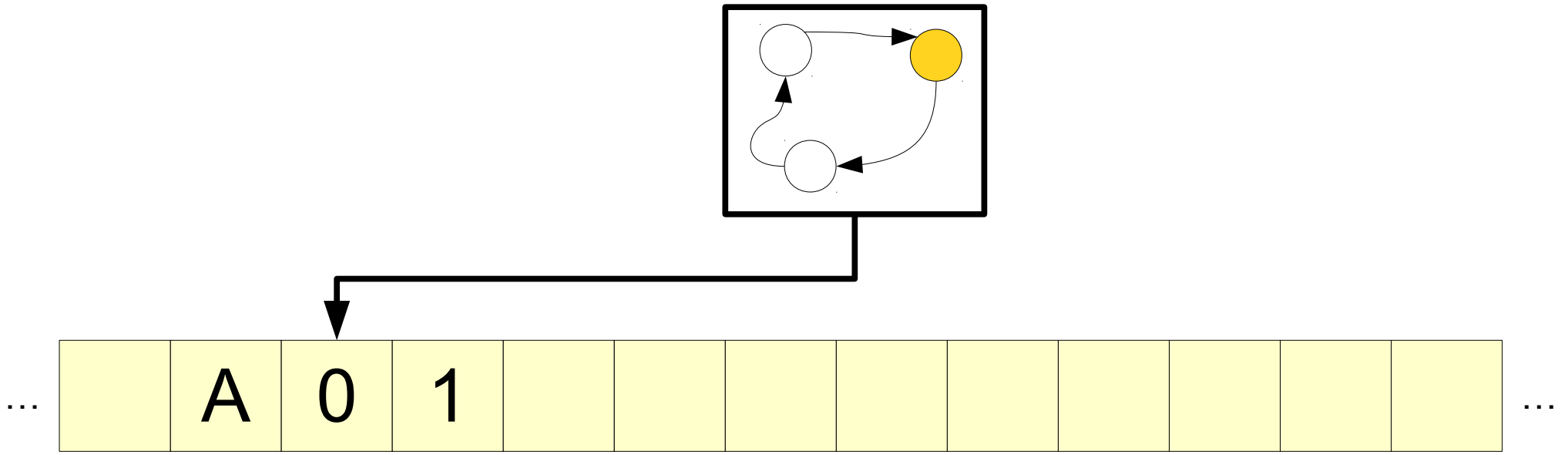
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



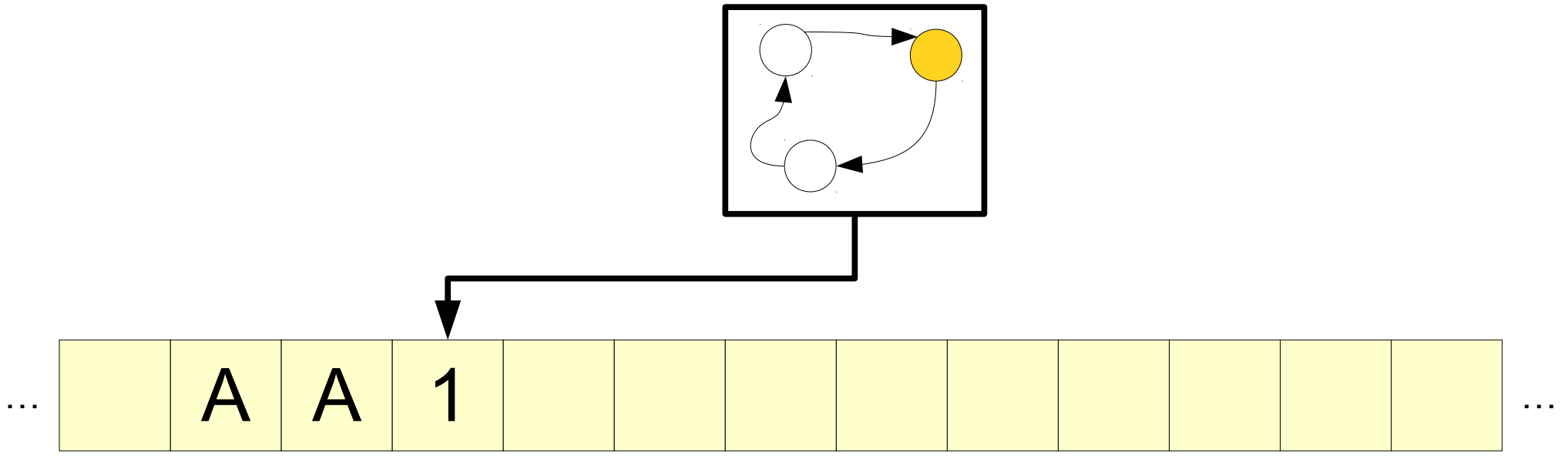
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



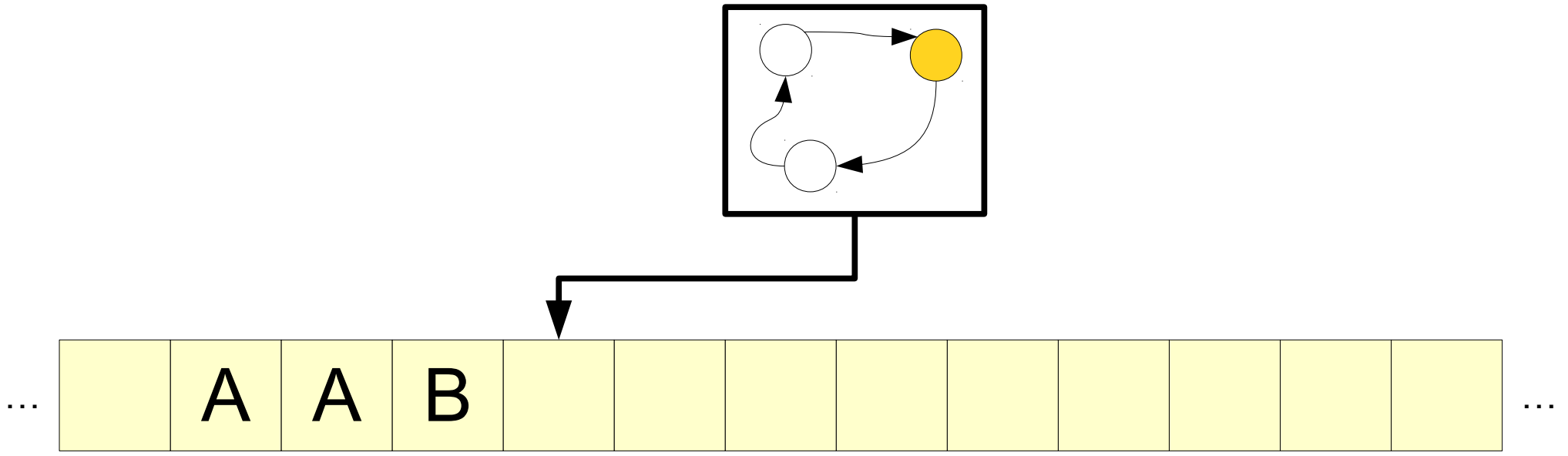
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



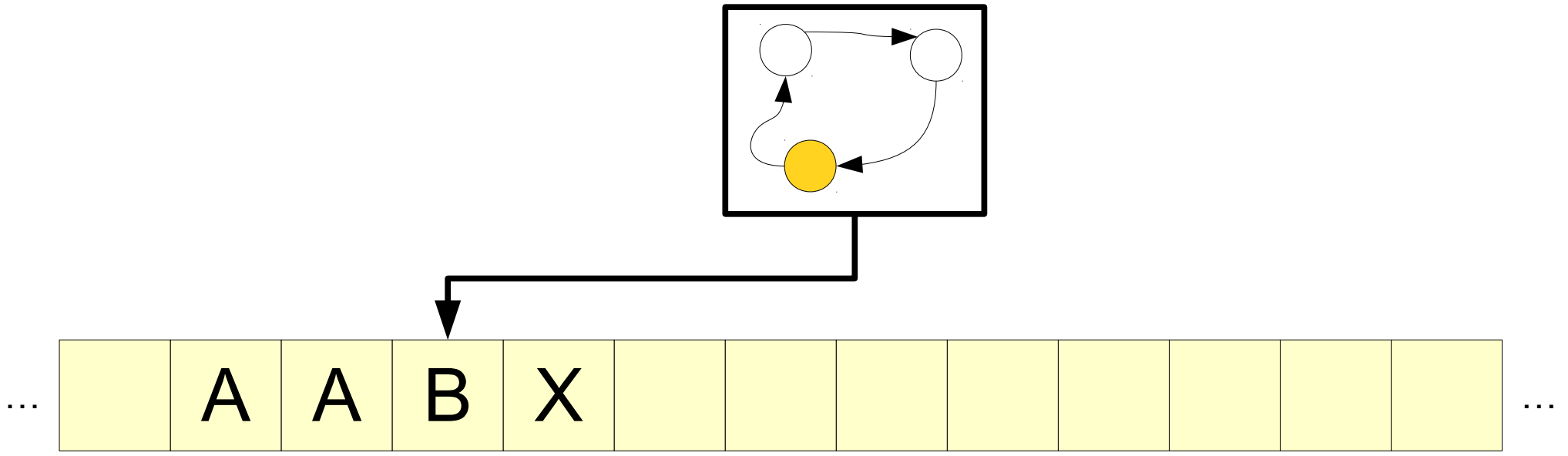
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



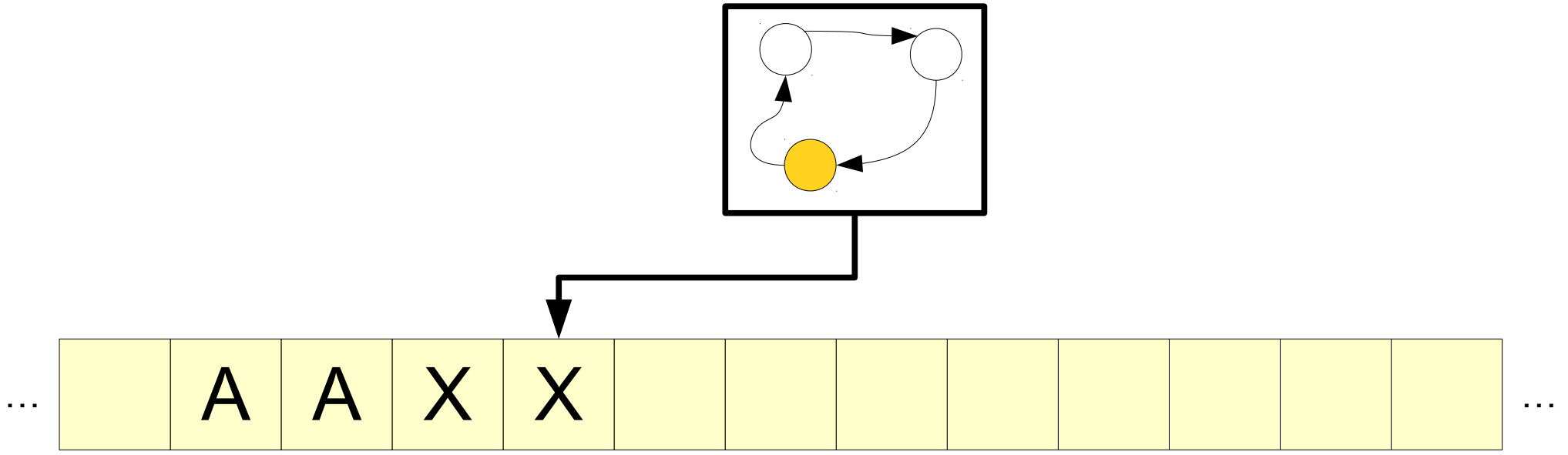
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



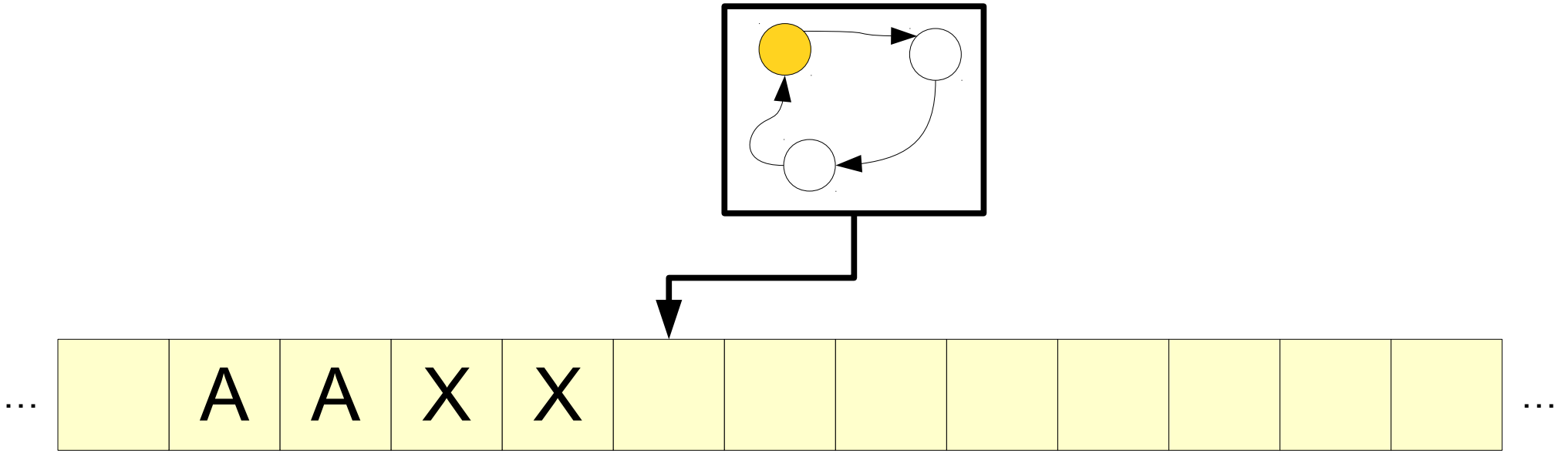
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



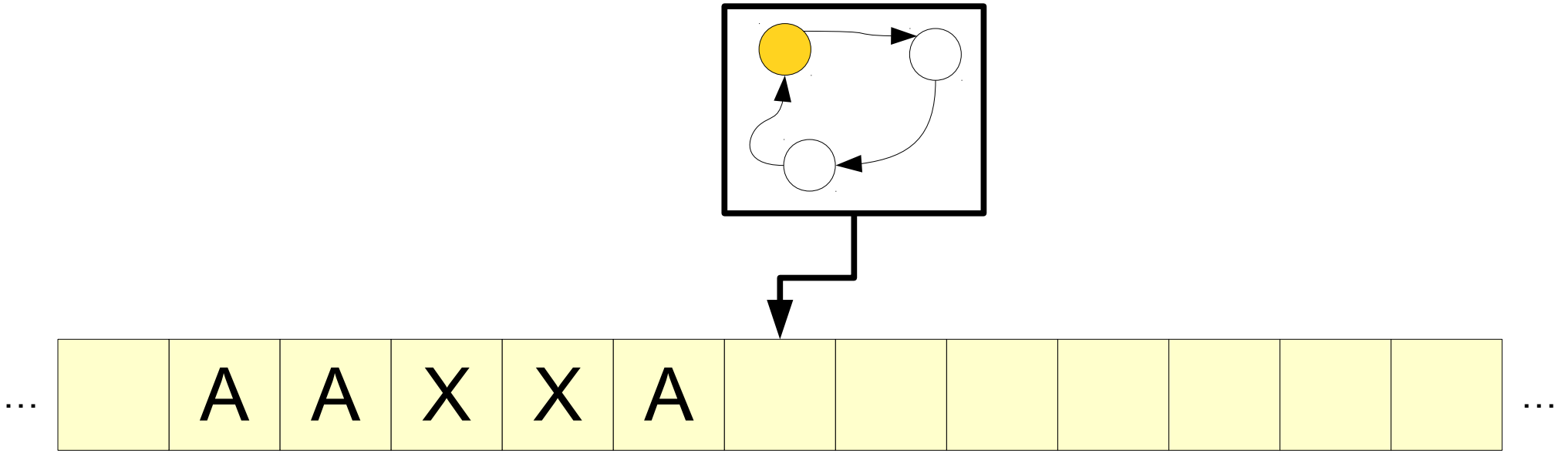
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



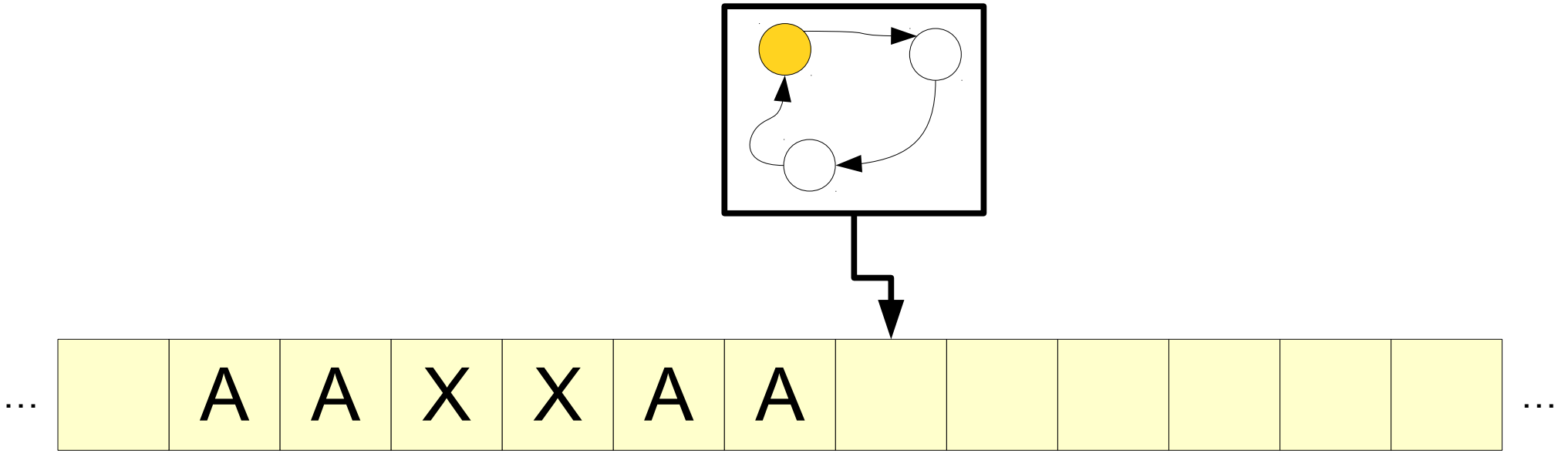
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



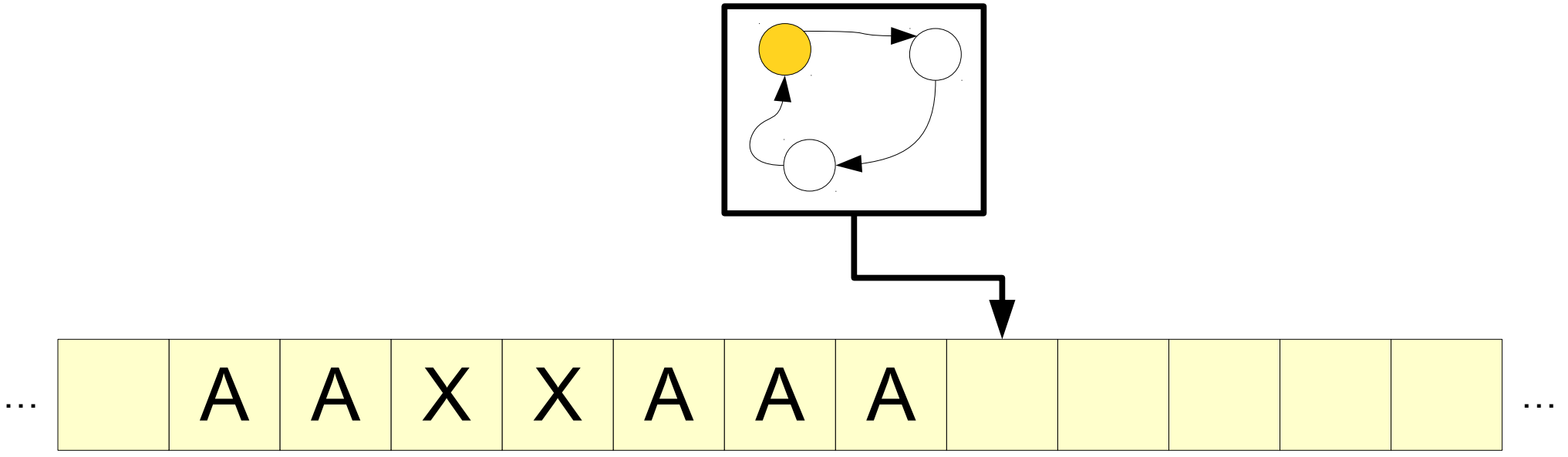
A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



A Better Memory Device

- A **Turing machine** is a finite automaton equipped with an **infinite tape** as its memory.
- The tape begins with the input to the machine written on it, surrounded by infinitely many blank cells.
- The machine has a **tape head** that can read and write a single memory cell at a time.



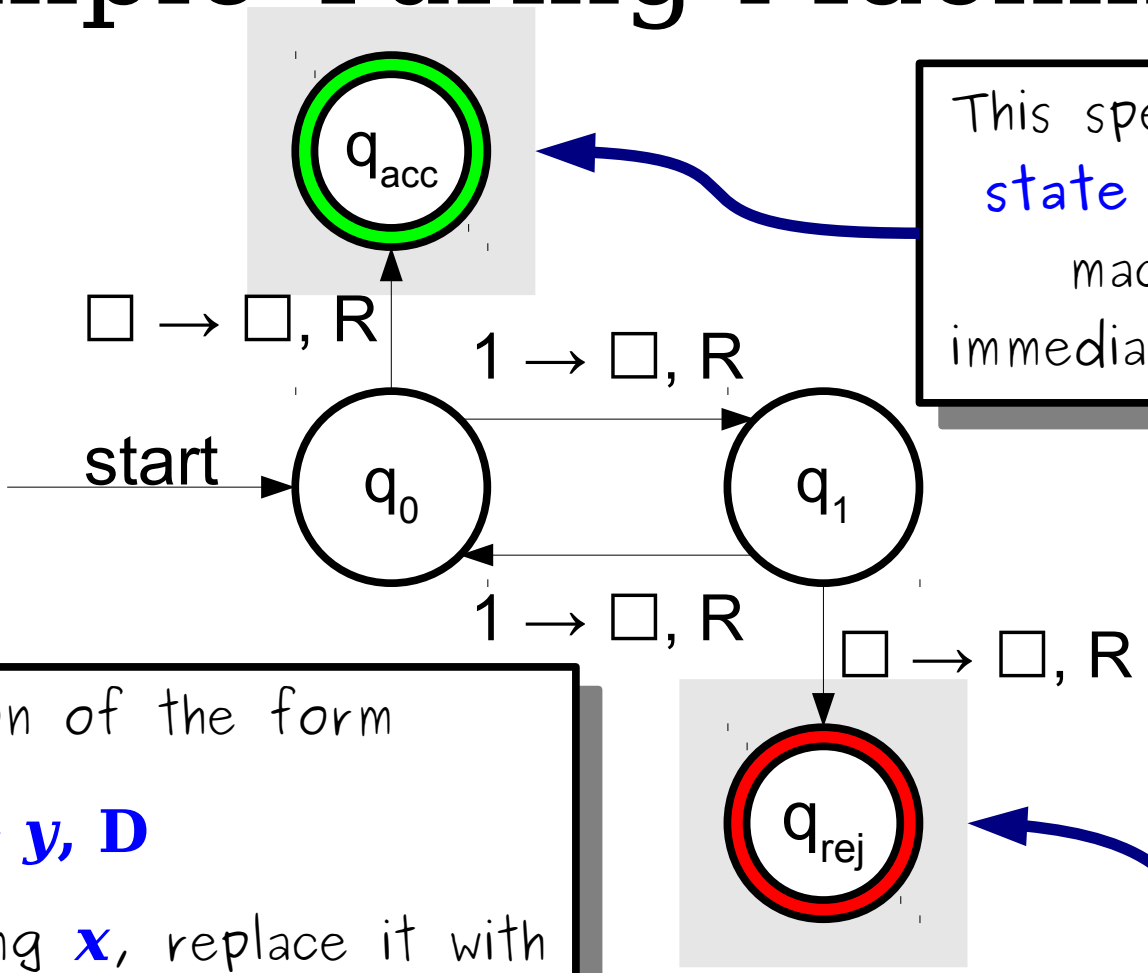
The Turing Machine

- A Turing machine consists of three parts:
  - A **finite-state control** that issues commands,
  - an **infinite tape** for input and scratch space, and
  - a **tape head** that can read and write a single tape cell.
- At each step, the Turing machine
  - writes a symbol to the tape cell under the tape head,
  - changes state, and
  - moves the tape head to the left or to the right.

Input and Tape Alphabets

- A Turing machine has two alphabets:
  - An **input alphabet** Σ. All input strings are written in the input alphabet.
  - A **tape alphabet** Γ, where Σ ⊆ Γ. The tape alphabet contains all symbols that can be written onto the tape.
- The tape alphabet Γ can contain any number of symbols, but always contains at least one **blank symbol**, denoted □. You are guaranteed □ ∉ Σ.
- At startup, the Turing machine begins with an infinite tape of □ symbols with the input written at some location. The tape head is positioned at the start of the input.

A Simple Turing Machine



This special **accept state** causes the machine to immediately accept.

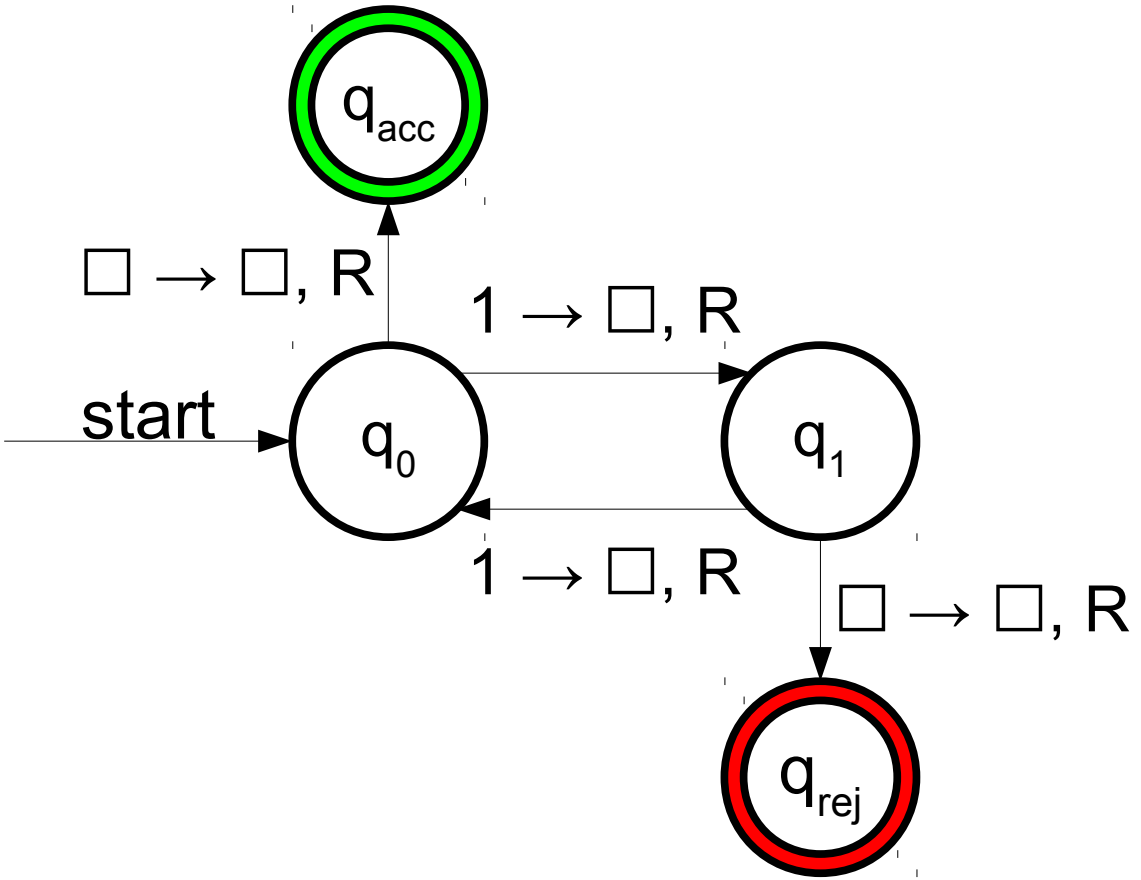
Each transition of the form

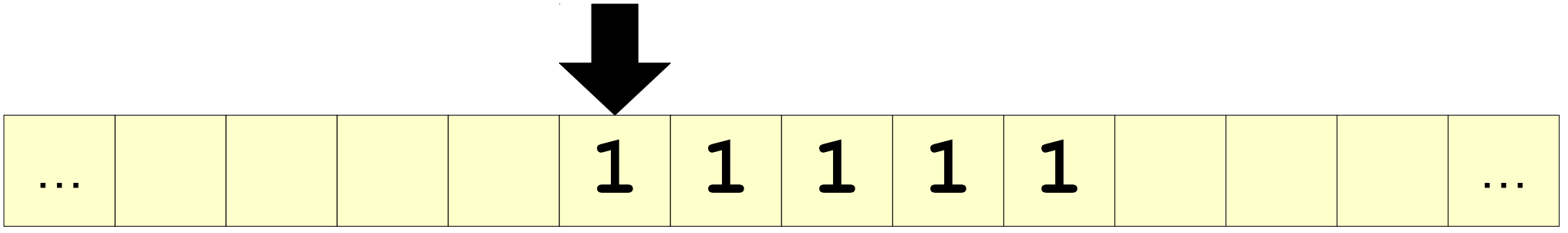
$$x \to y, D$$

means "upon reading x, replace it with symbol y and move the tape head in direction D (which is either **L** or **R**). The symbol □ represents the **blank symbol**.

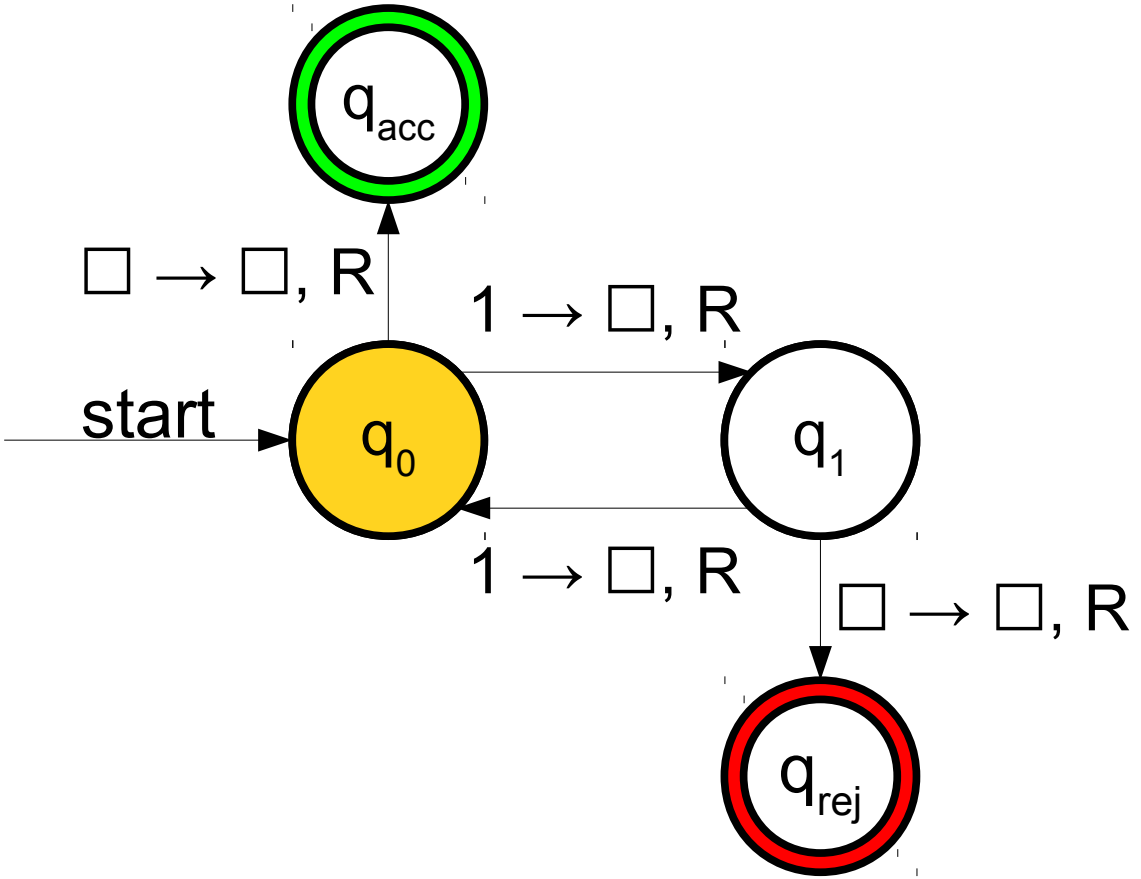
This special **reject state** causes the machine to immediately reject.

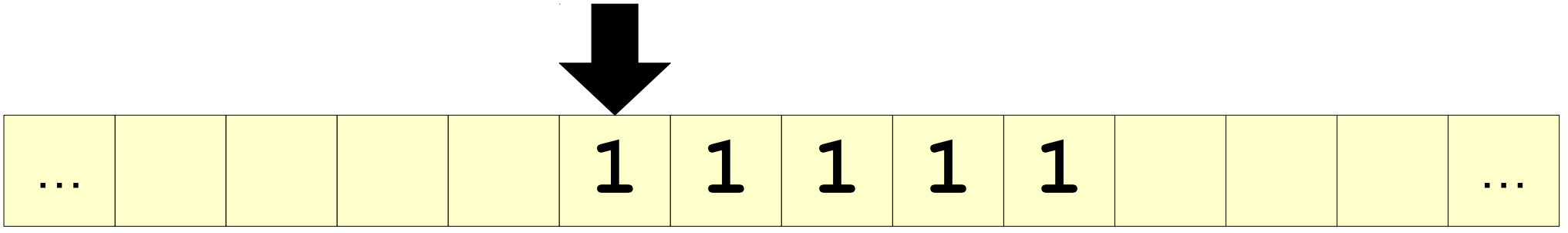
A Simple Turing Machine



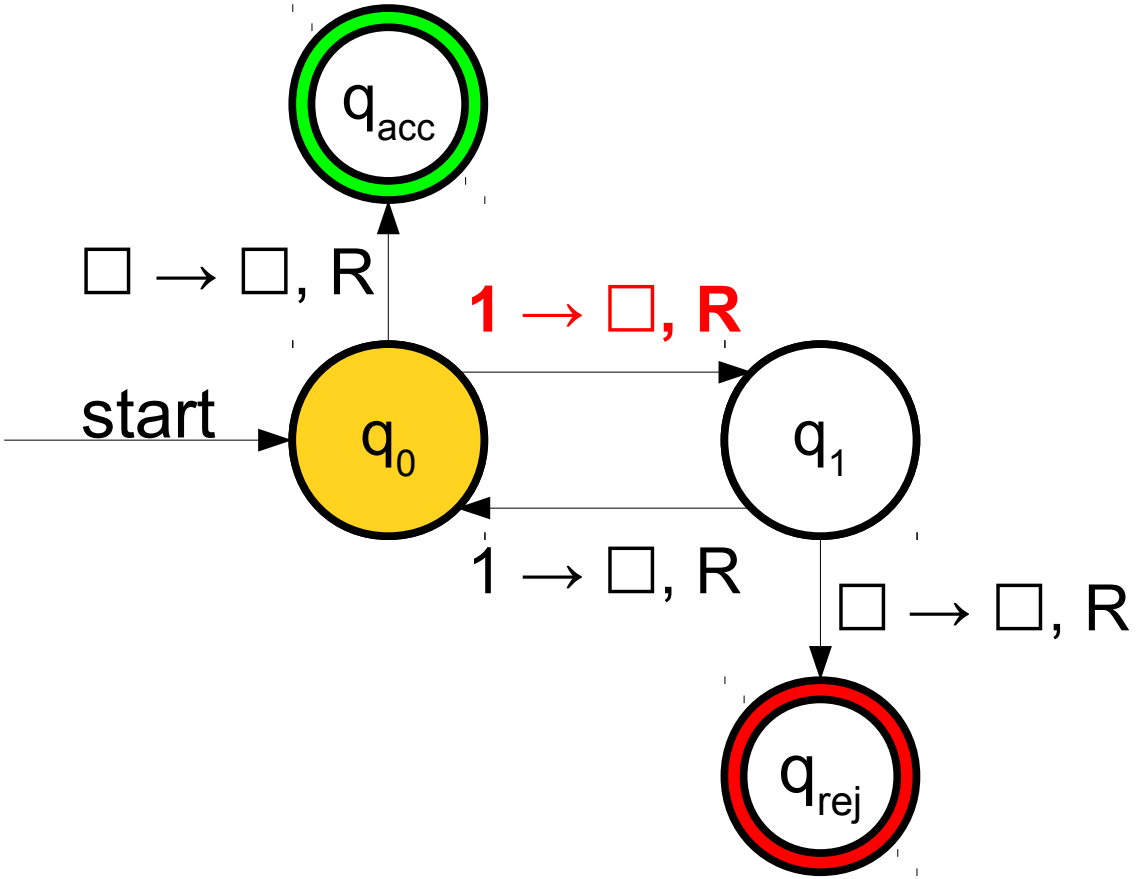


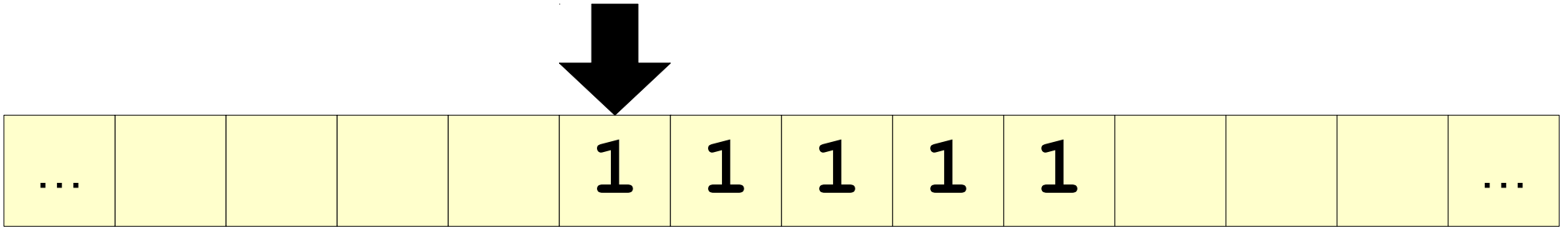
A Simple Turing Machine



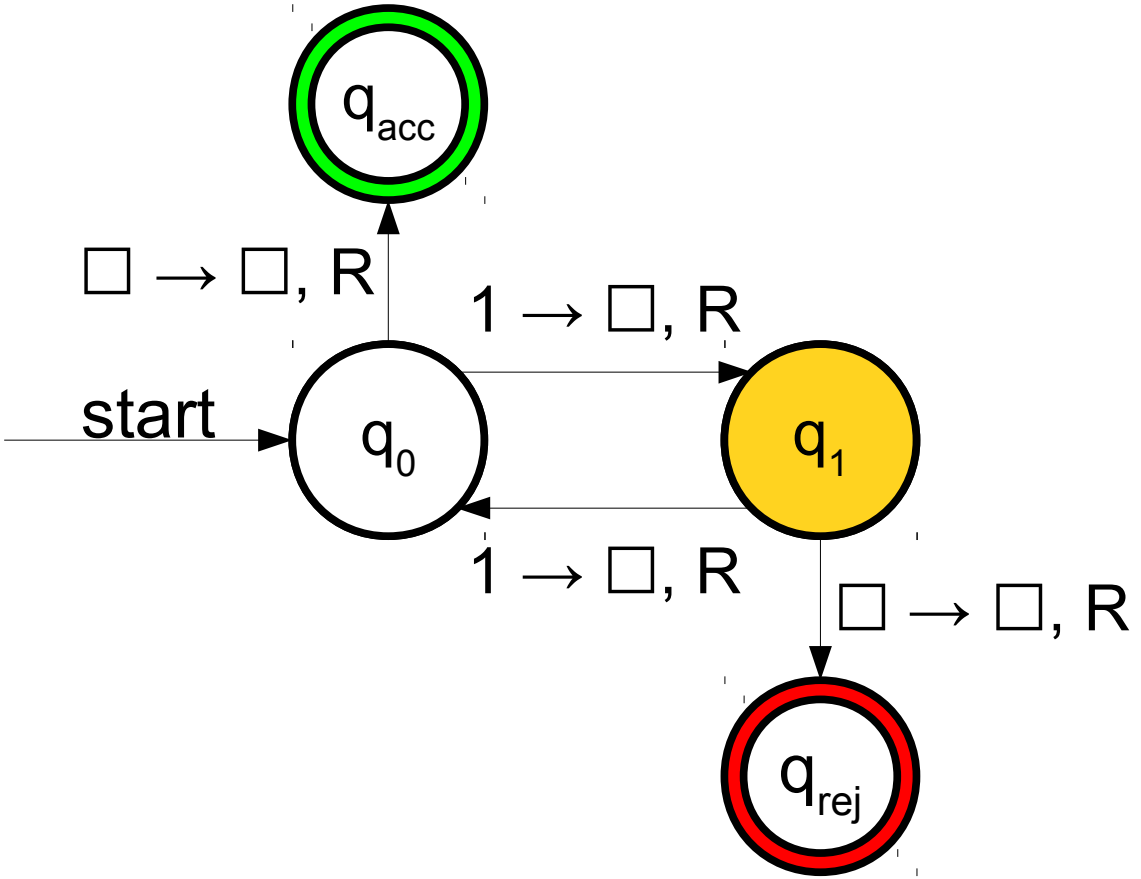


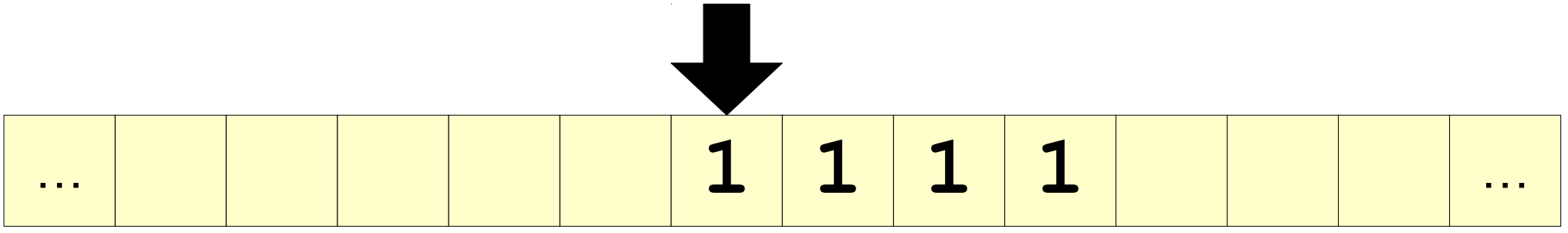
A Simple Turing Machine



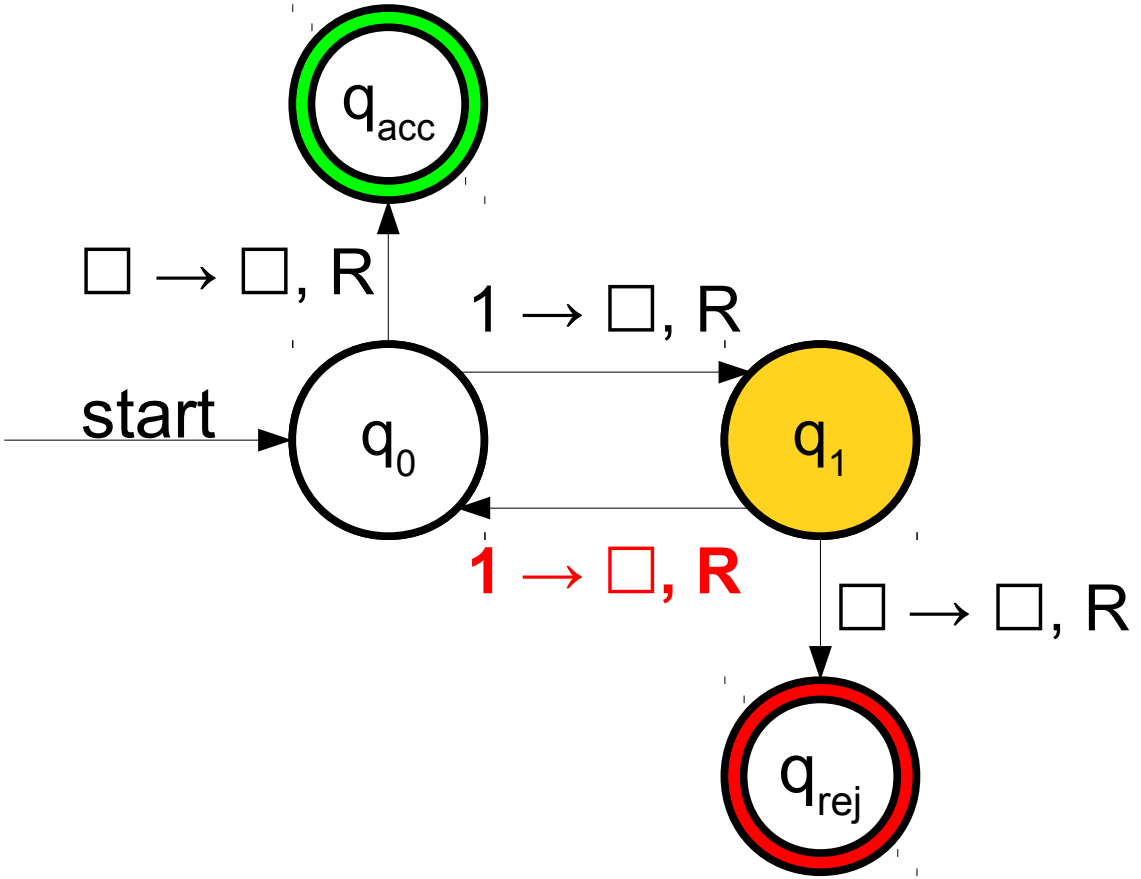


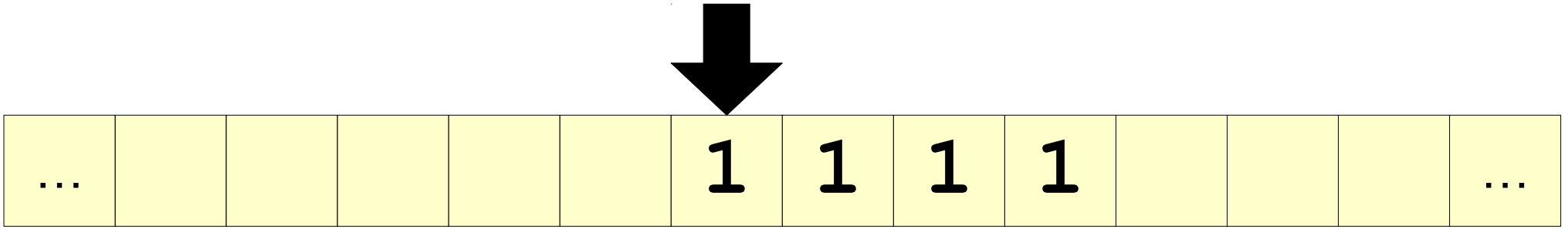
A Simple Turing Machine



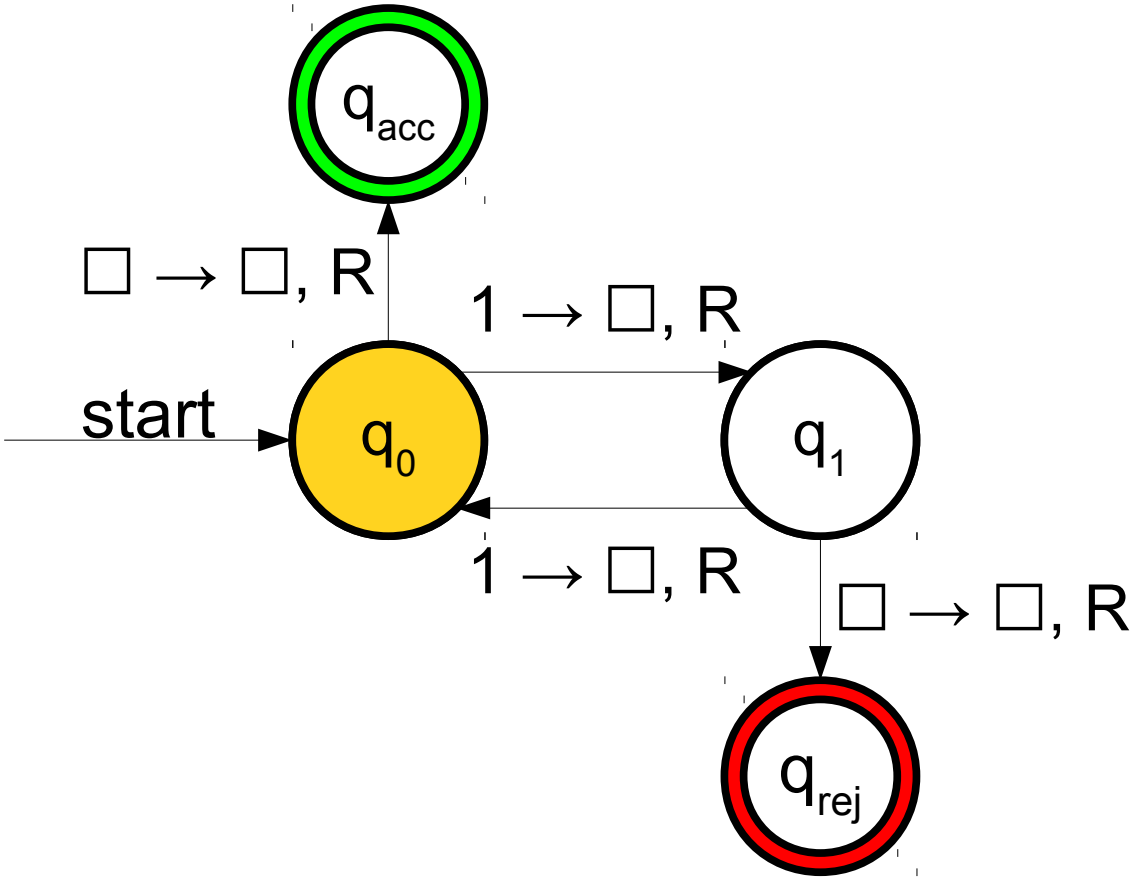


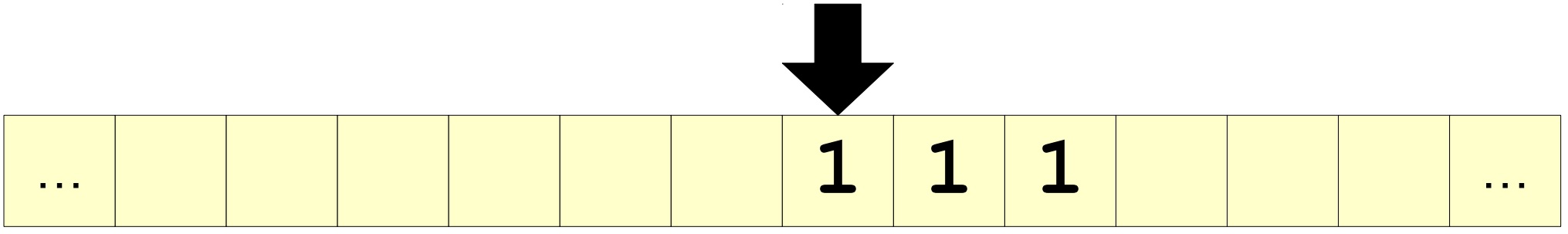
A Simple Turing Machine



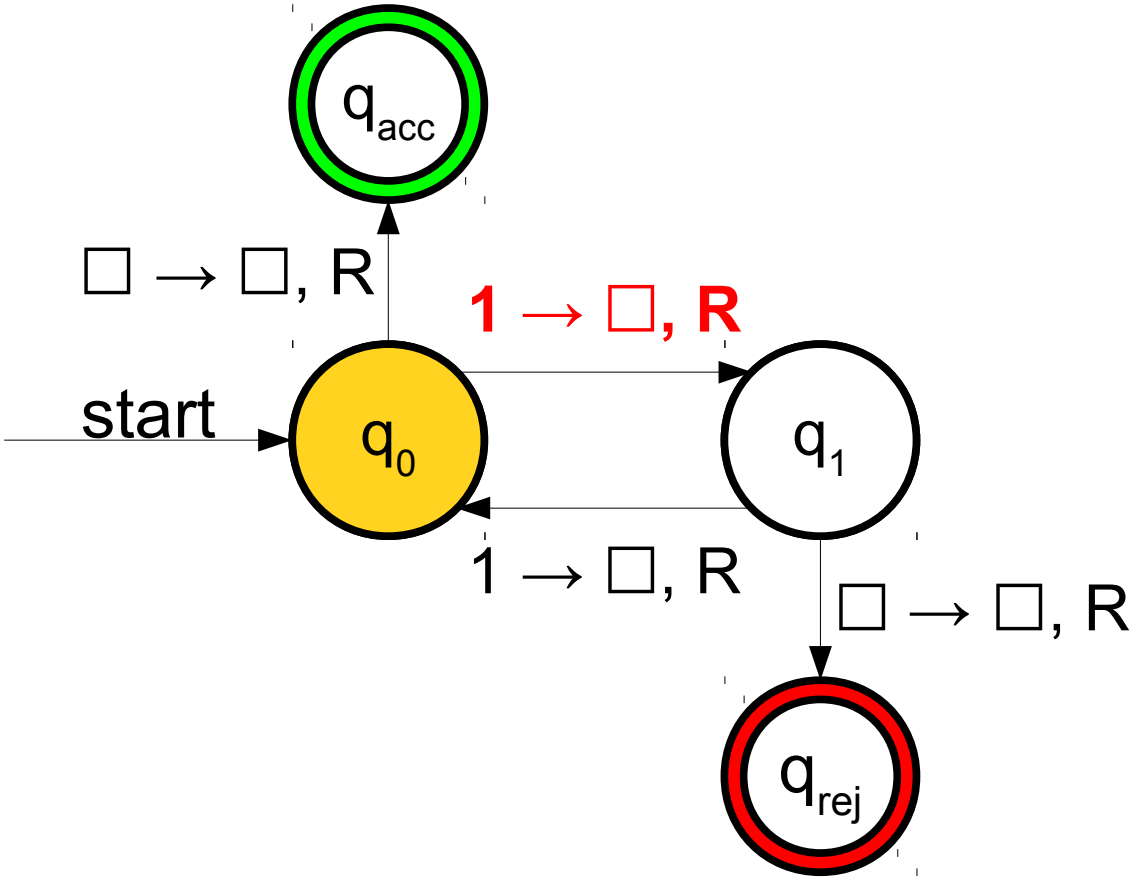


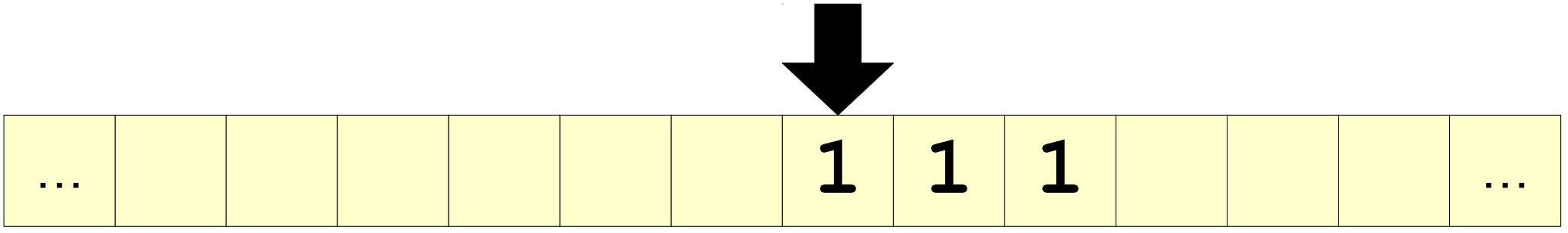
A Simple Turing Machine



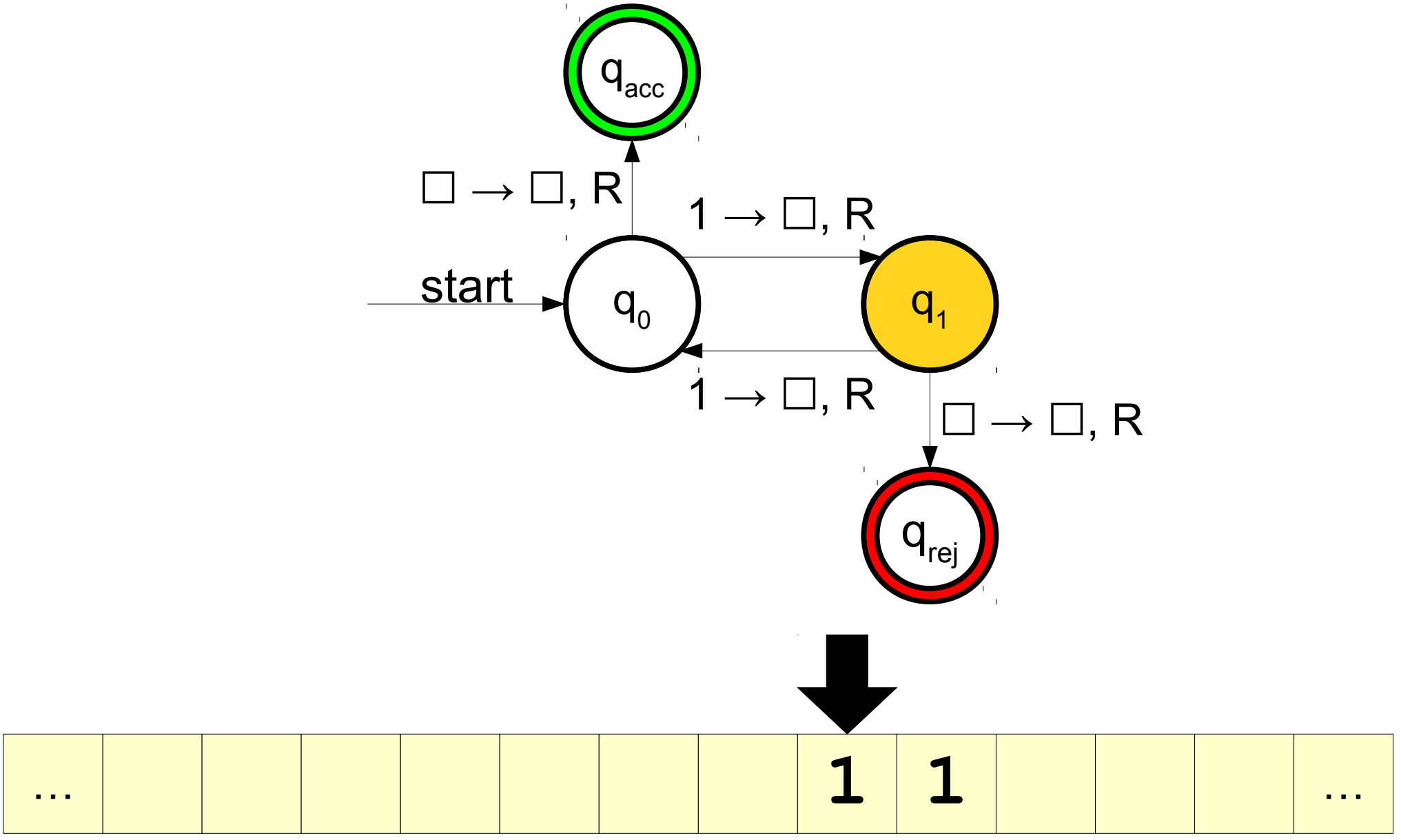


A Simple Turing Machine

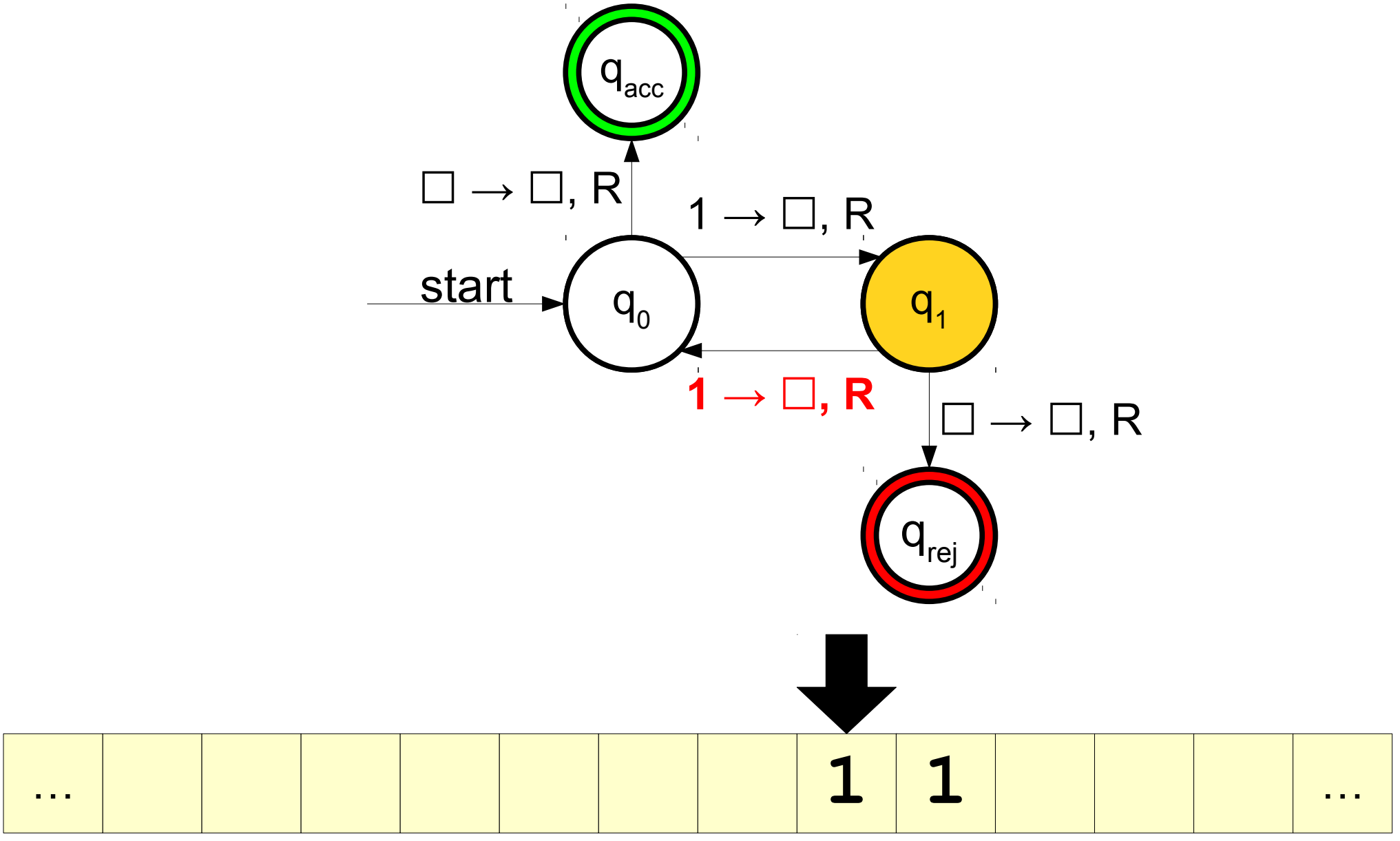




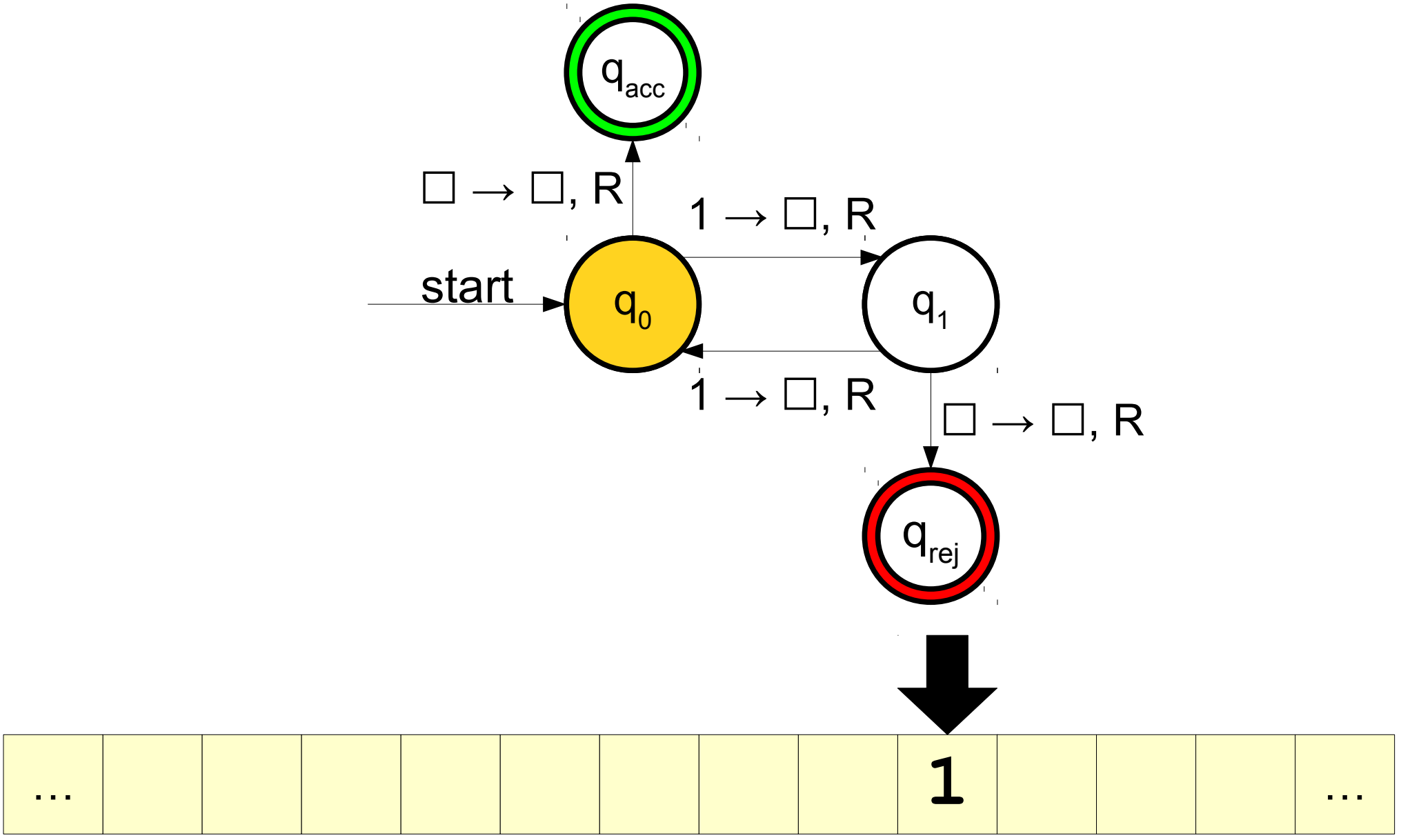
A Simple Turing Machine



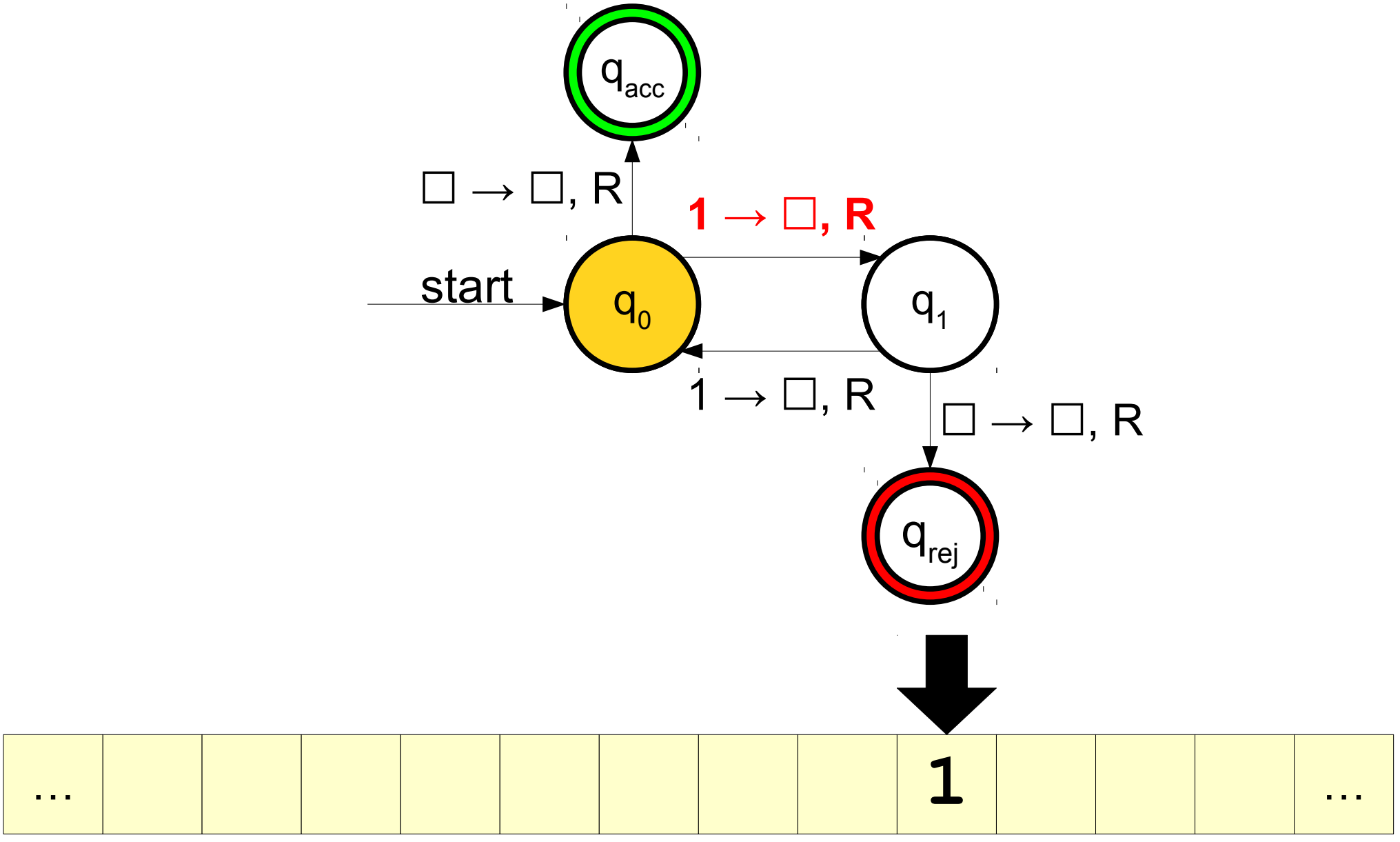
A Simple Turing Machine



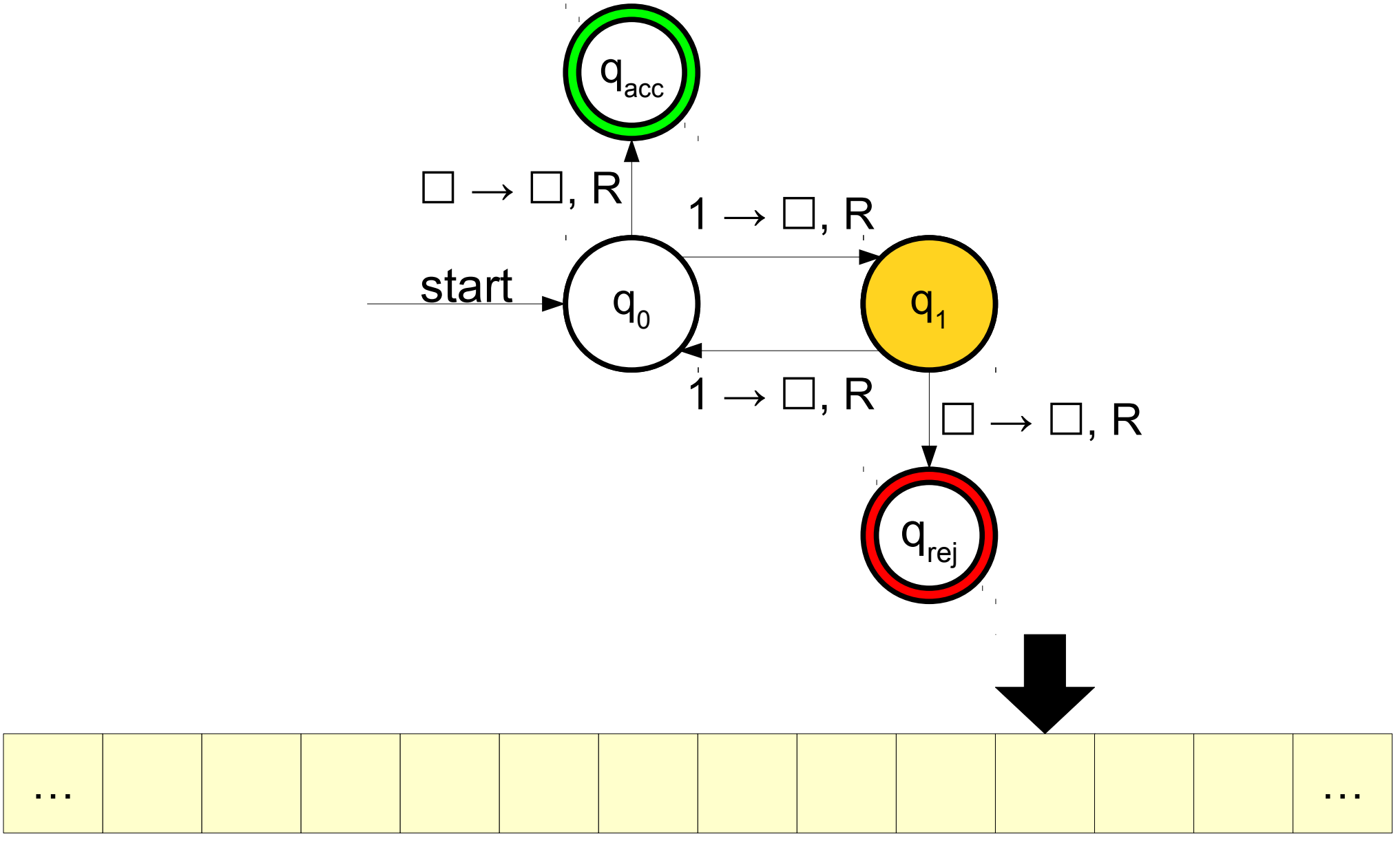
A Simple Turing Machine



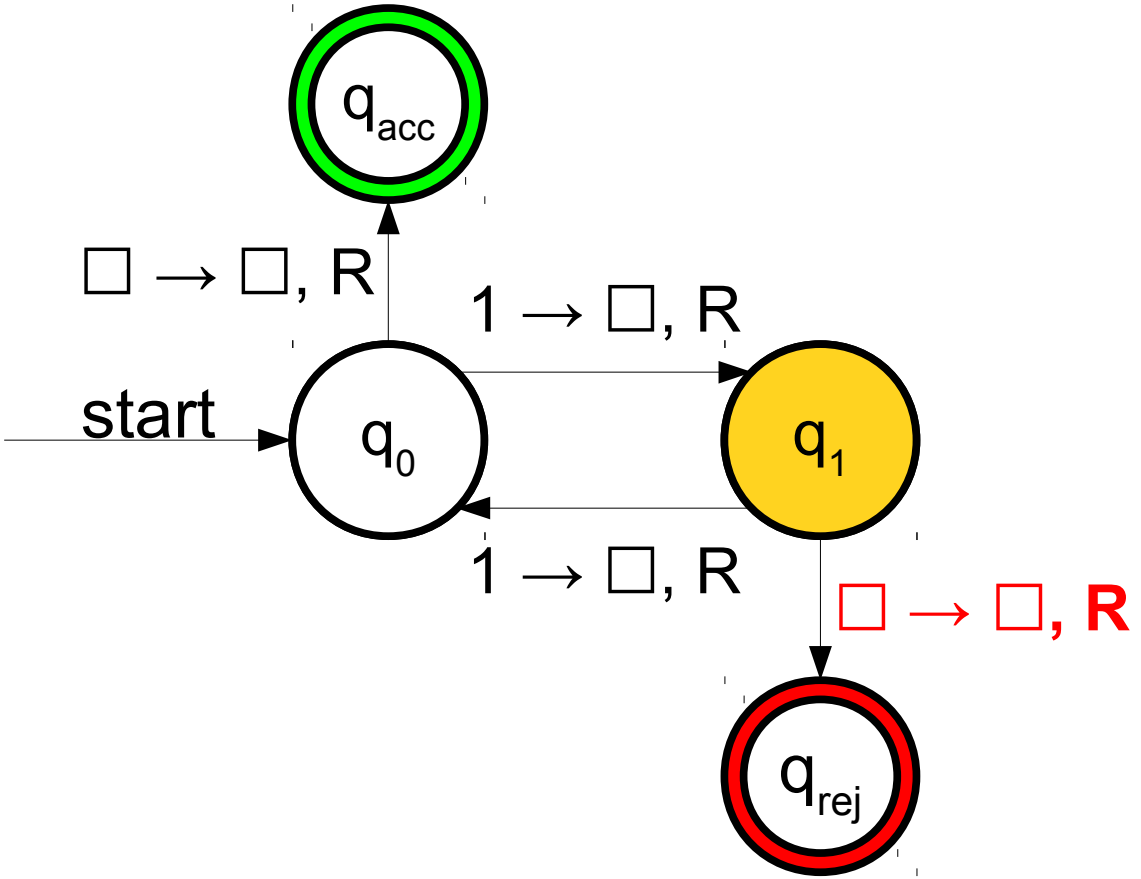
A Simple Turing Machine

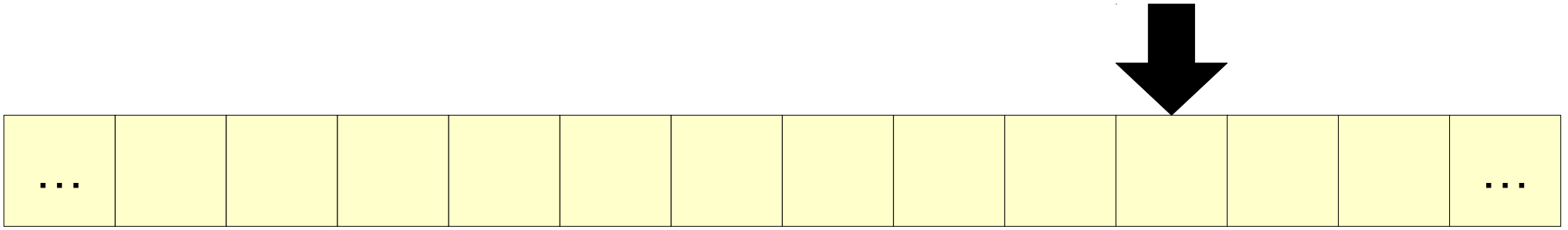


A Simple Turing Machine

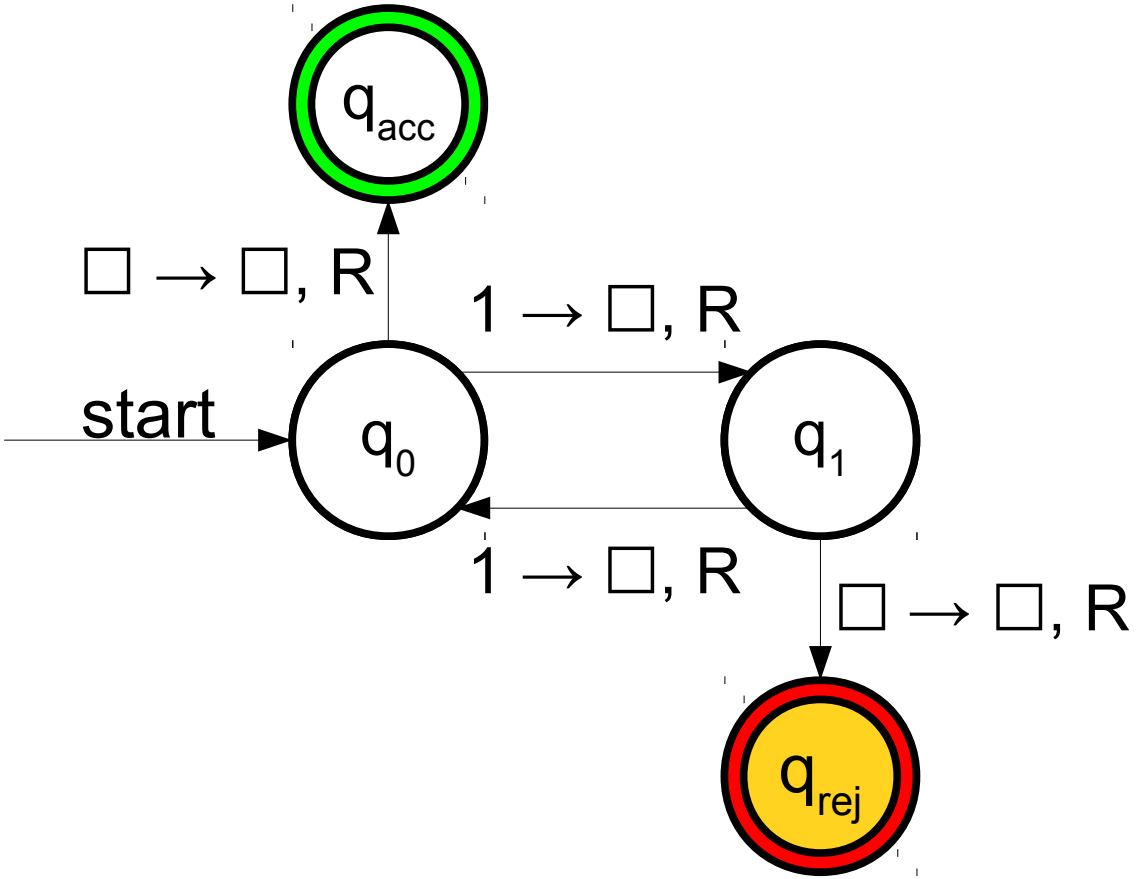


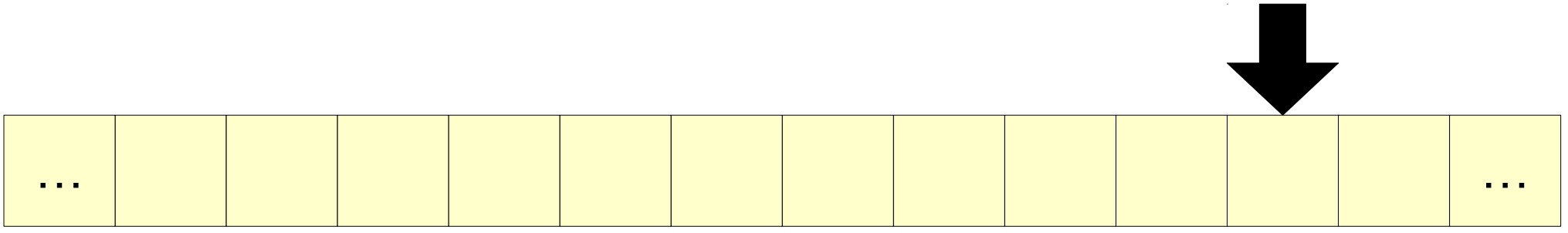
A Simple Turing Machine



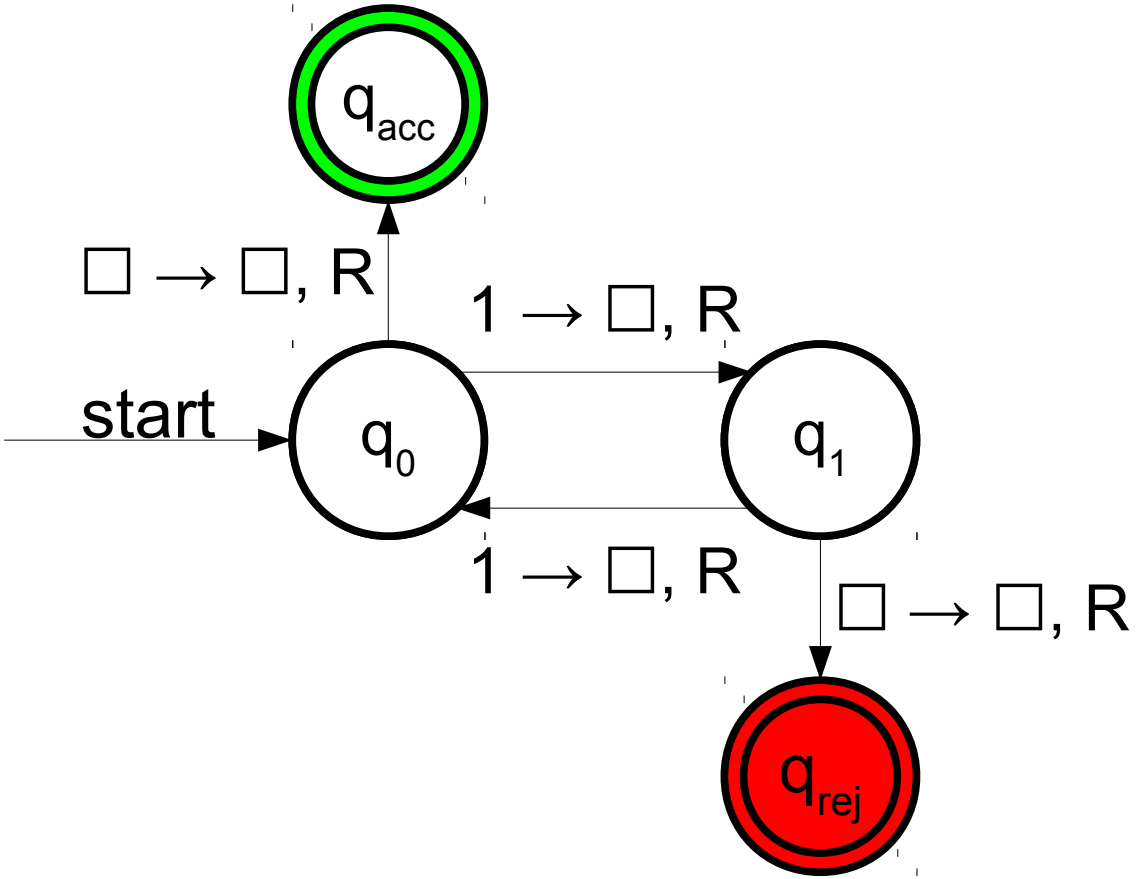


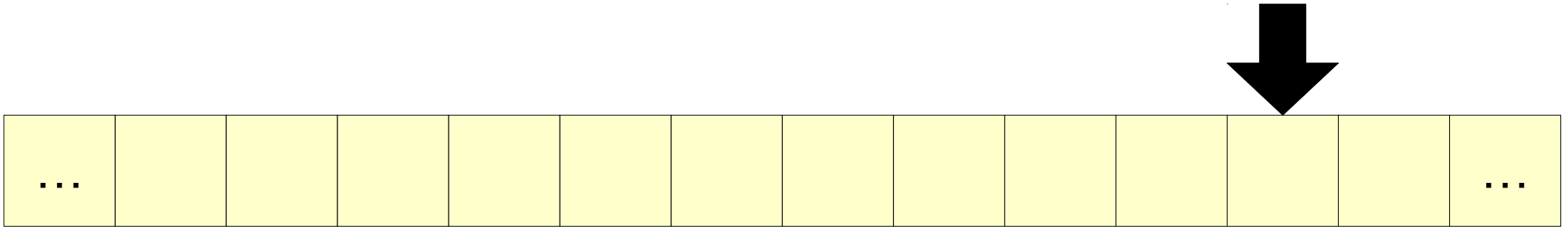
A Simple Turing Machine



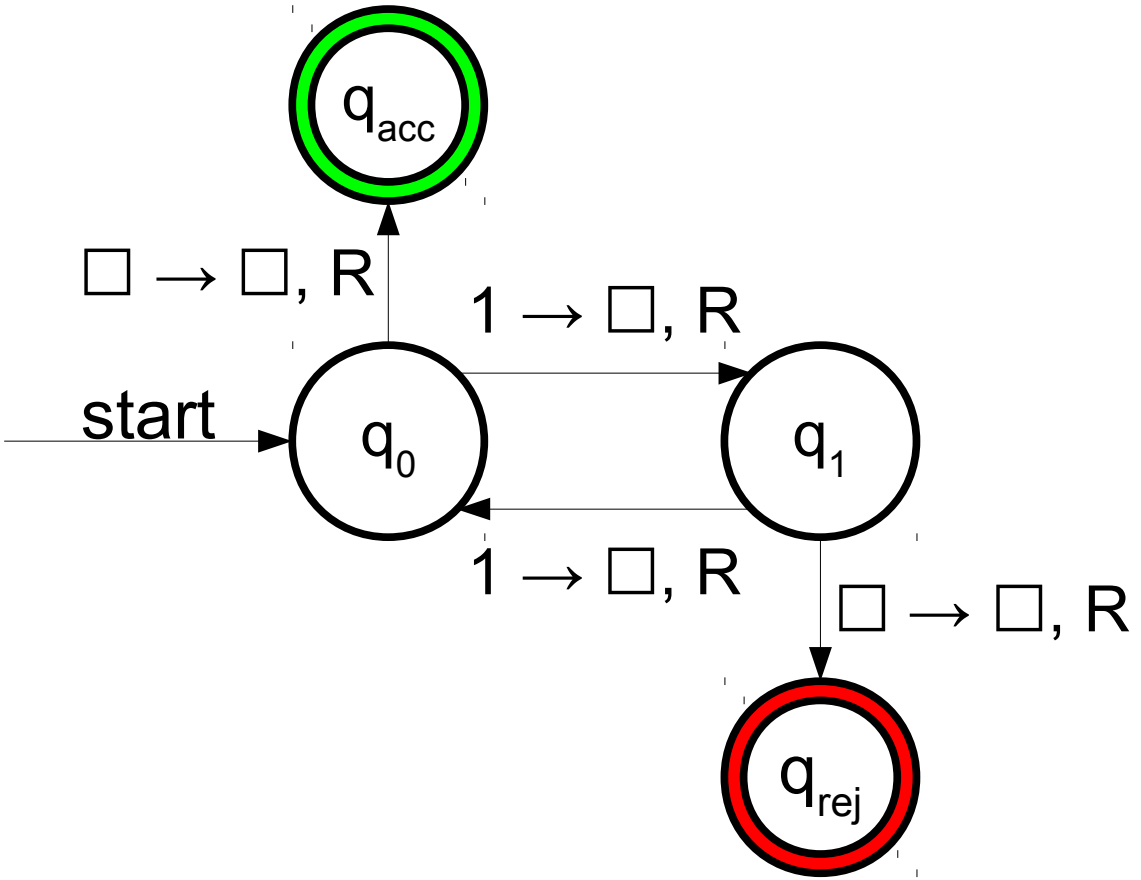


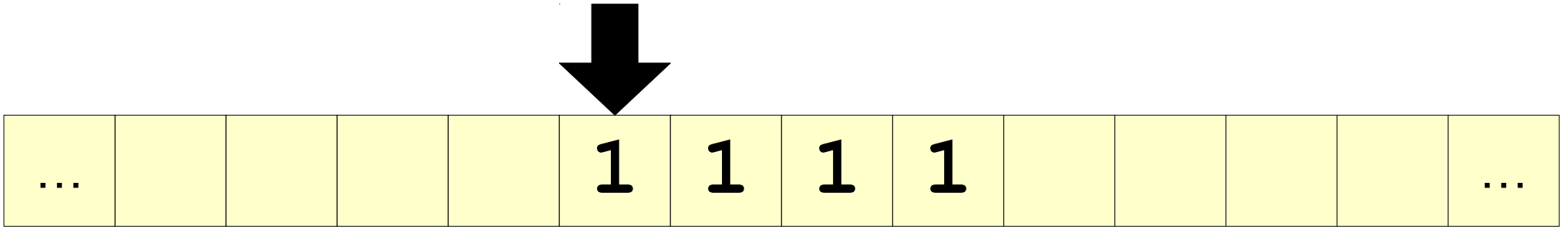
A Simple Turing Machine



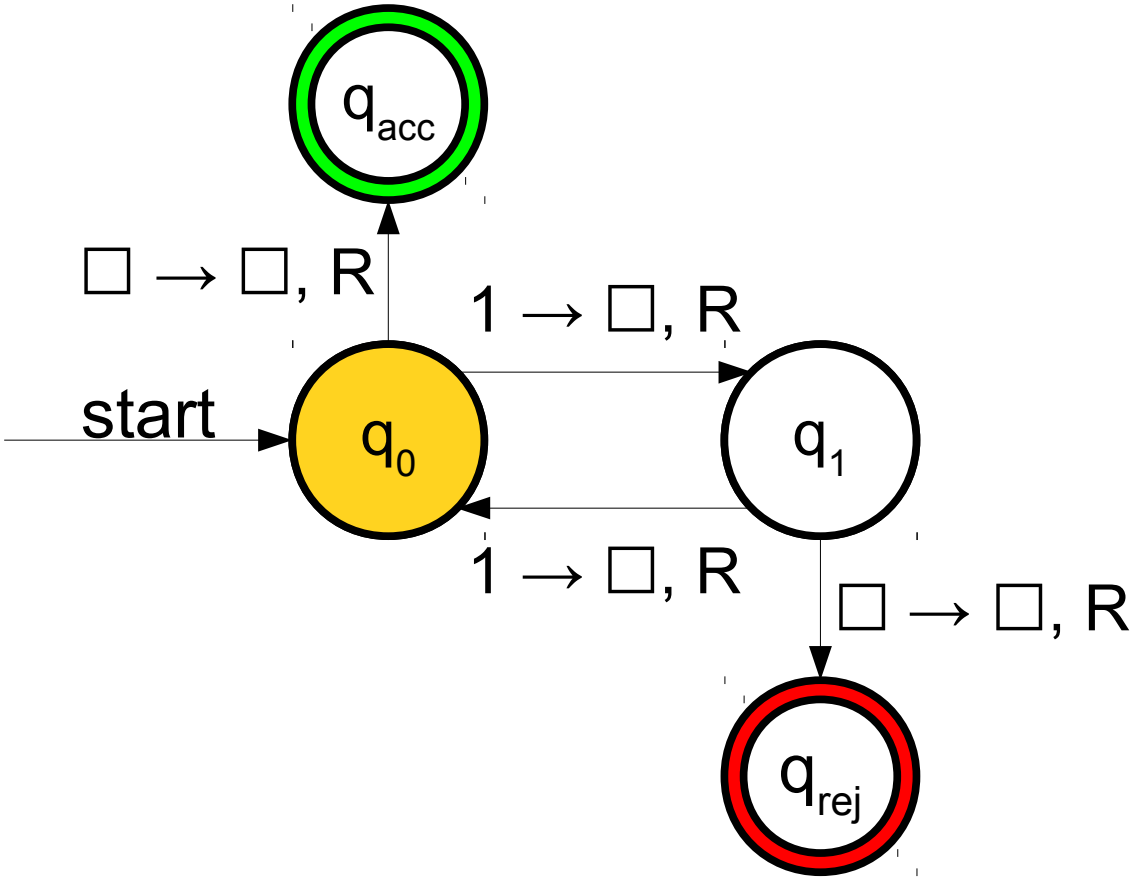


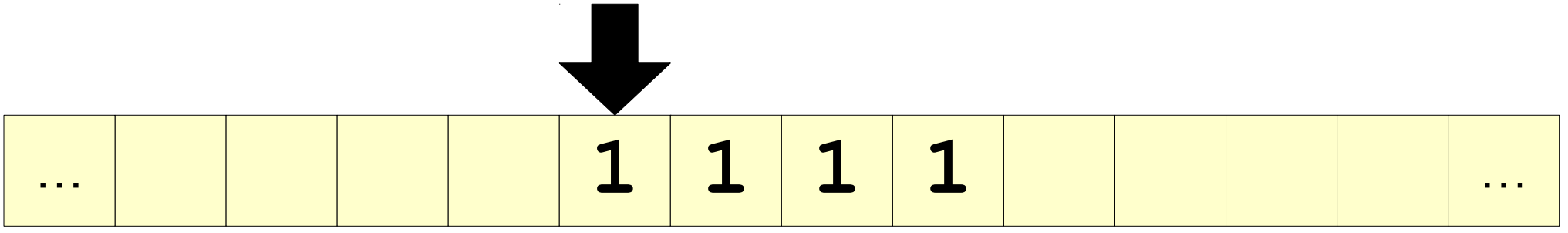
A Simple Turing Machine



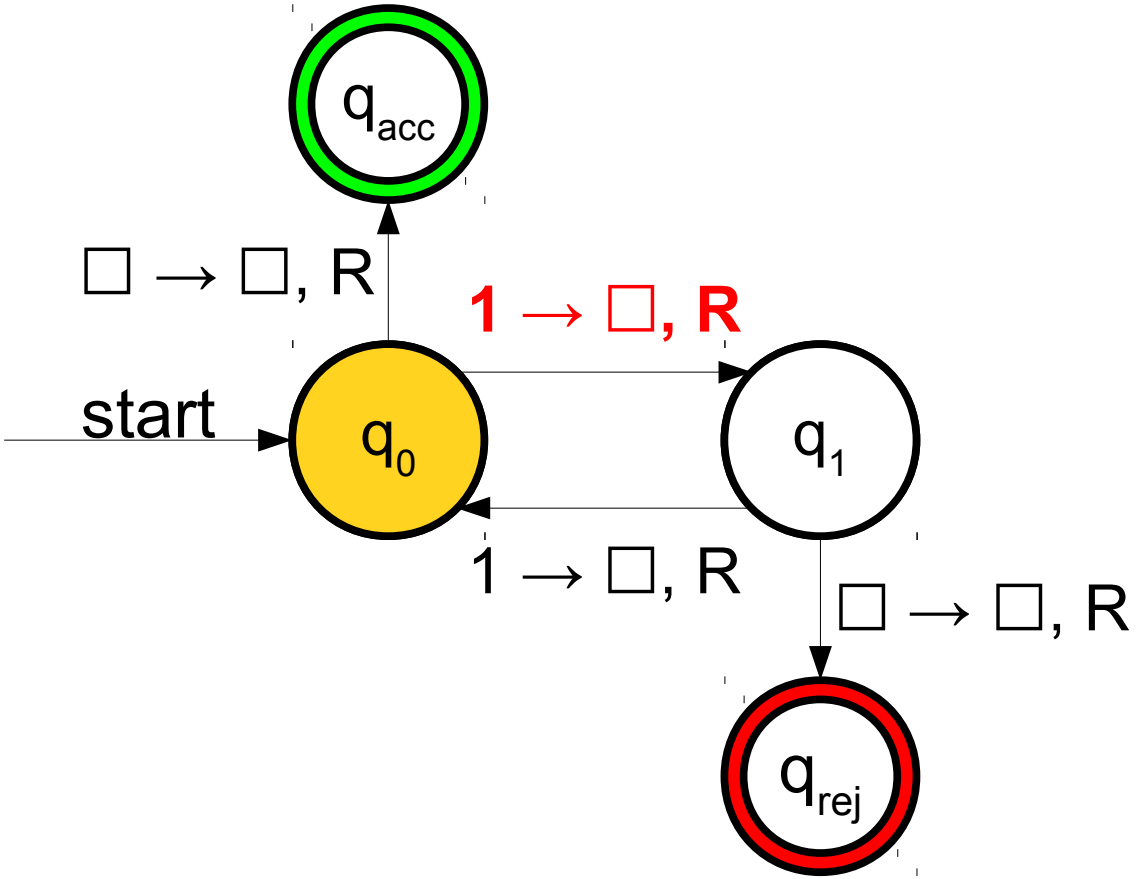


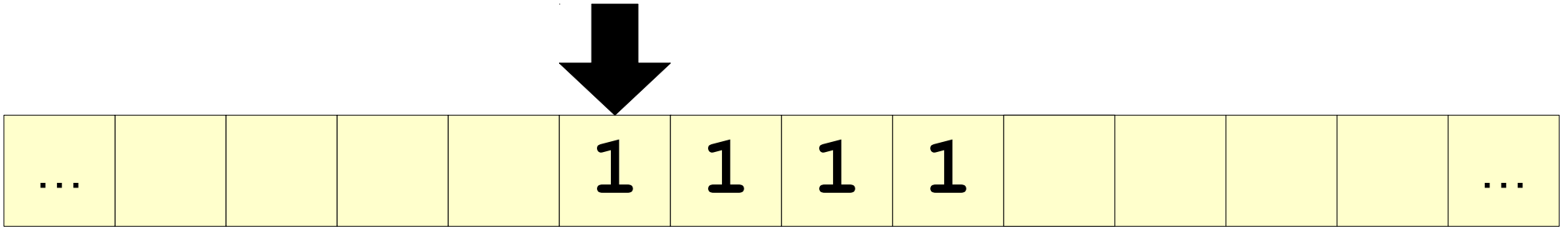
A Simple Turing Machine



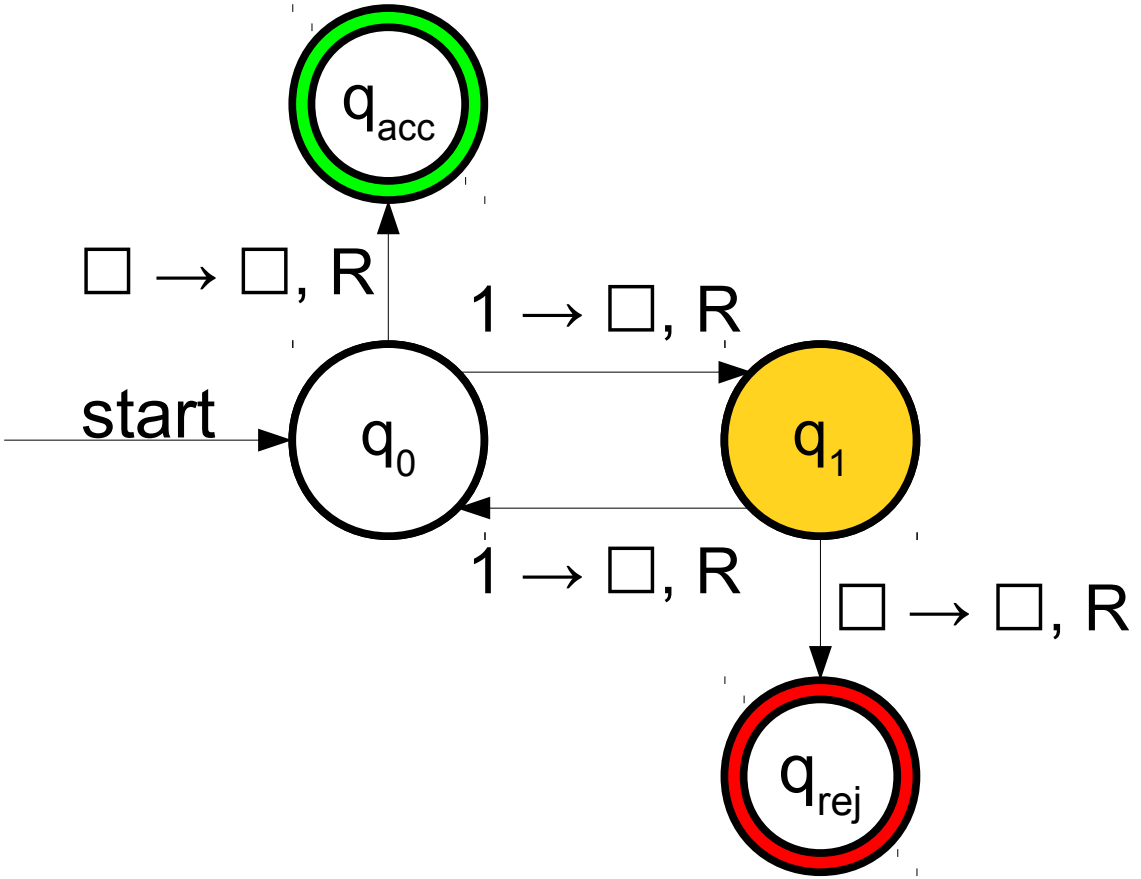


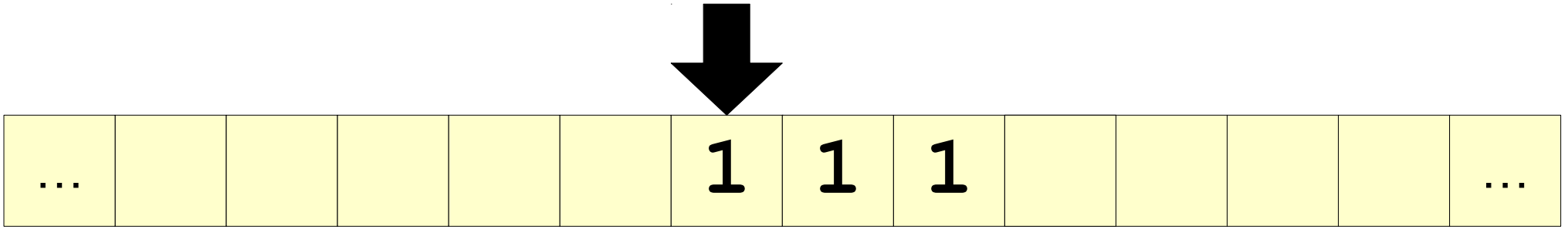
A Simple Turing Machine



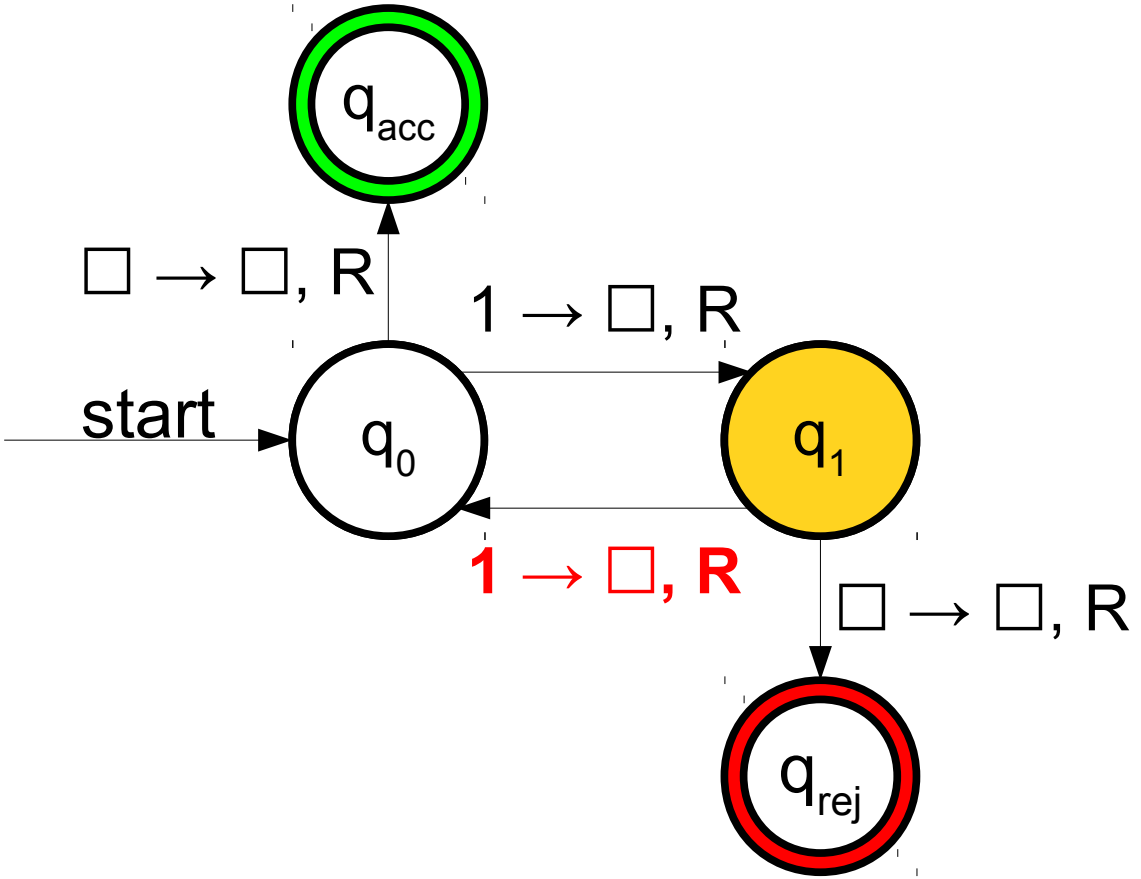


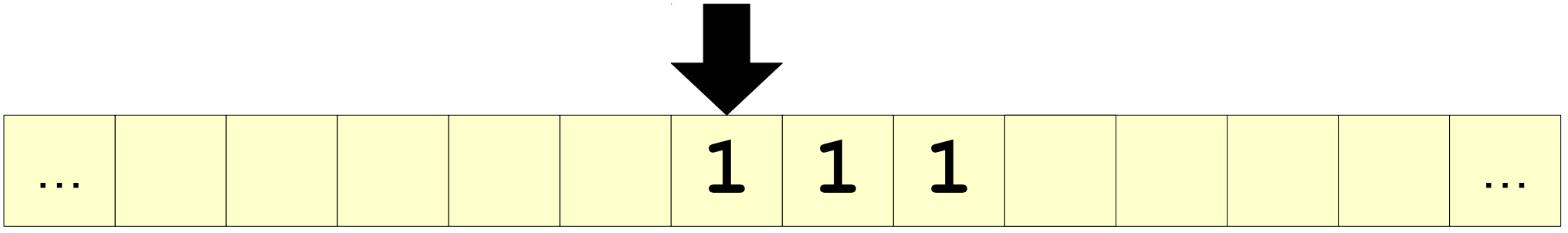
A Simple Turing Machine



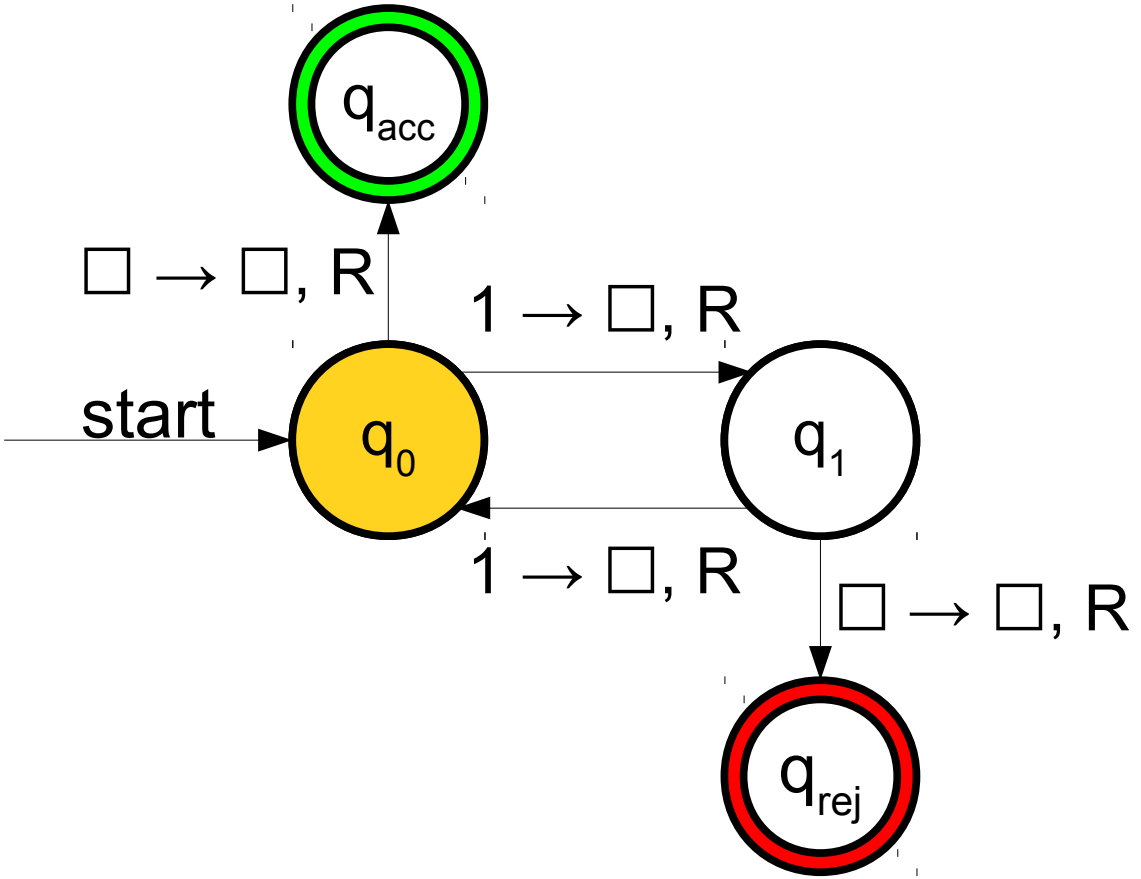


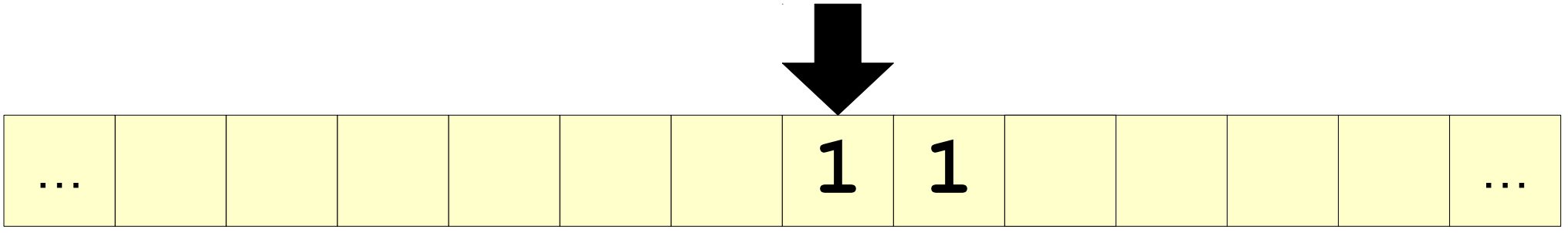
A Simple Turing Machine



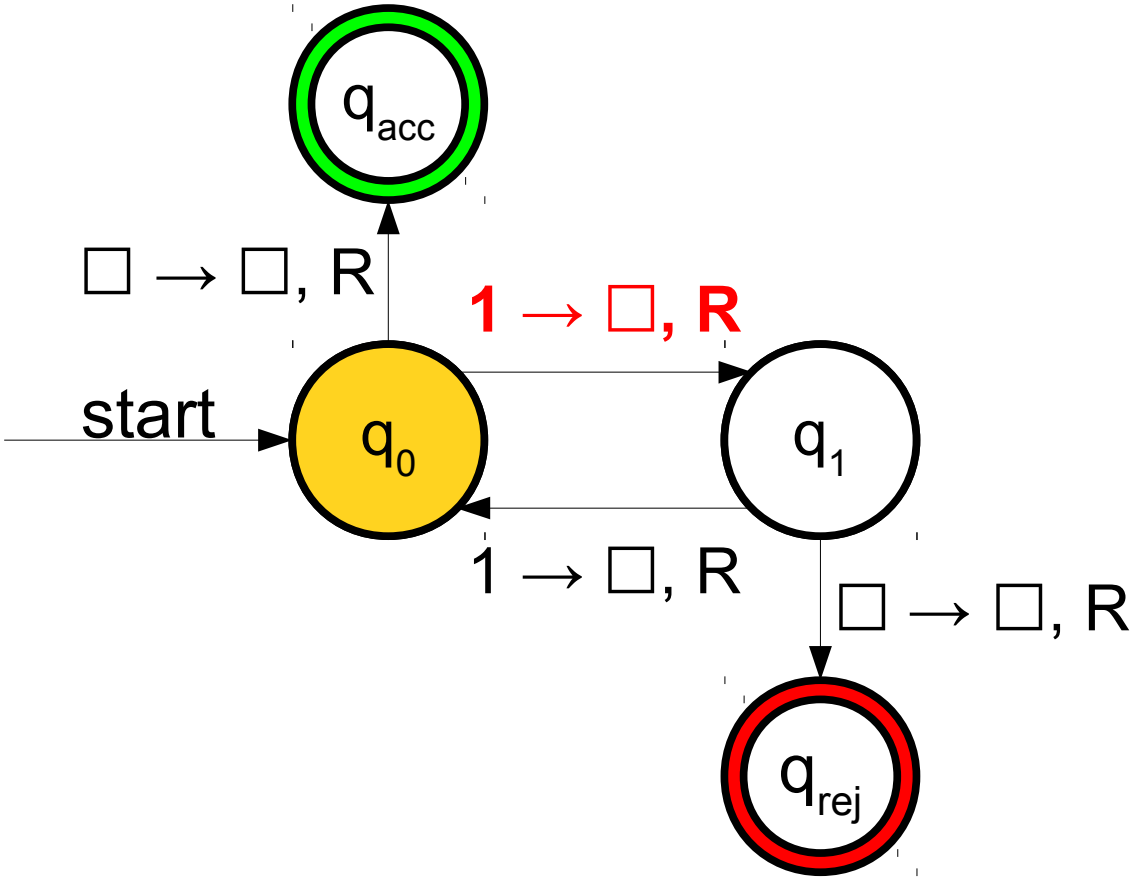


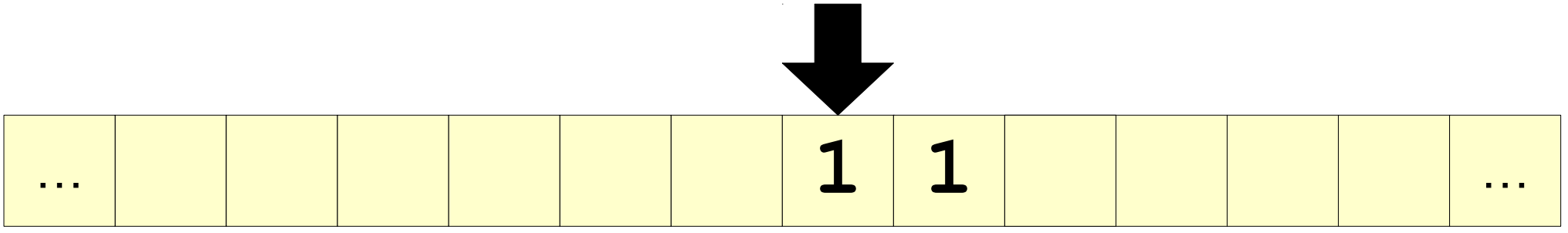
A Simple Turing Machine



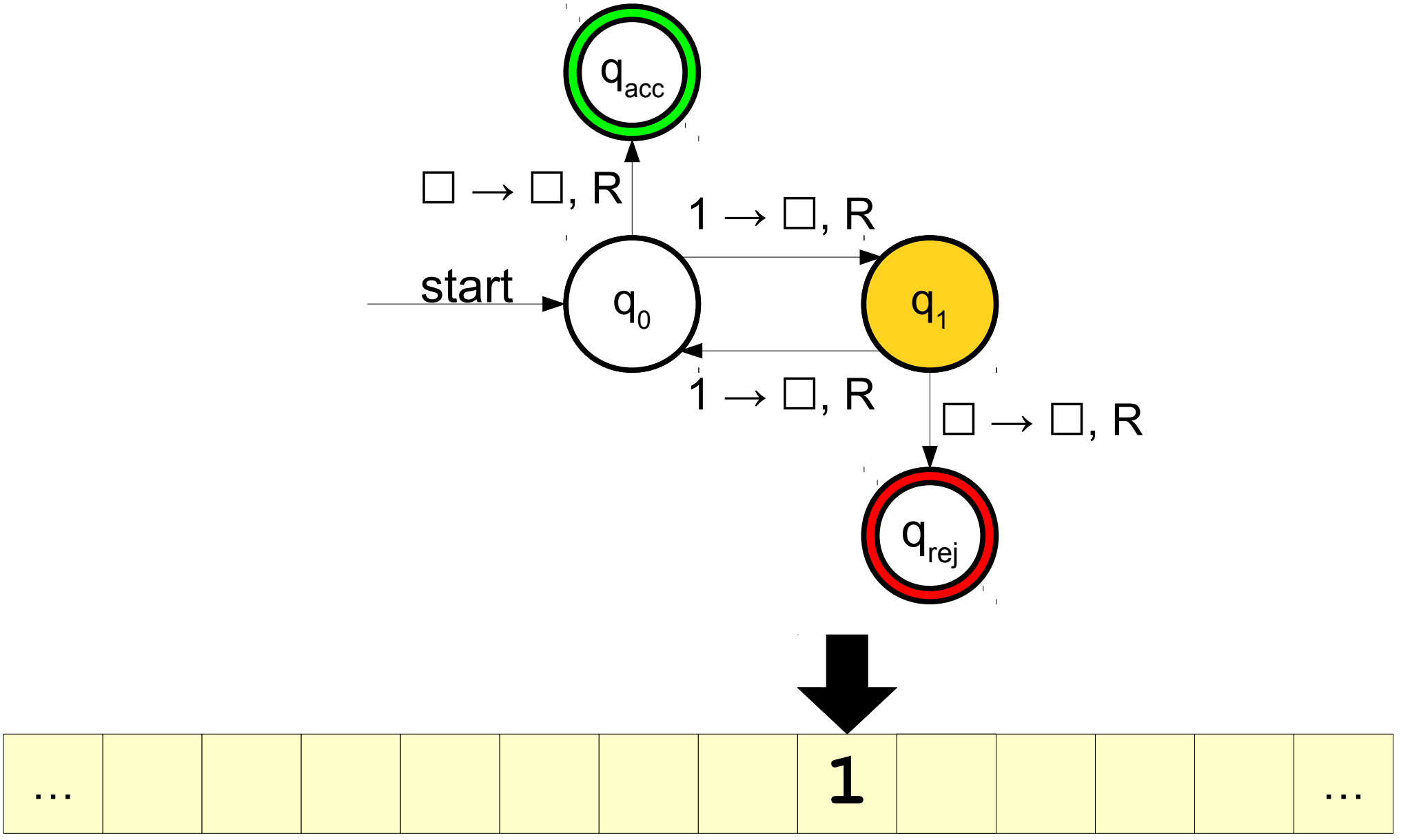


A Simple Turing Machine

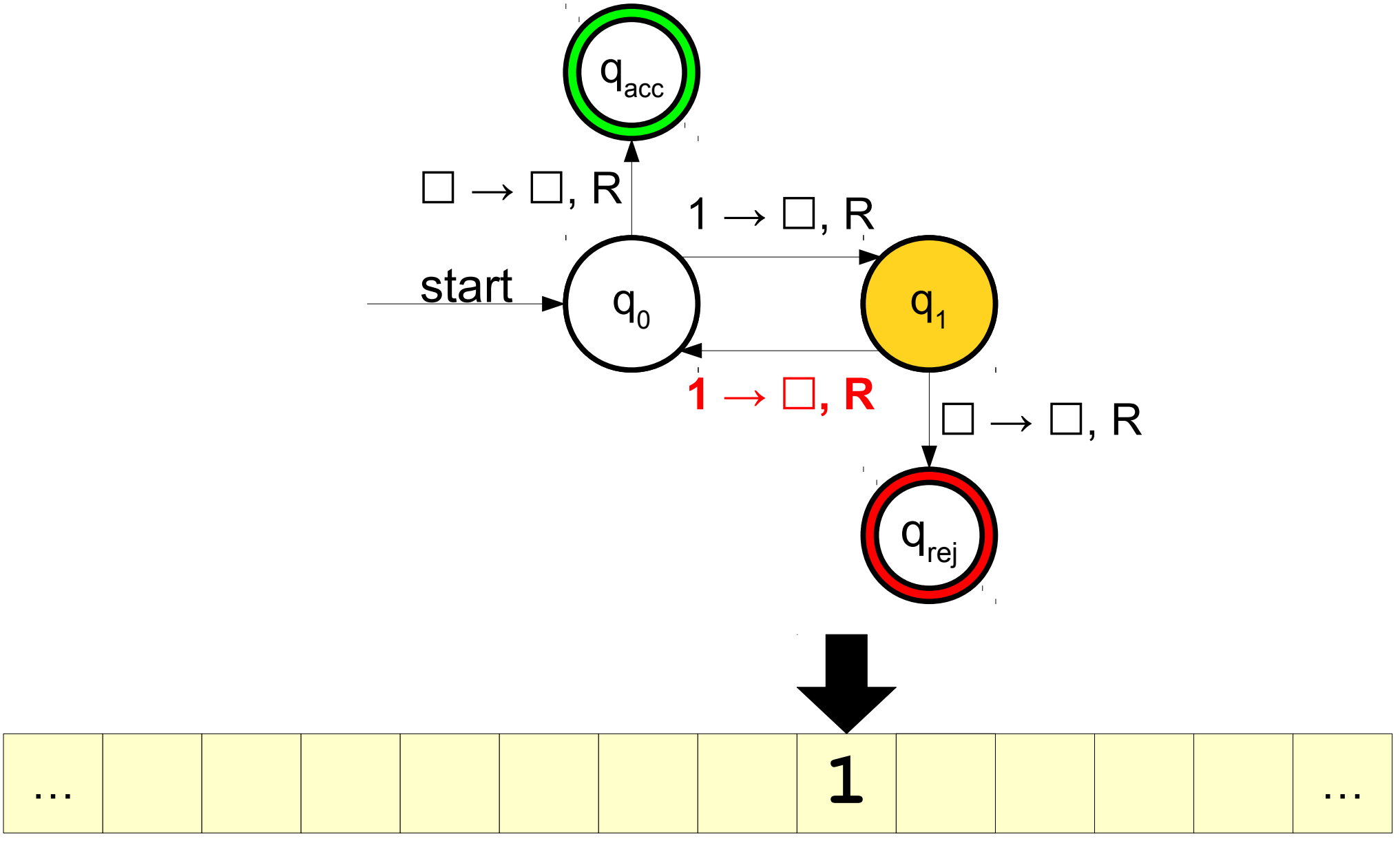




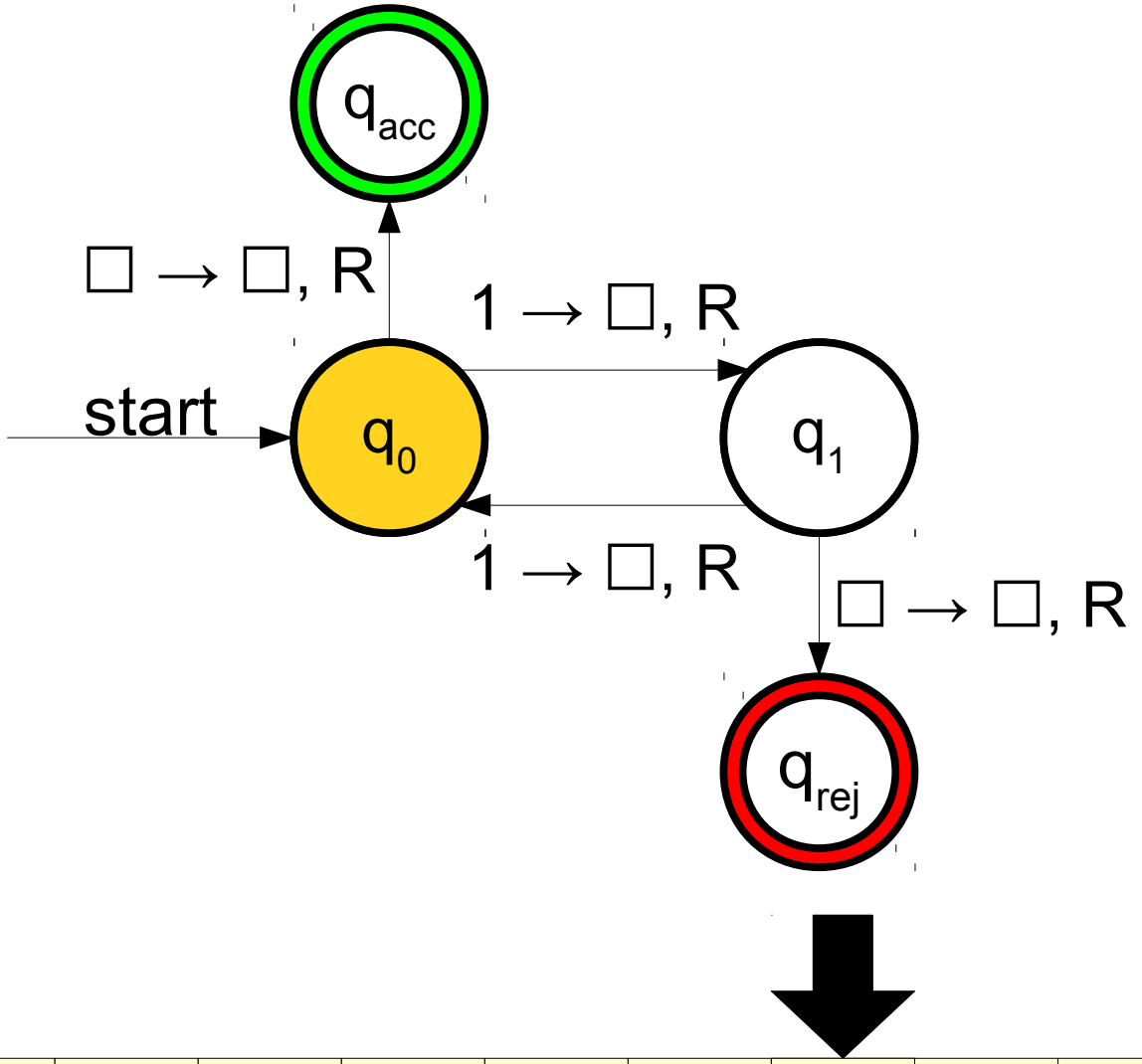
A Simple Turing Machine



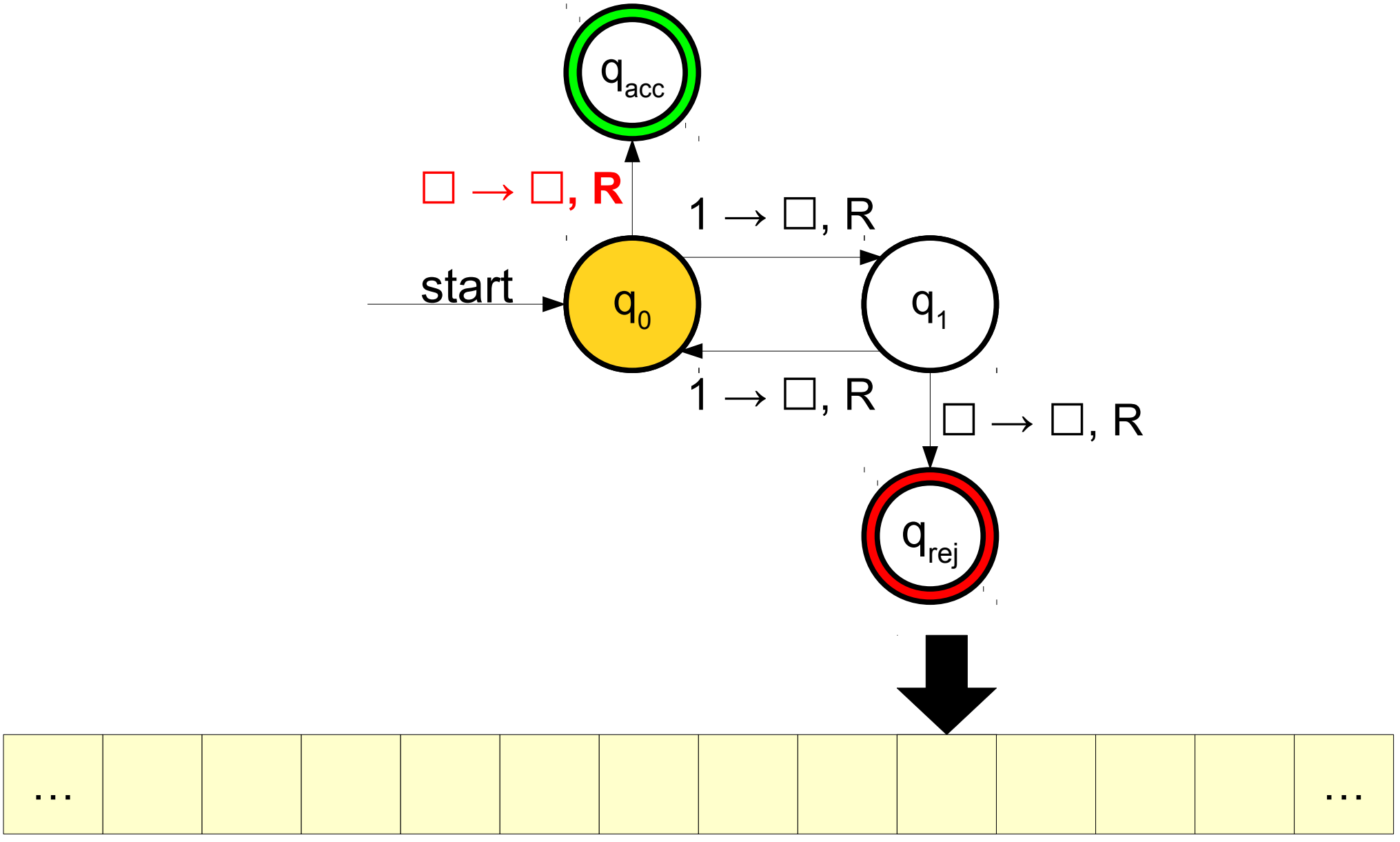
A Simple Turing Machine



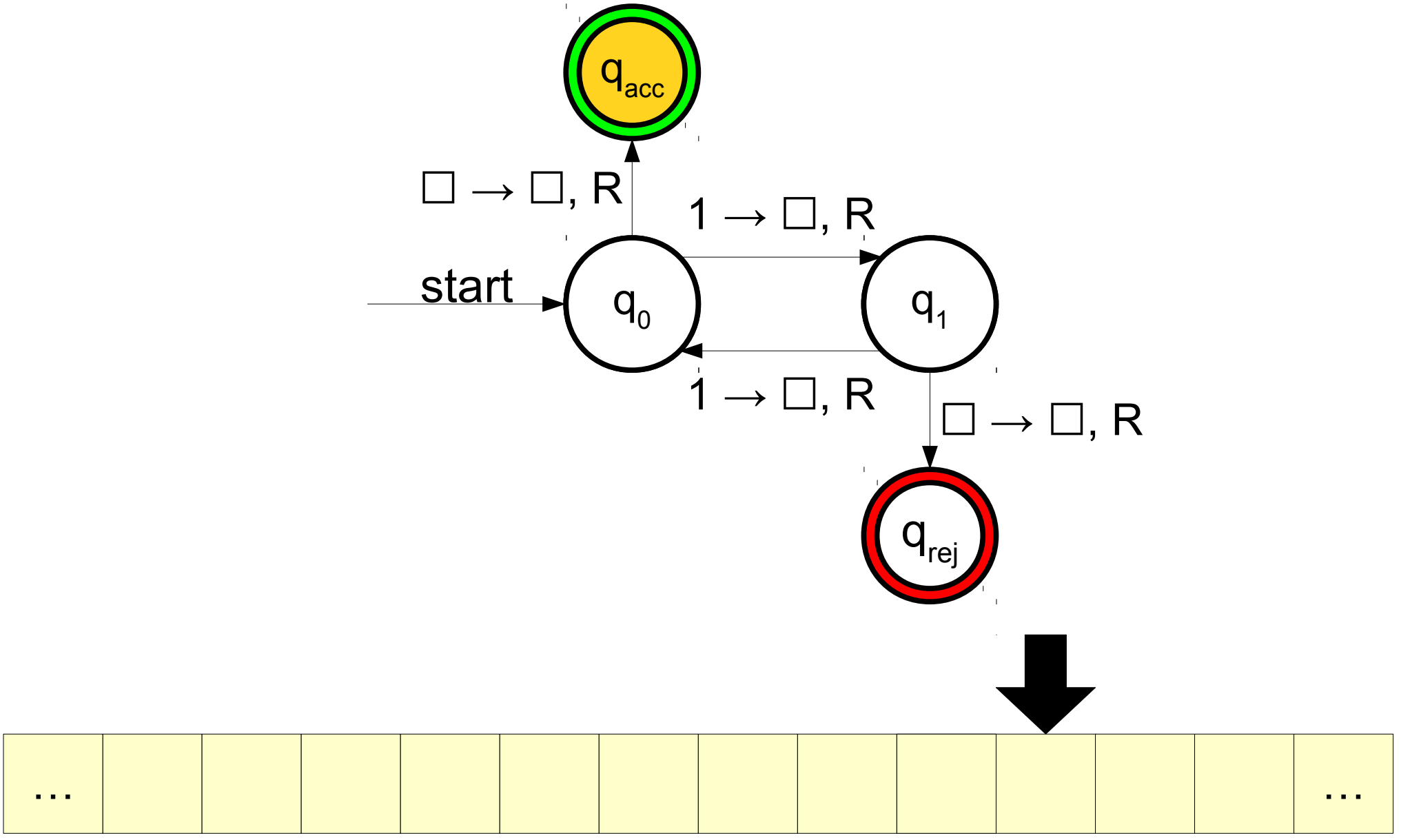
A Simple Turing Machine



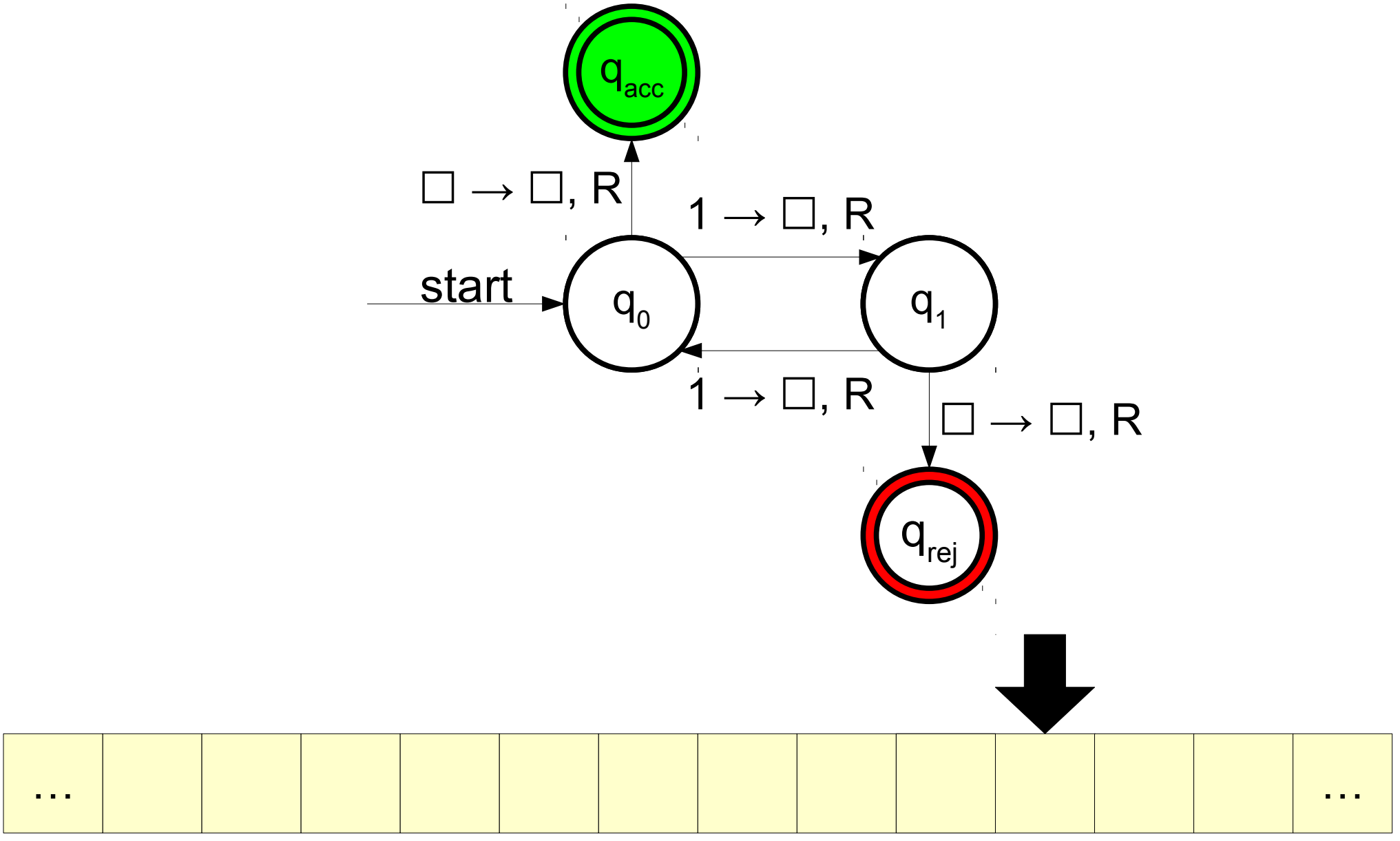
A Simple Turing Machine



A Simple Turing Machine



A Simple Turing Machine



Accepting and Rejecting States

- Unlike DFAs, Turing machines do not stop processing the input when they finish reading it.
- Turing machines decide when (and if!) they will accept or reject their input.
- Turing machines can enter infinite loops and never accept or reject; more on that later...

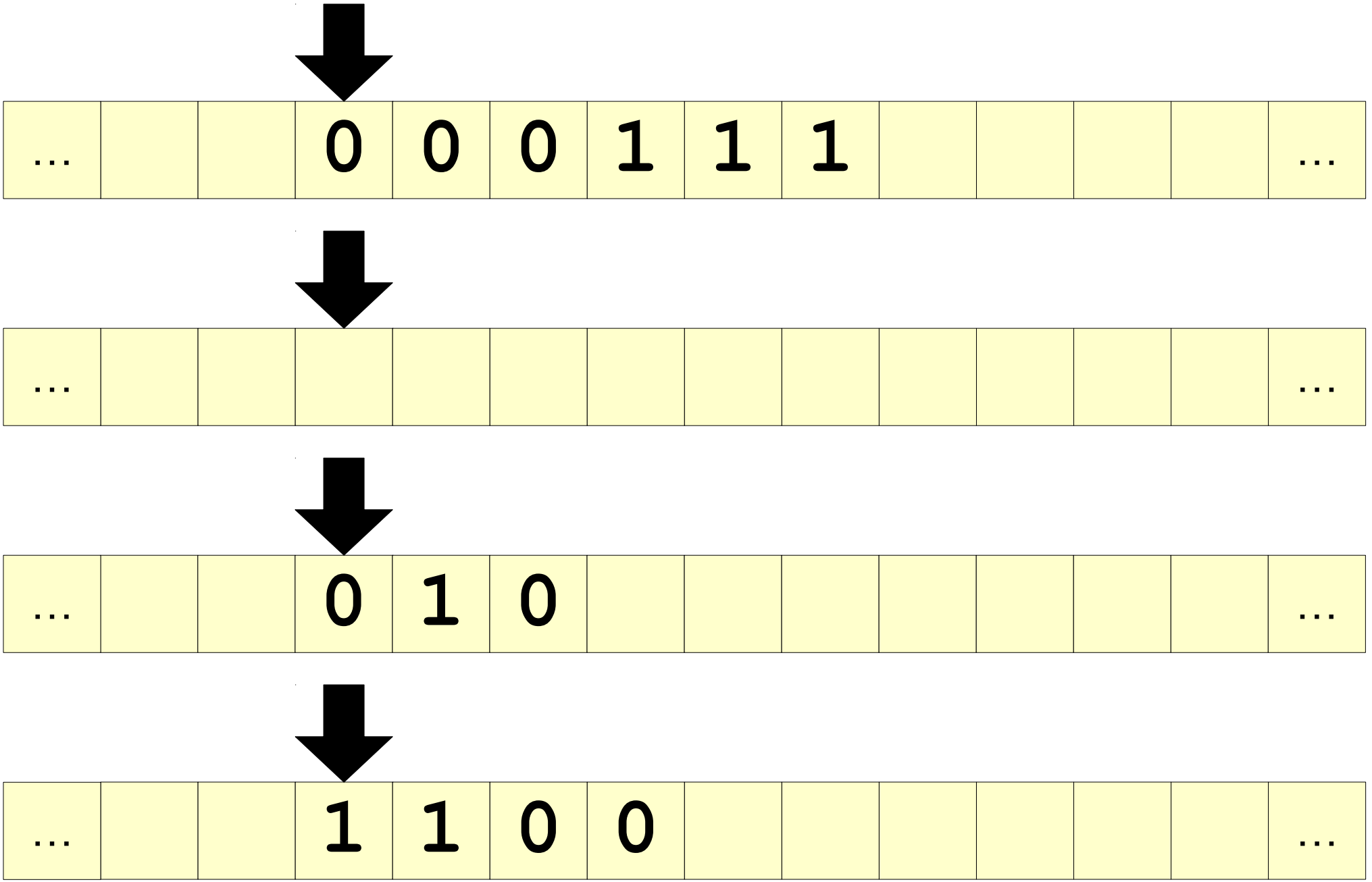
Designing Turing Machines

- Despite their simplicity, Turing machines are very powerful computing devices.
- Today's lecture explores how to design Turing machines for various languages.

Designing Turing Machines

- Let $\Sigma = \{\texttt{0}, \texttt{1}\}$ and consider the language $L = \{\texttt{0}^n\texttt{1}^n \mid n \in \mathbb{N}\}$.
- We know that L is context-free.
- How might we build a Turing machine for it?

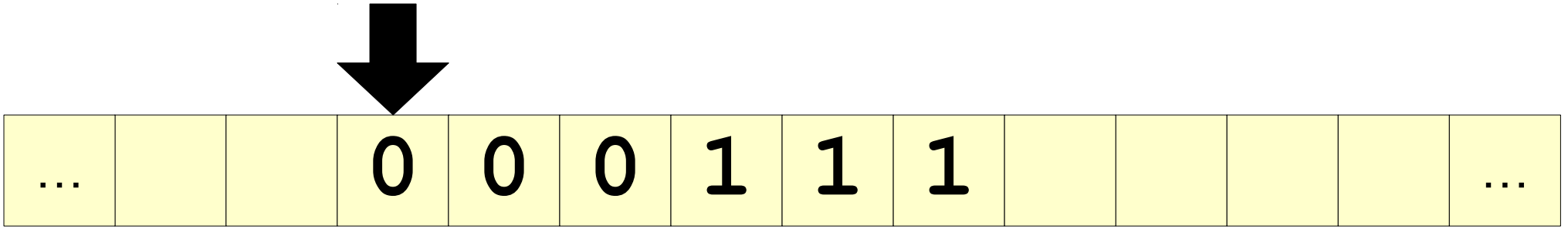
$$L = \{ \mathtt{0}^n \mathtt{1}^n \mid n \in \mathbb{N} \}$$



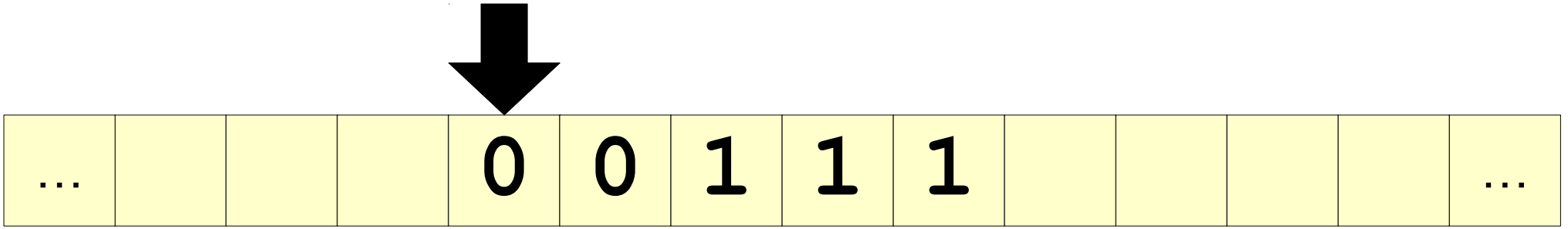
A Recursive Approach

- The string ε is in L.
- The string `0`w`1` is in L iff w is in L.
- Any string starting with `1` is not in L.
- Any string ending with `0` is not in L.

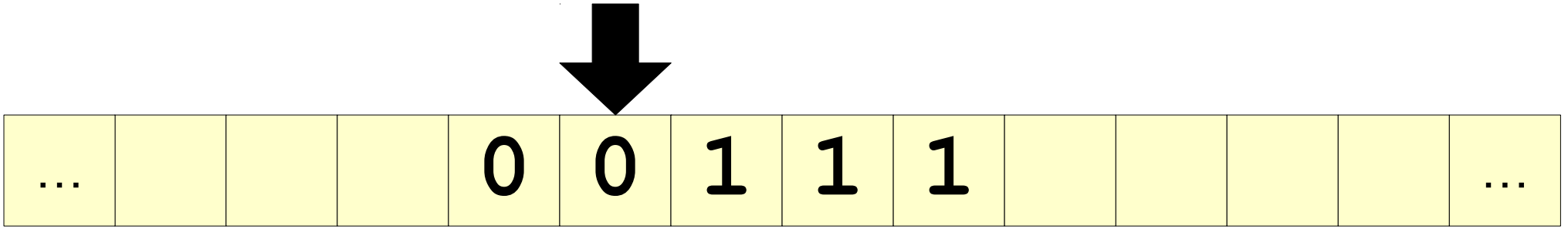
A Sketch of the TM



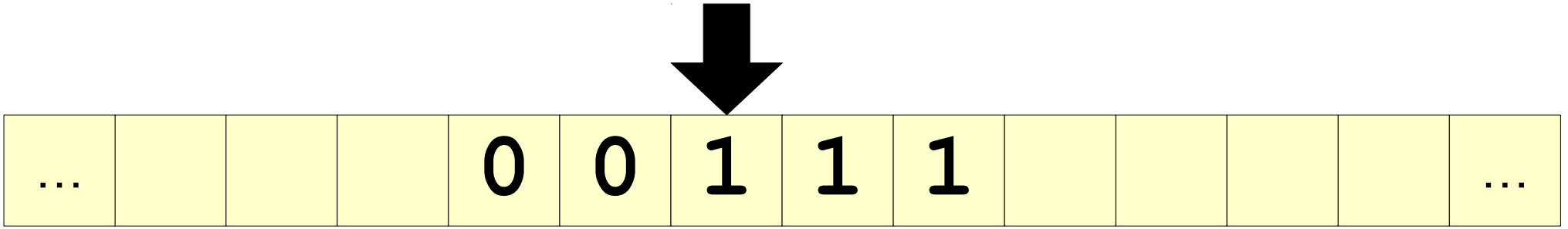
A Sketch of the TM



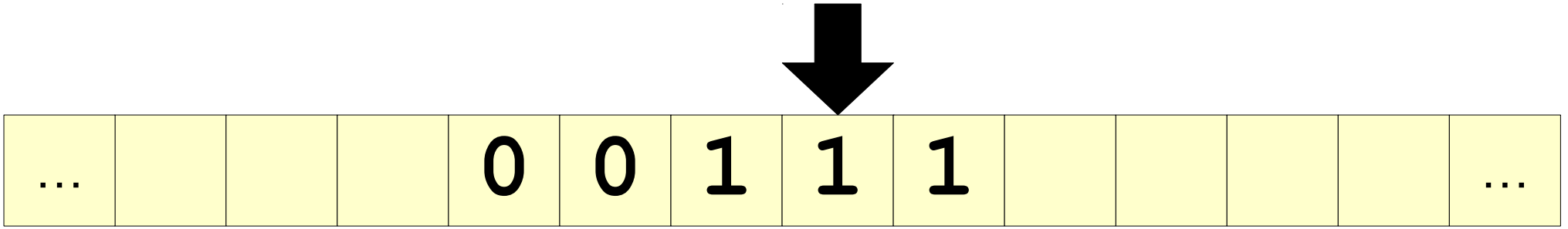
A Sketch of the TM



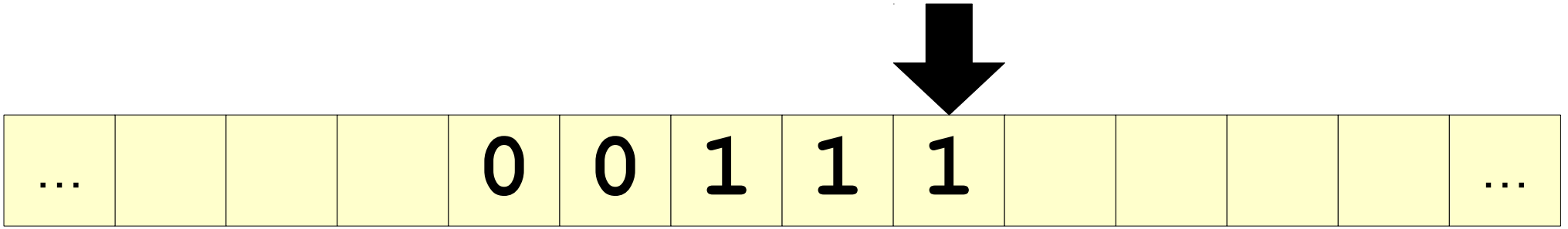
A Sketch of the TM



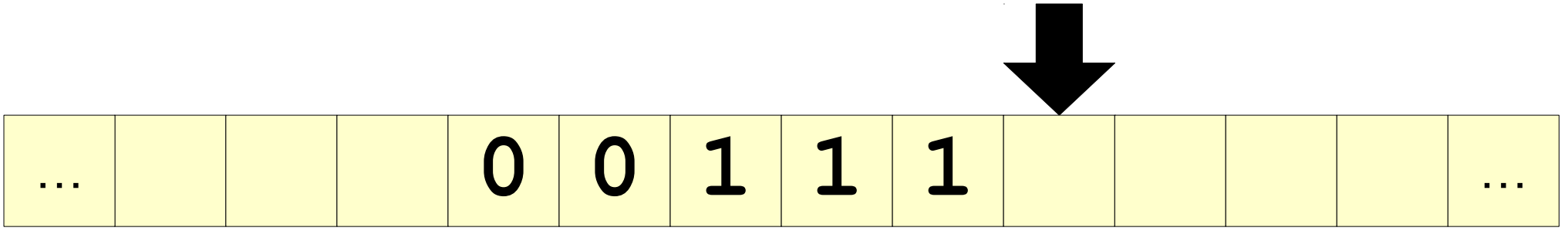
A Sketch of the TM



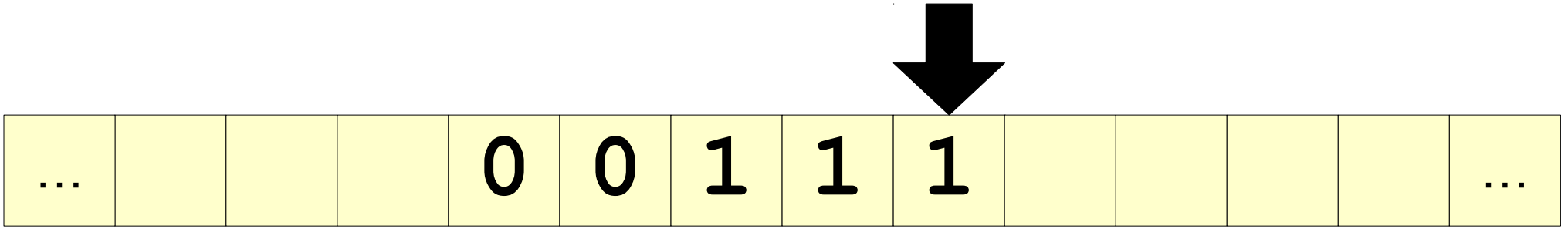
A Sketch of the TM



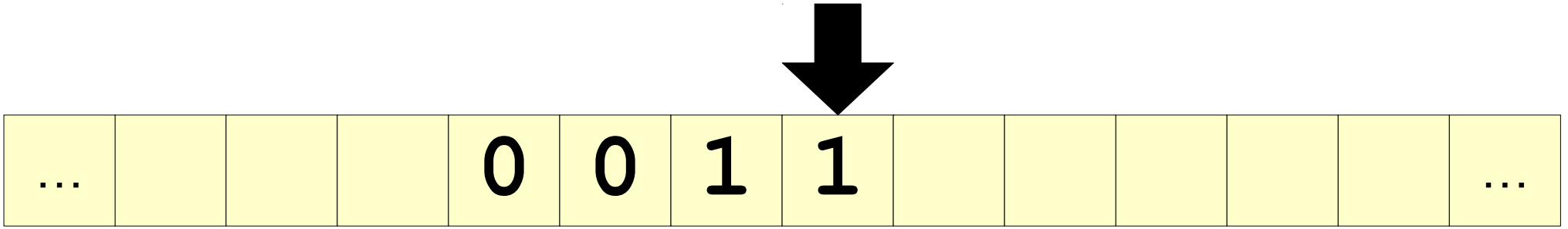
A Sketch of the TM



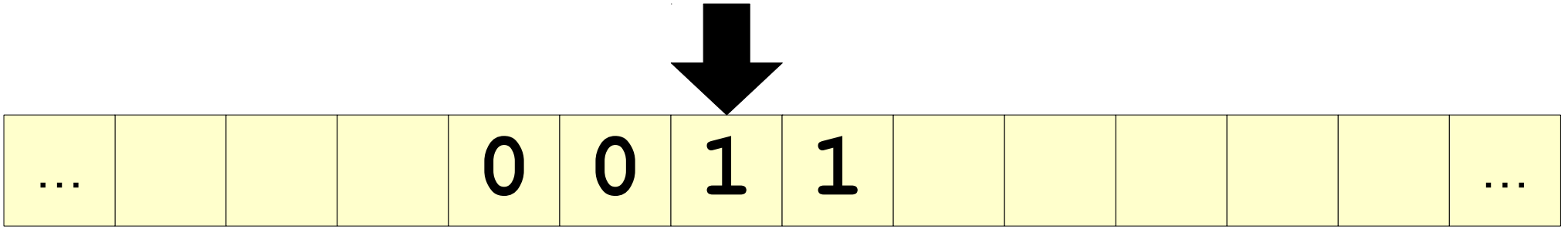
A Sketch of the TM



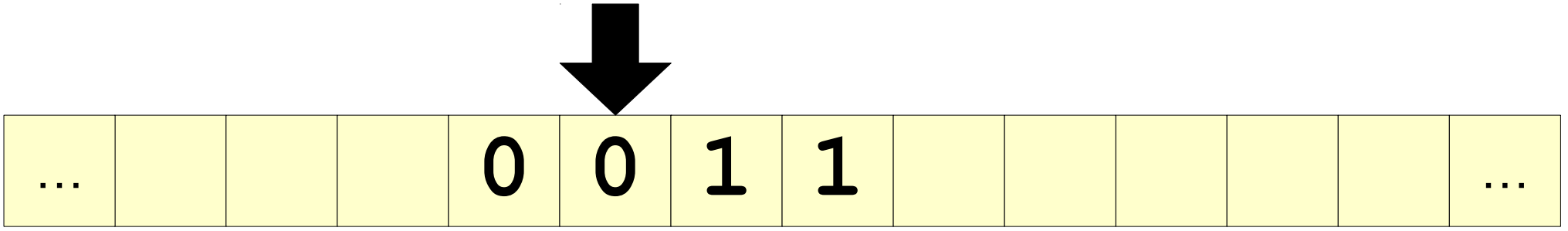
A Sketch of the TM



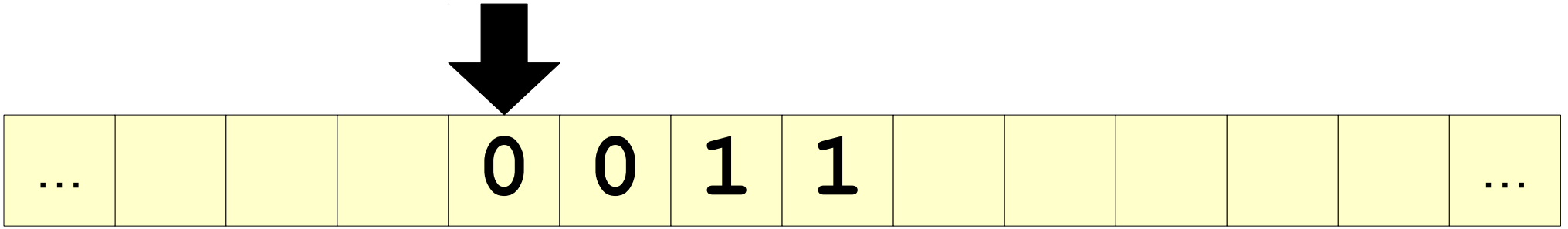
A Sketch of the TM



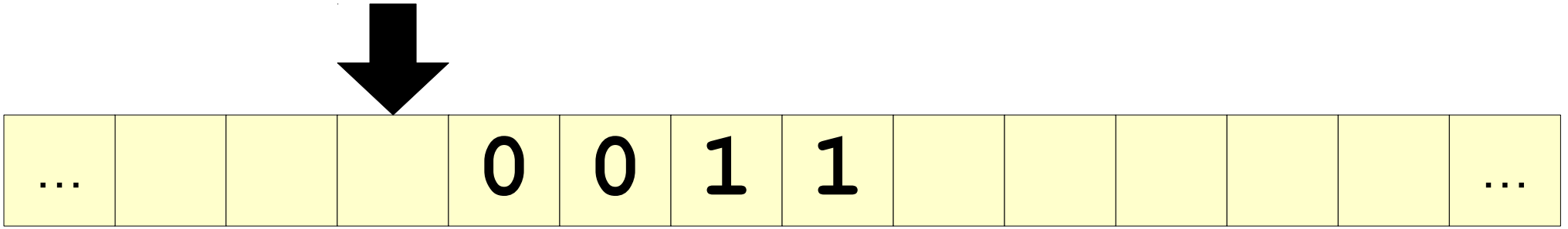
A Sketch of the TM



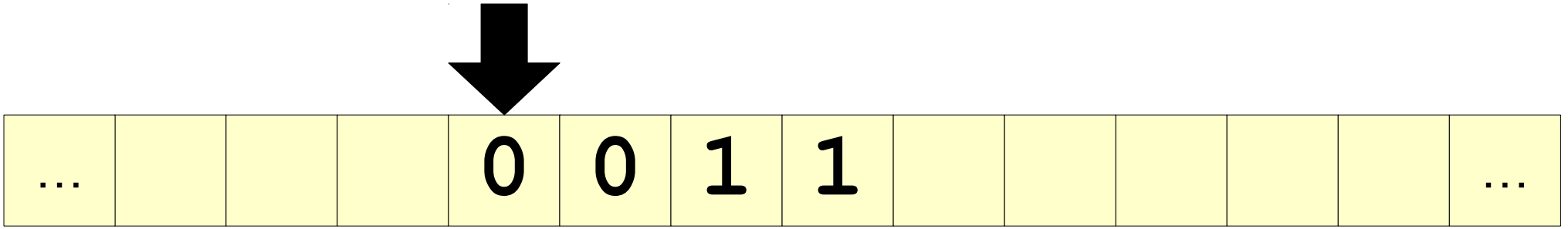
A Sketch of the TM



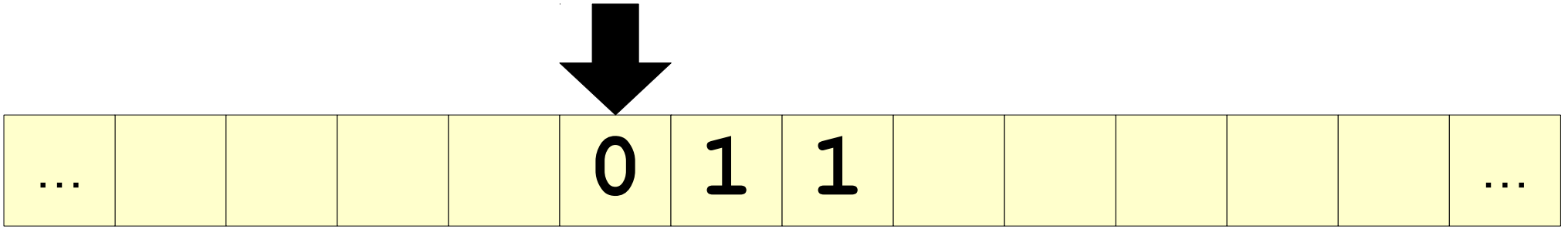
A Sketch of the TM



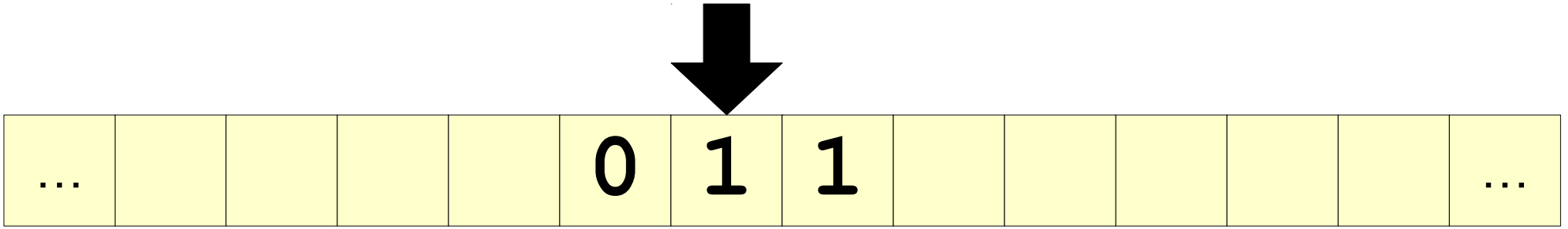
A Sketch of the TM



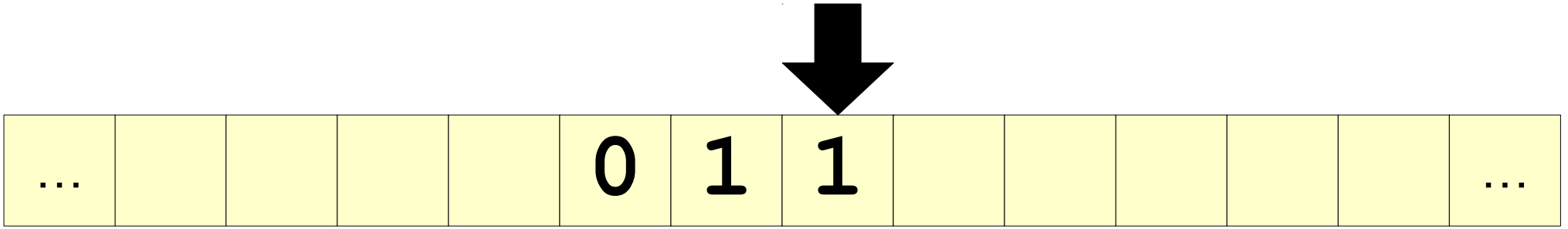
A Sketch of the TM



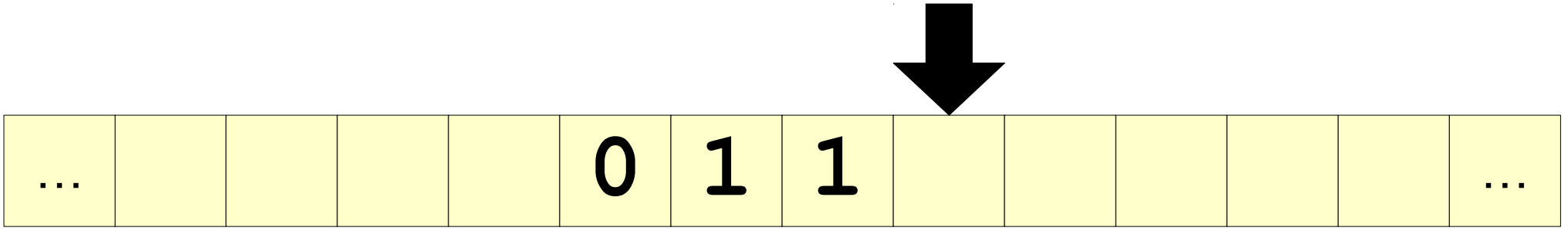
A Sketch of the TM



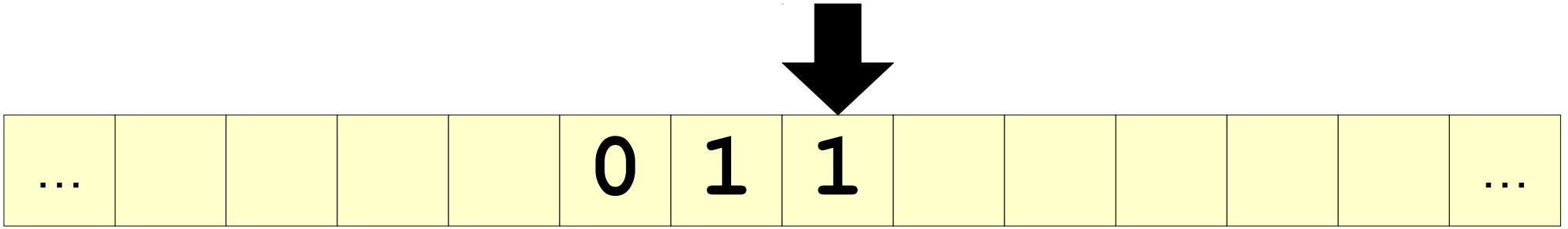
A Sketch of the TM



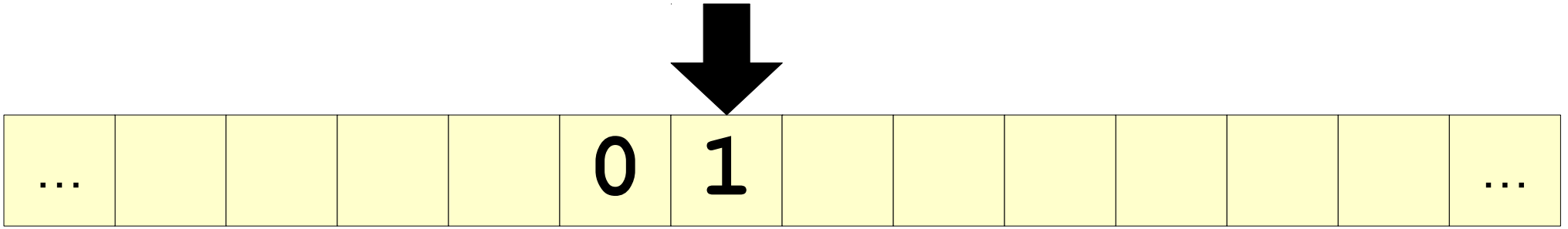
A Sketch of the TM



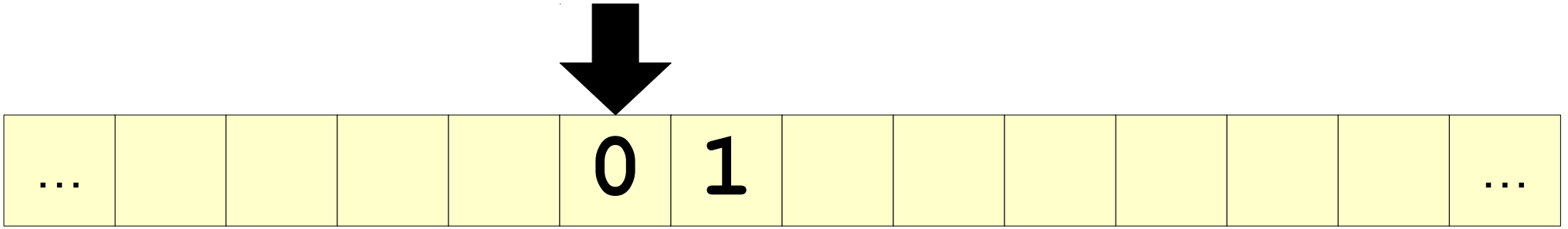
A Sketch of the TM



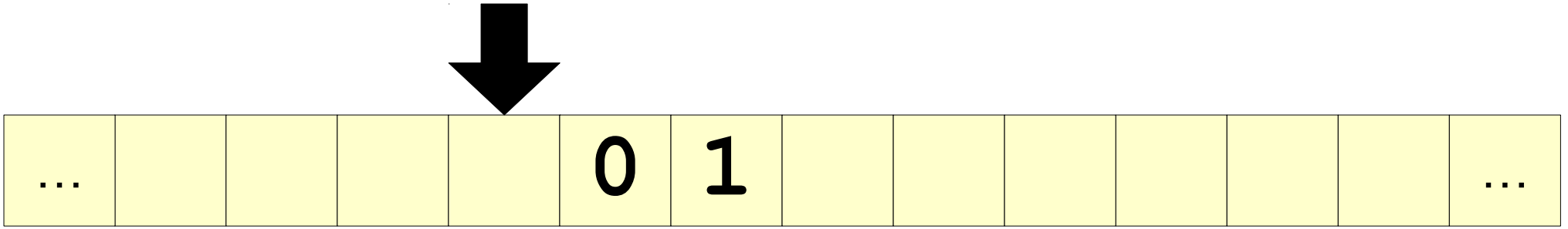
A Sketch of the TM



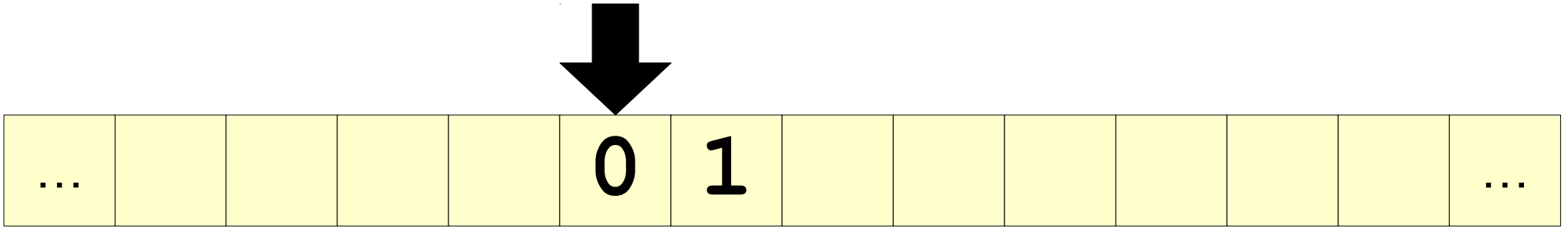
A Sketch of the TM



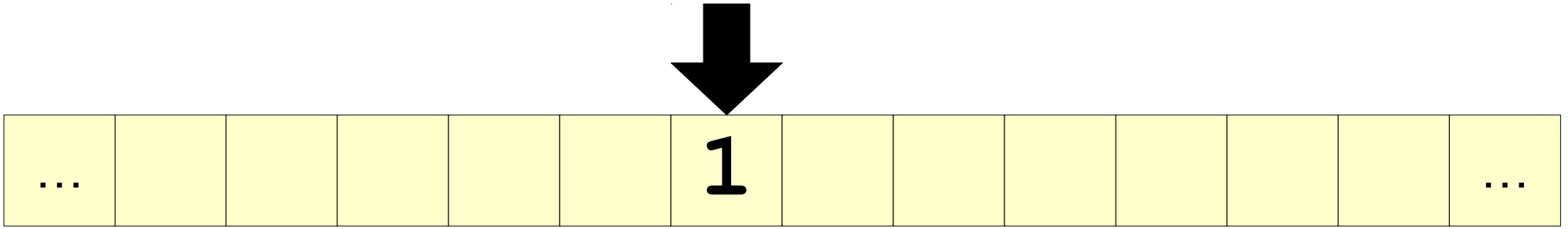
A Sketch of the TM



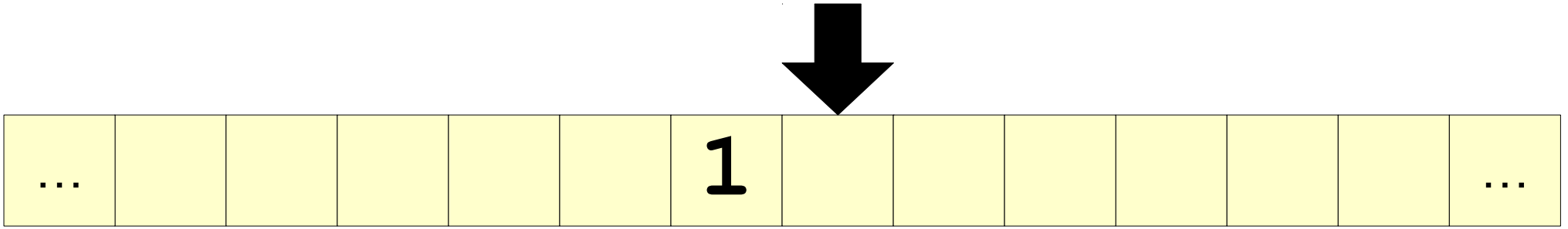
A Sketch of the TM



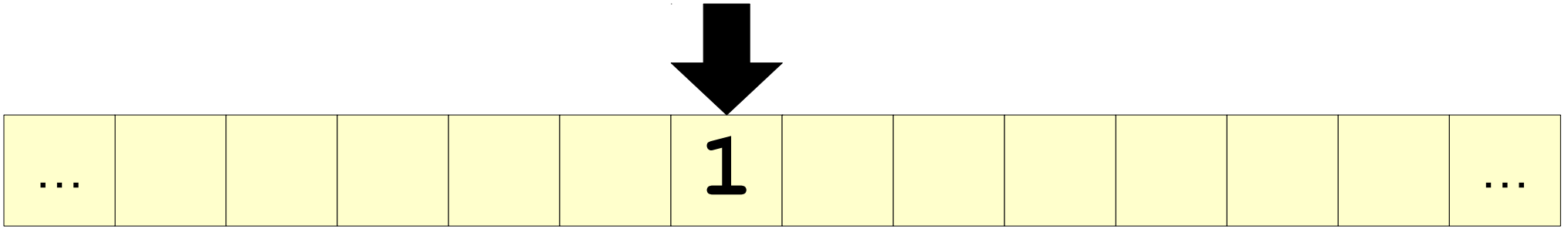
A Sketch of the TM



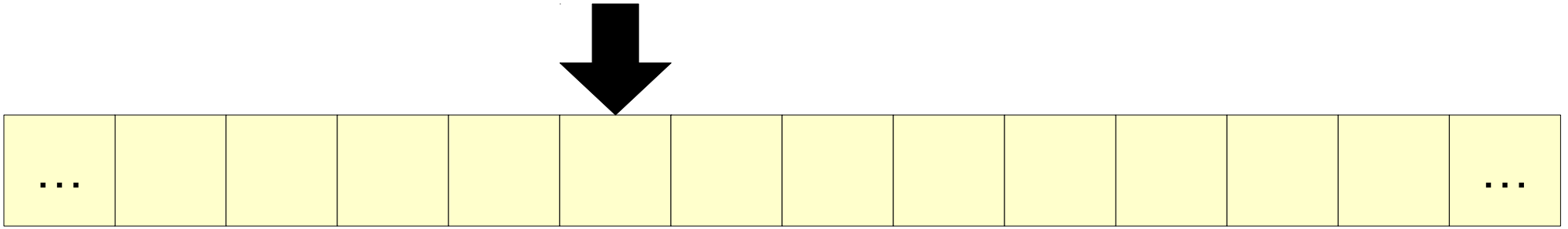
A Sketch of the TM



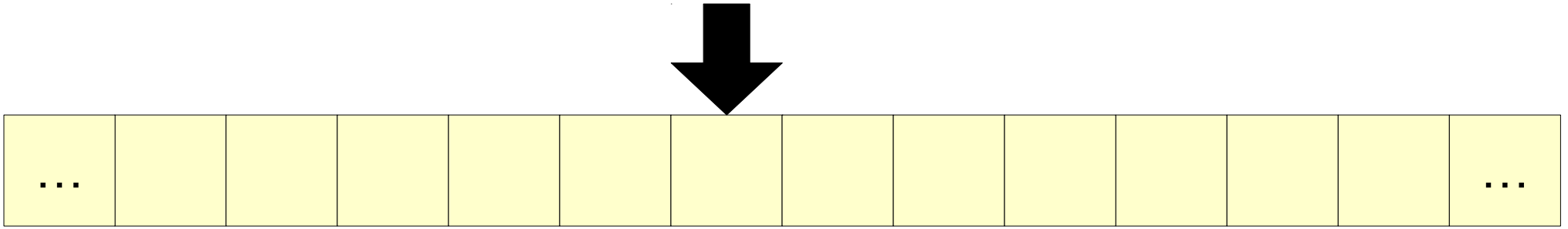
A Sketch of the TM

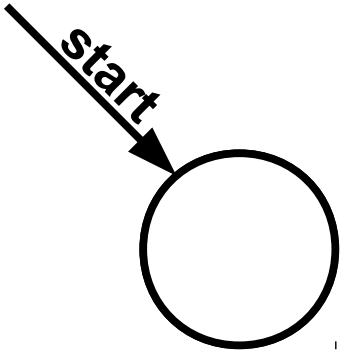


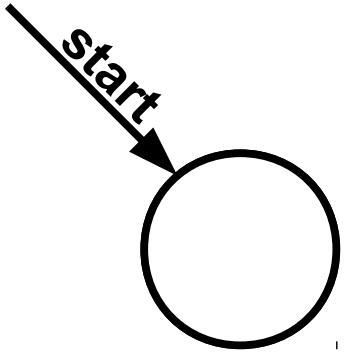
A Sketch of the TM

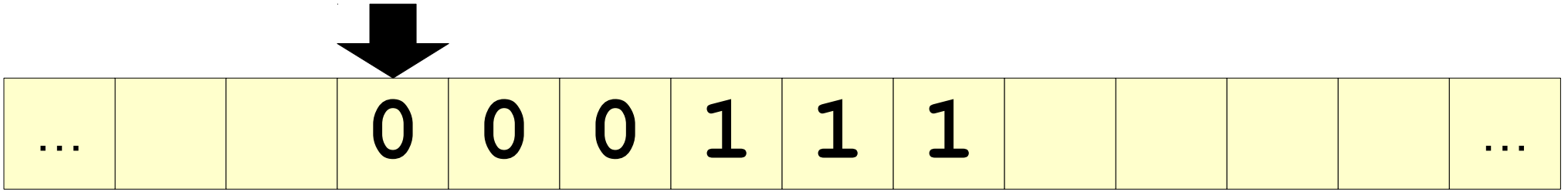


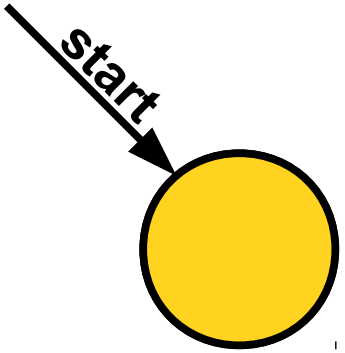
A Sketch of the TM

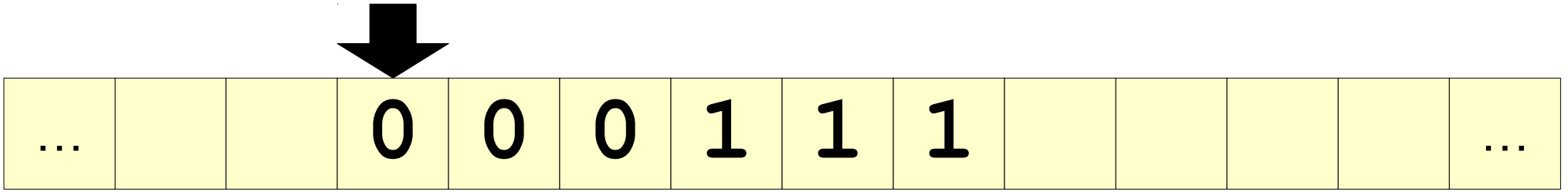


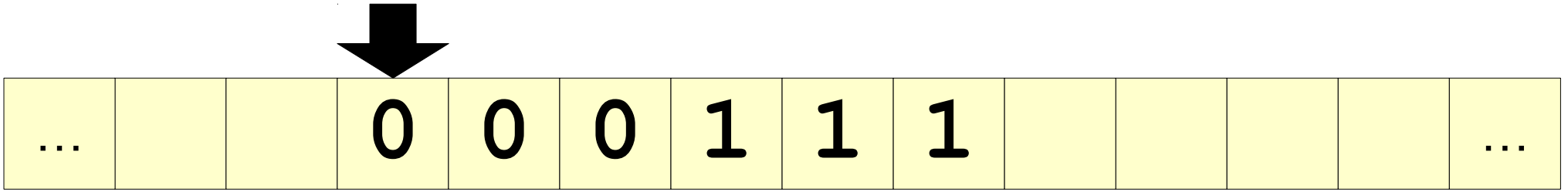

start

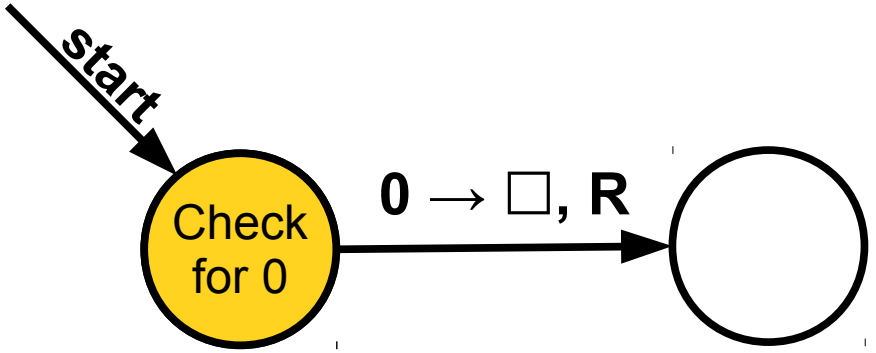

start

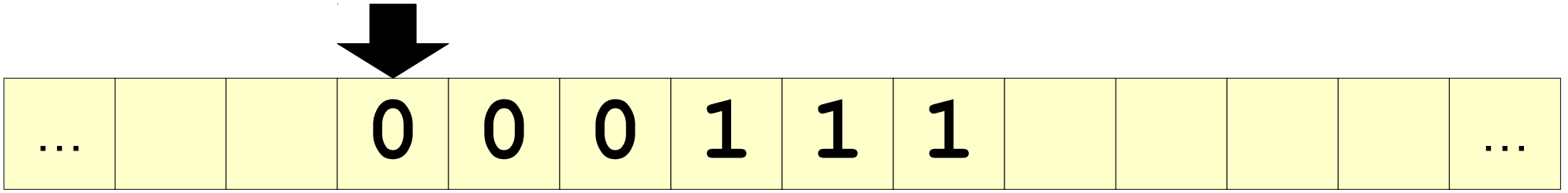

...
0 0 0 1 1 1
...

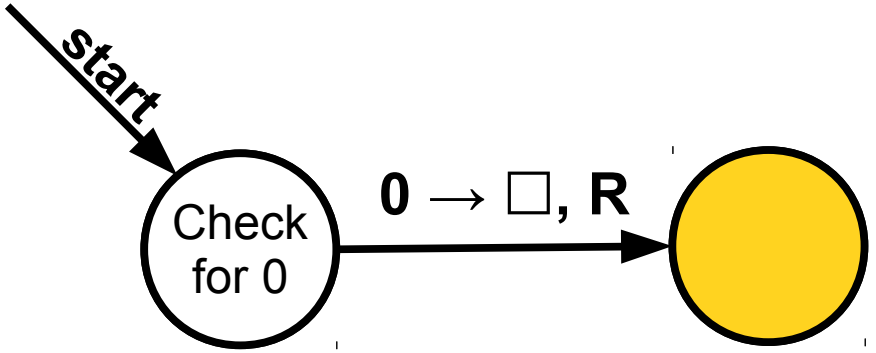

start

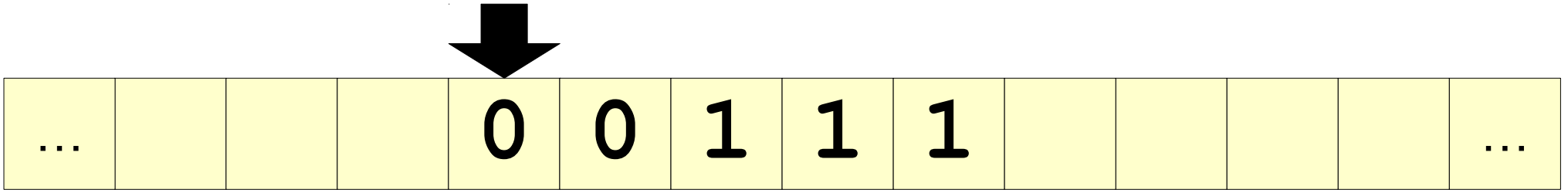

...
0 0 0 1 1 1
...

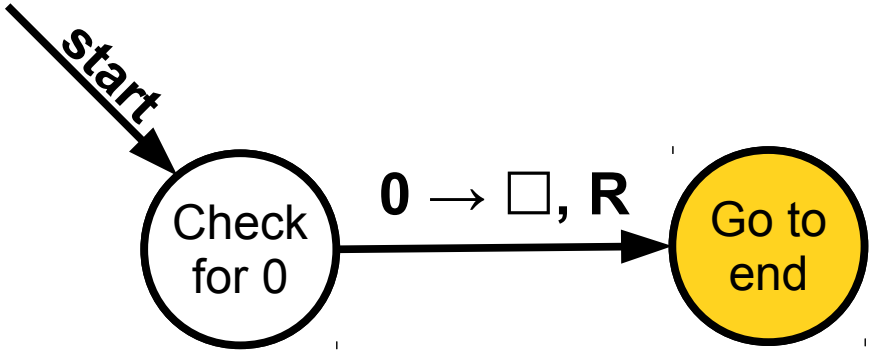

start
Check
for 0
...
0 0 0 1 1 1
...

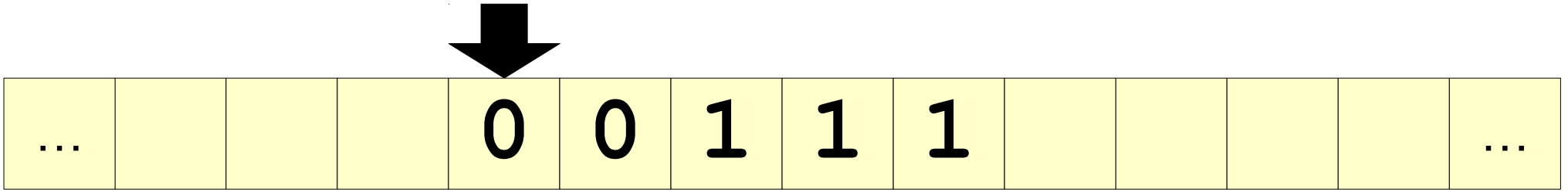

start
Check
for 0
0 → □, R

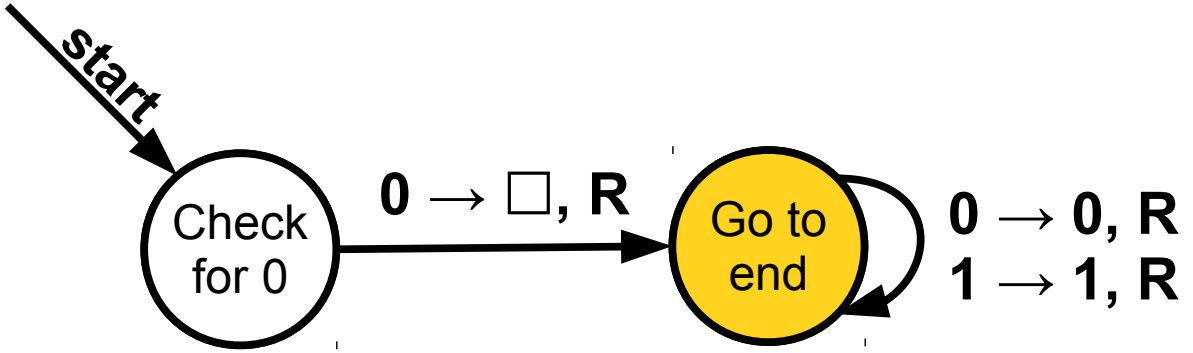

…
0 0 0 1 1 1
…

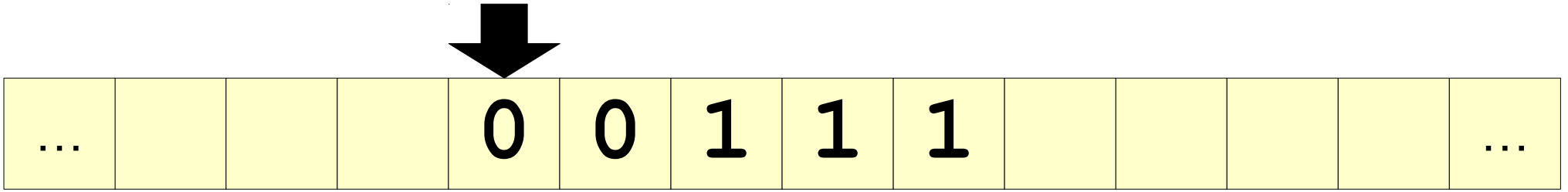

start
Check
for 0
0 → □, R

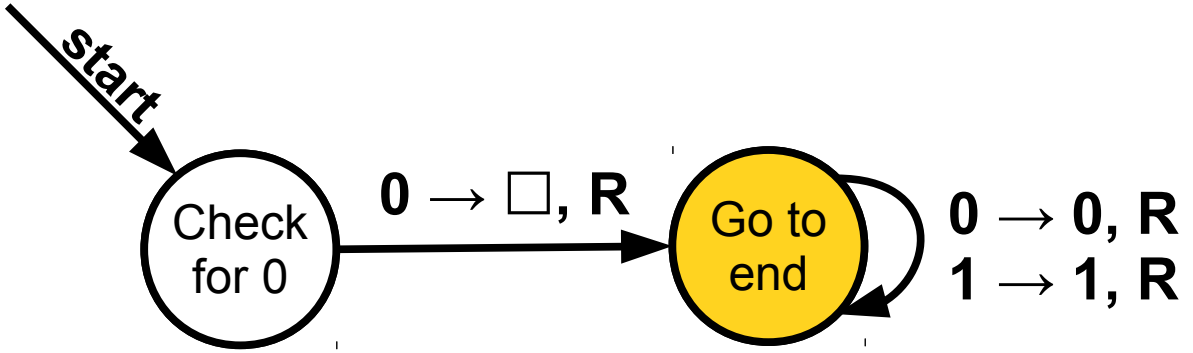

...
0
0
1
1
1
...

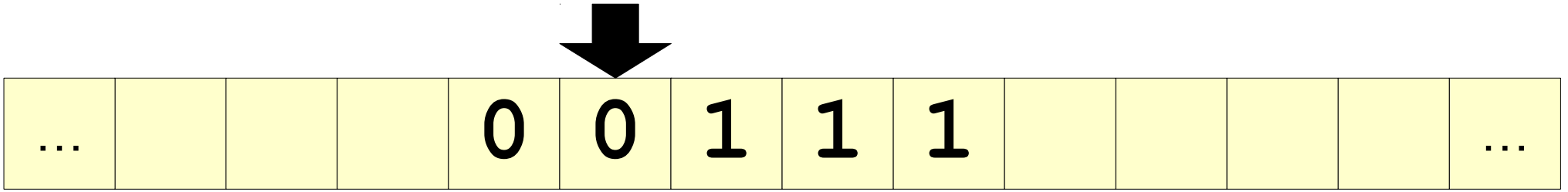

start
Check for 0
0 → □, R
Go to end

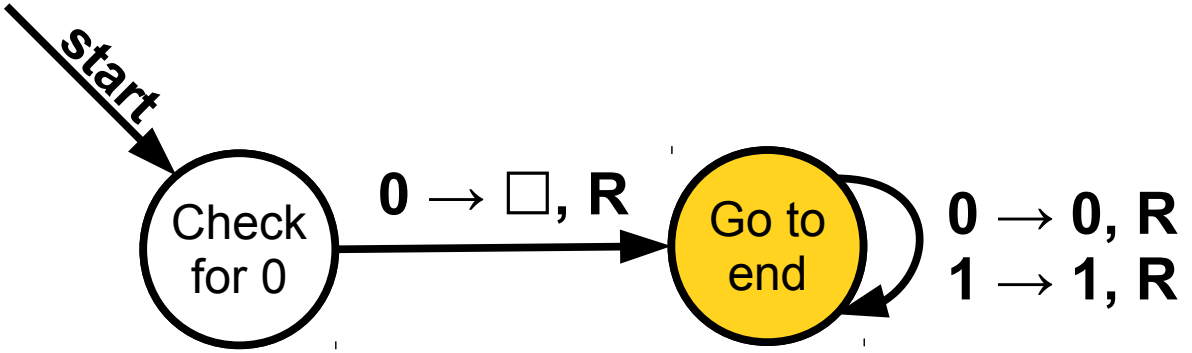

...
0
0
1
1
1
...

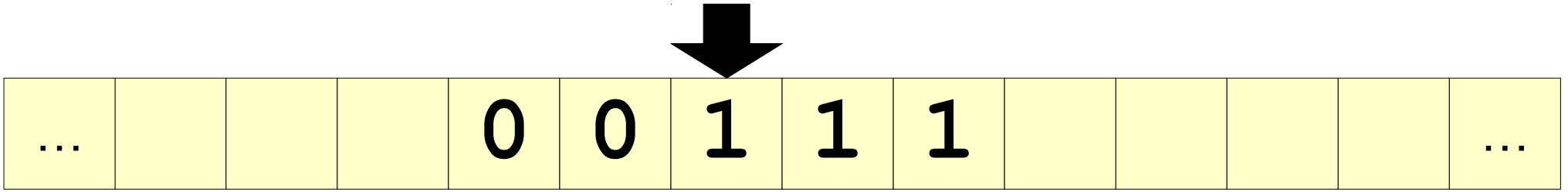

start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

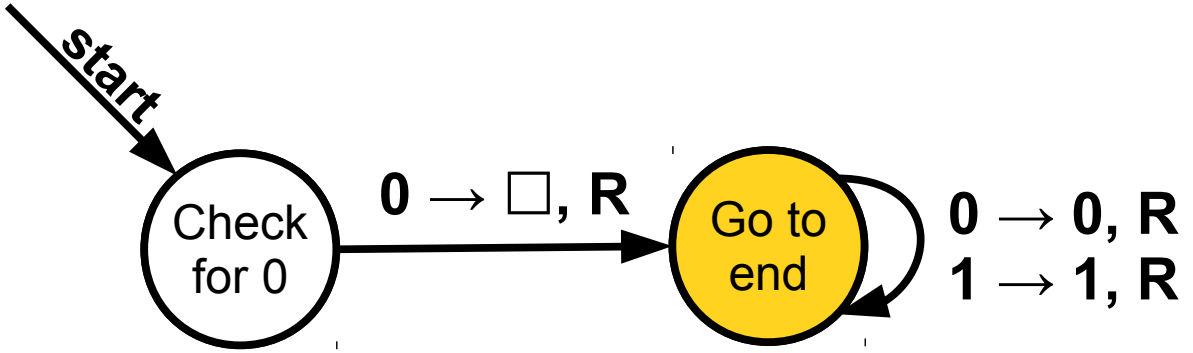

...
0 0 1 1 1
...

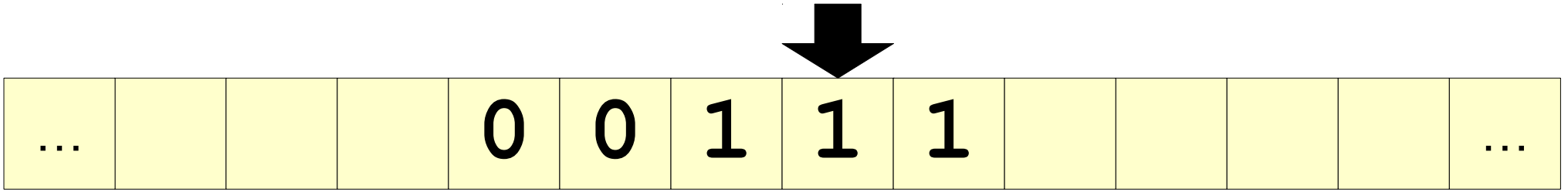

start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

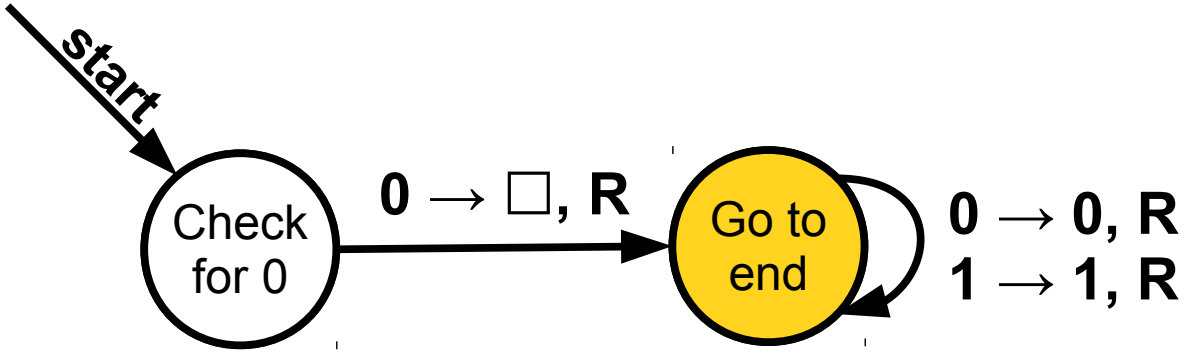

...
0 0 1 1 1
...

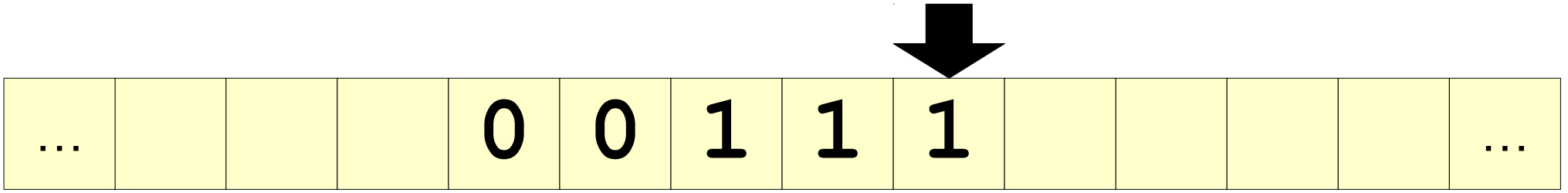

start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

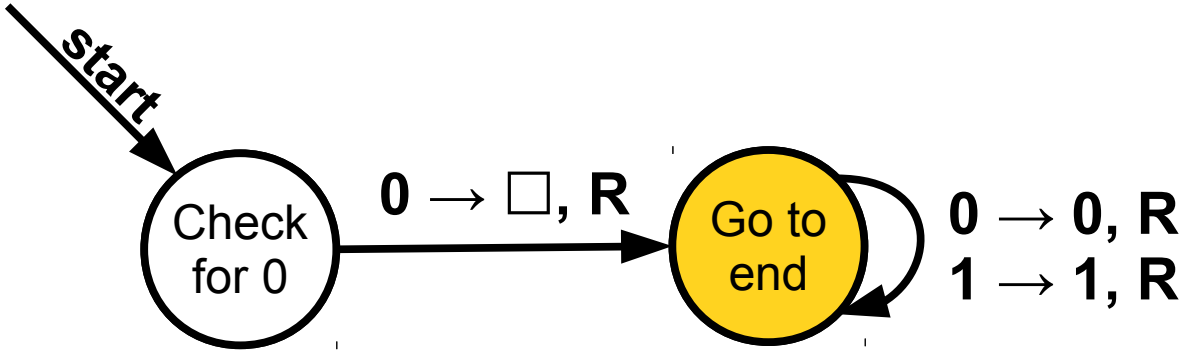

...
0 0 1 1 1
...

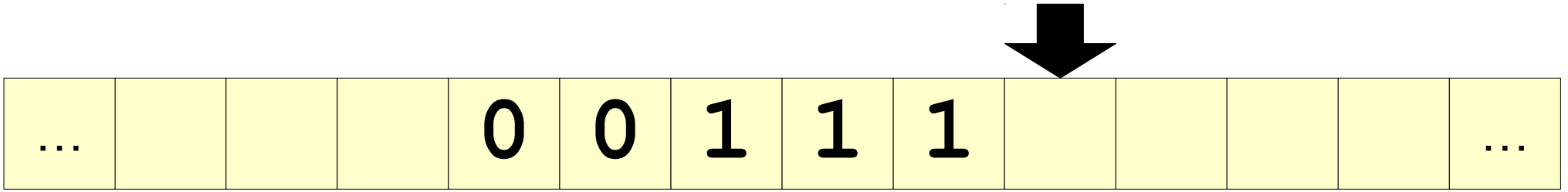

start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

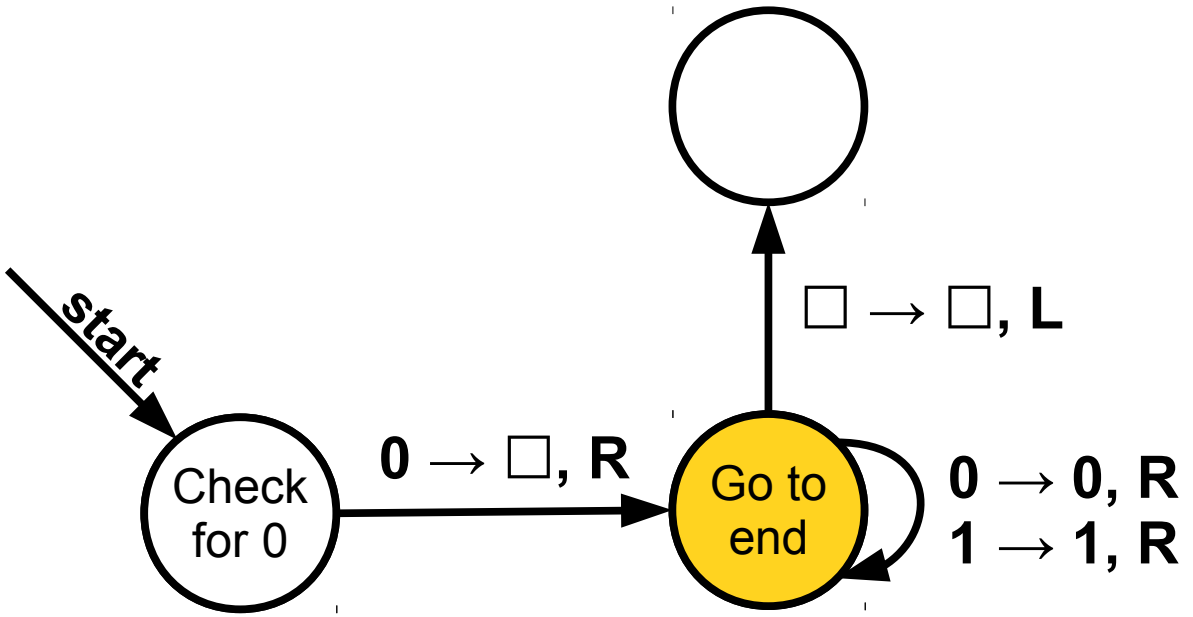

...
0 0 1 1 1
...

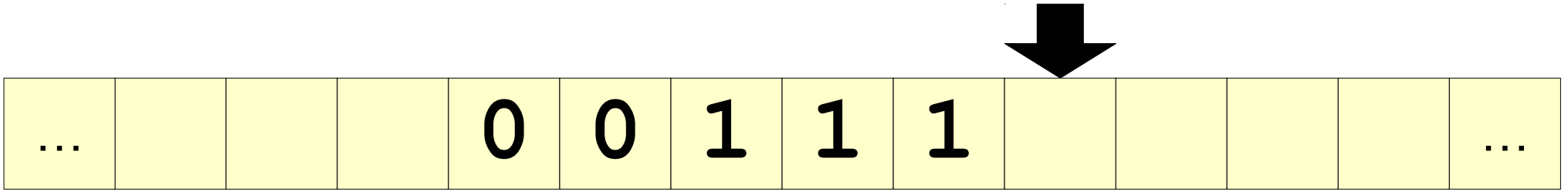

start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

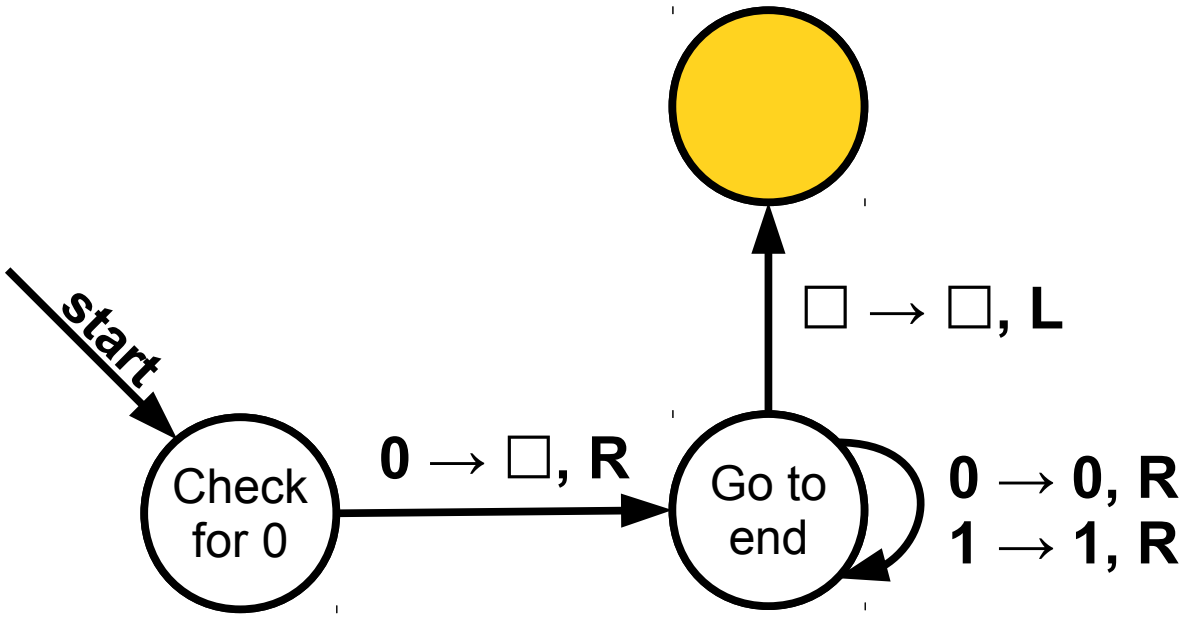

...
0 0 1 1 1
...

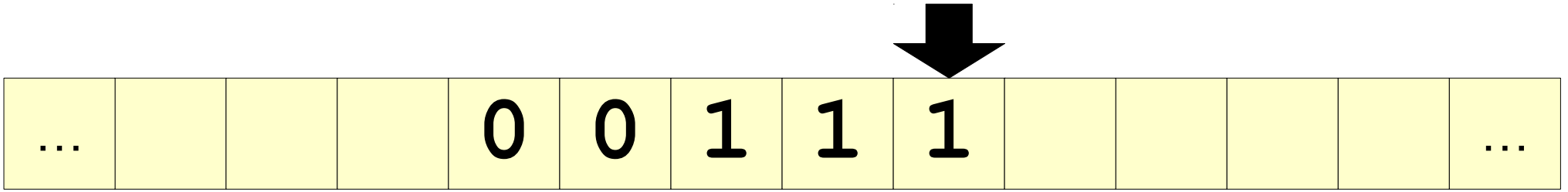

start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

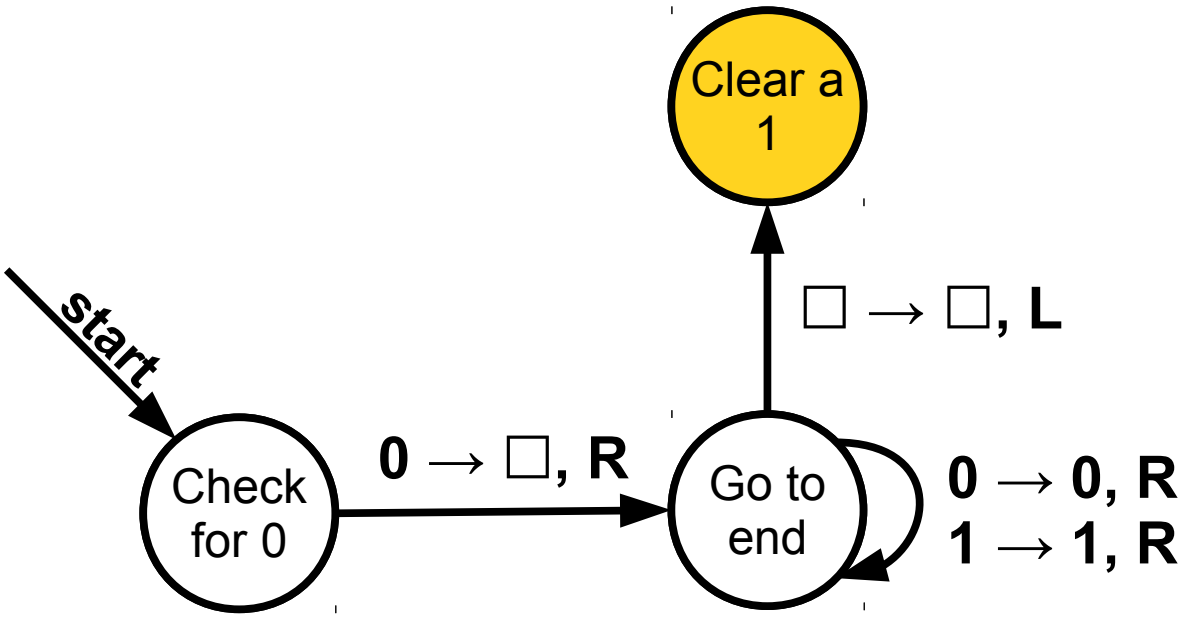

...
0 0 1 1 1
...

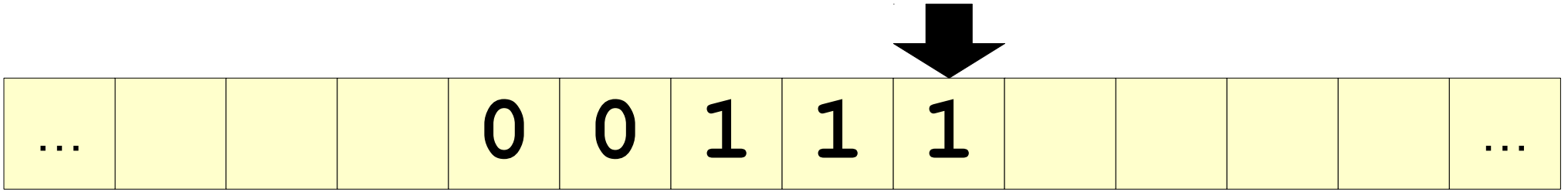

start
Check for 0
0 → □, R
Go to end
□ → □, L
0 → 0, R
1 → 1, R

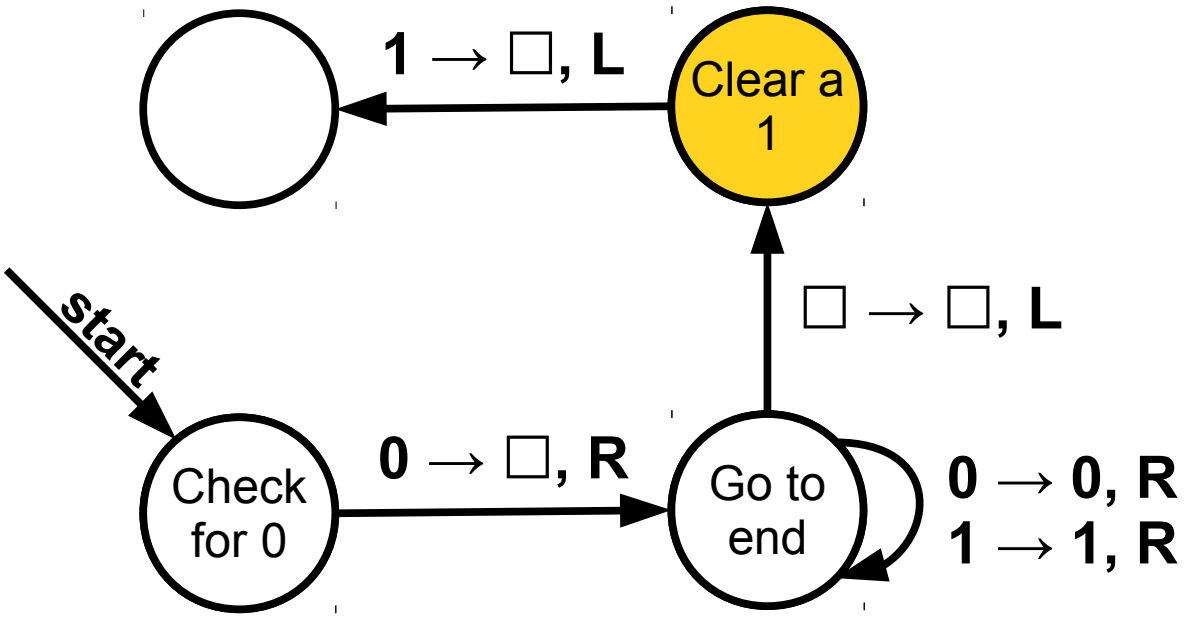

...
0 0 1 1 1
...

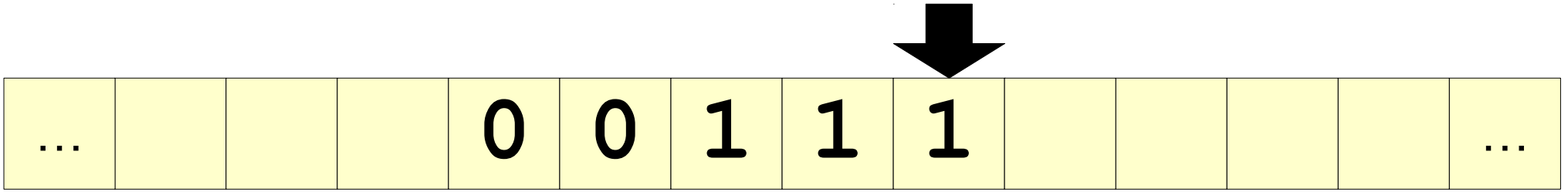

start
Check for 0
0 → □, R
Go to end
□ → □, L
0 → 0, R
1 → 1, R

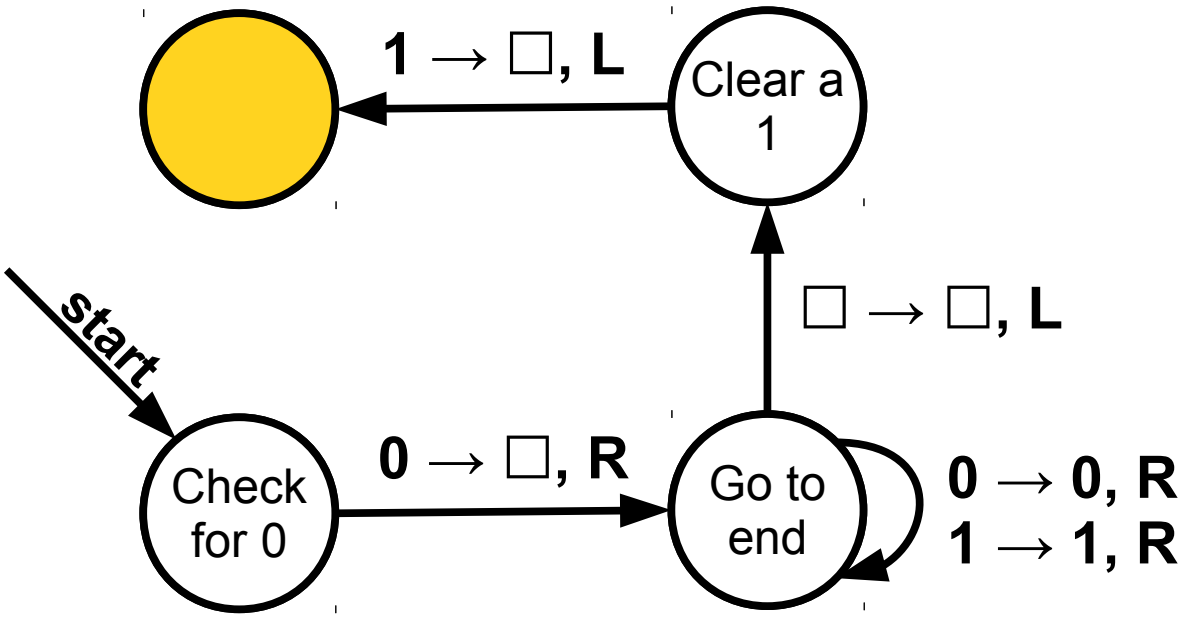

...
0 0 1 1 1
...

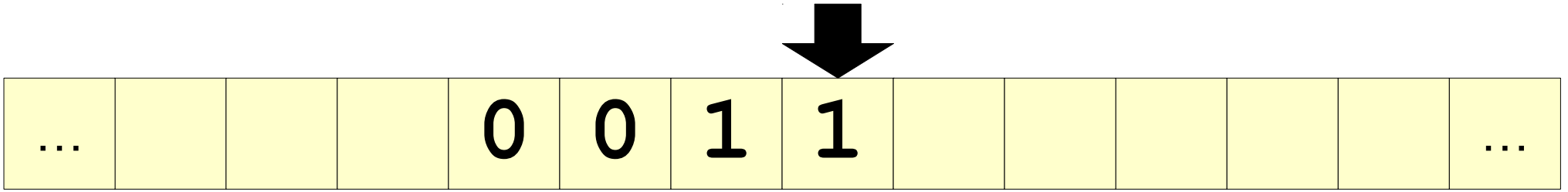

Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

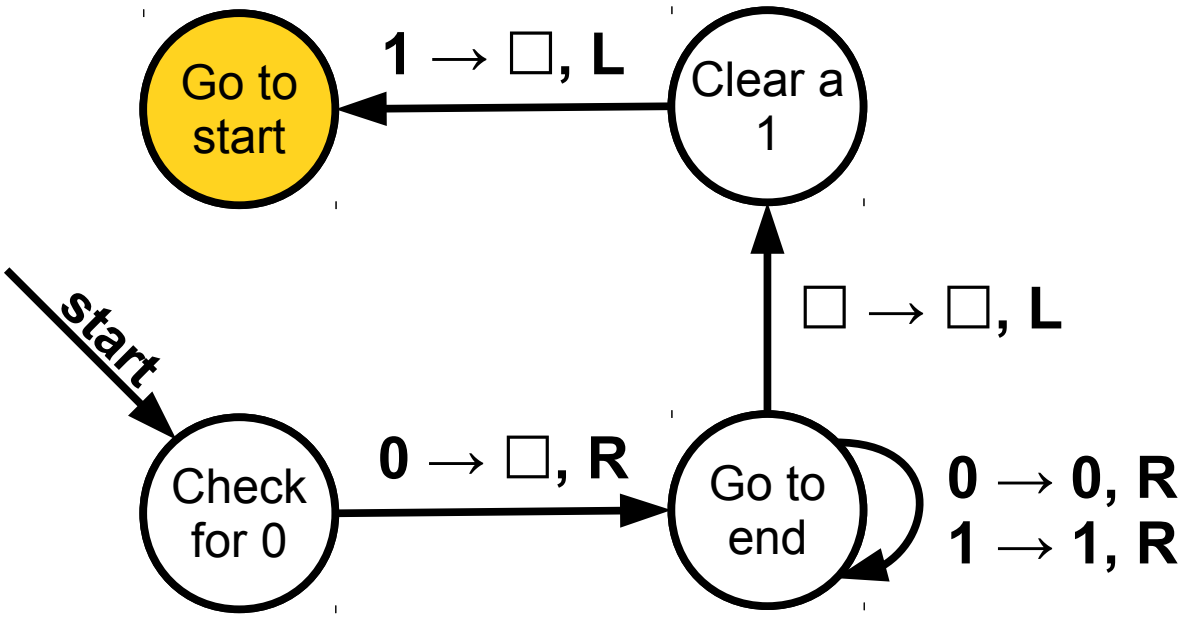

...
0 0 1 1 1
...

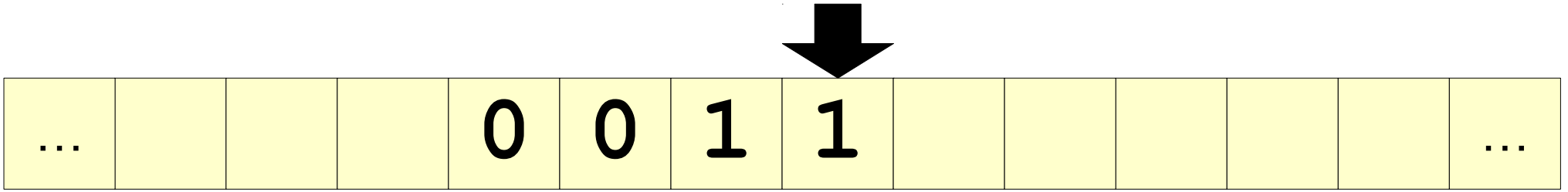

1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

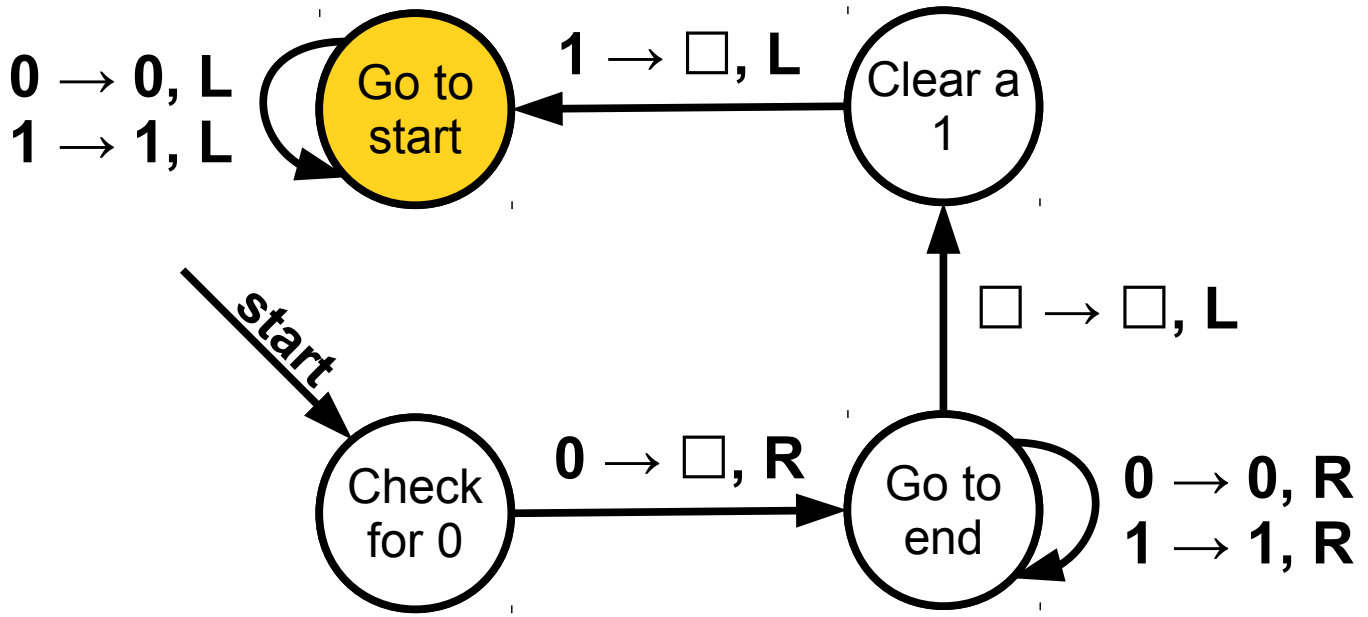

...
0 0 1 1 1
...

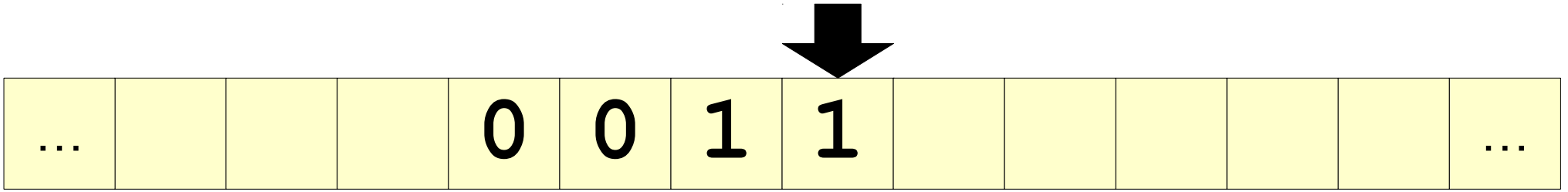

1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

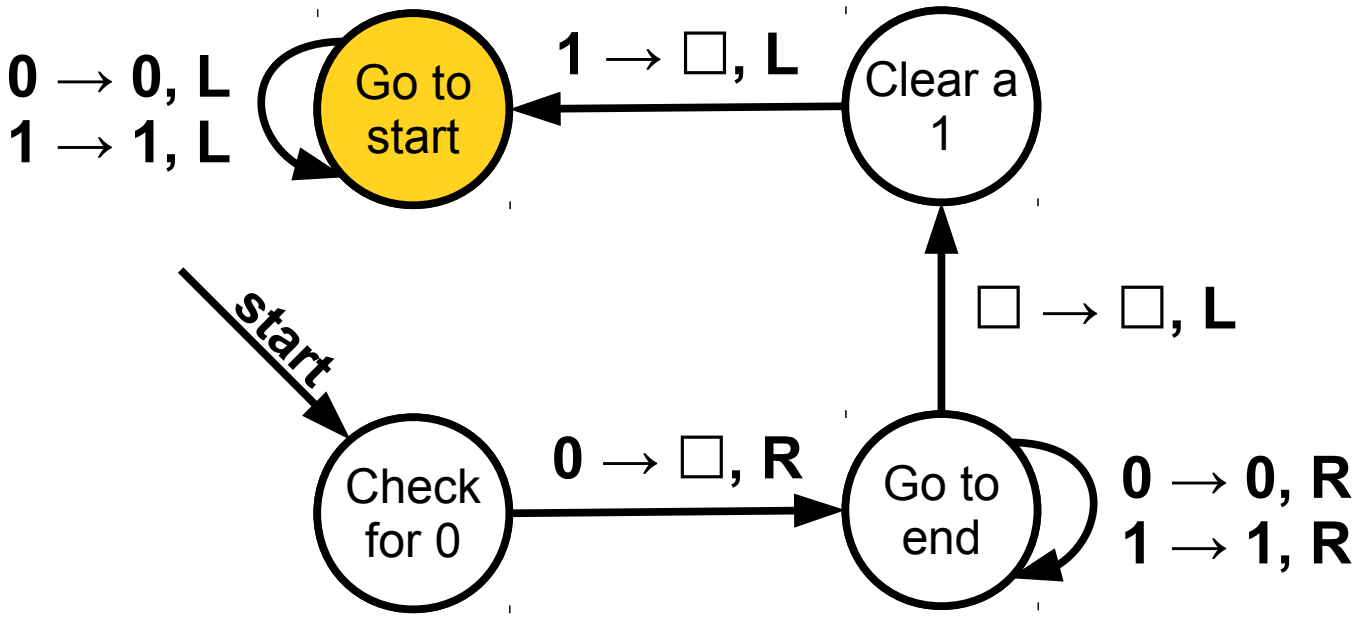

...
0 0 1 1
...

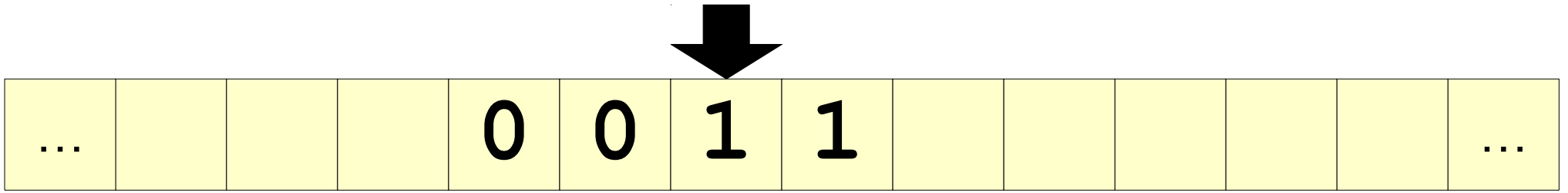

Go to start
1 → □, L
Clear a 1
□ → □, L
start
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

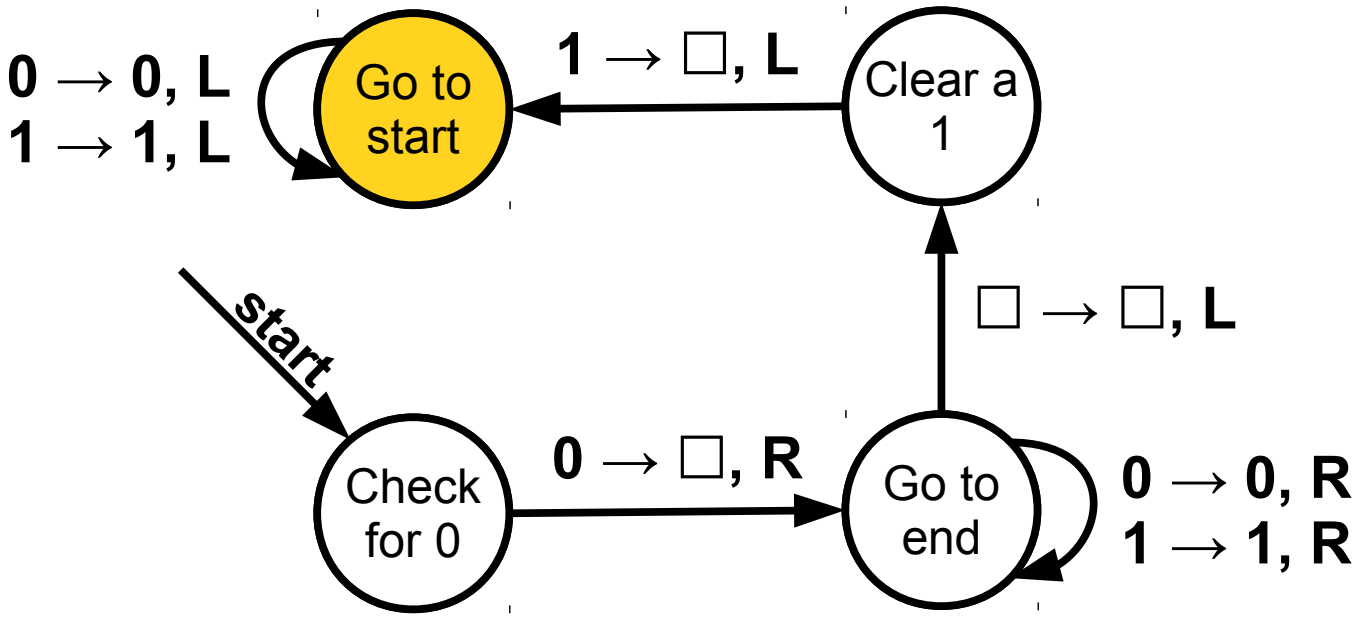

...
0 0 1 1
...

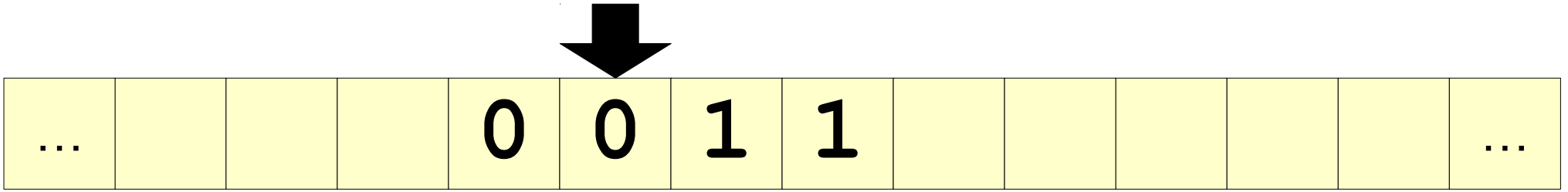

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

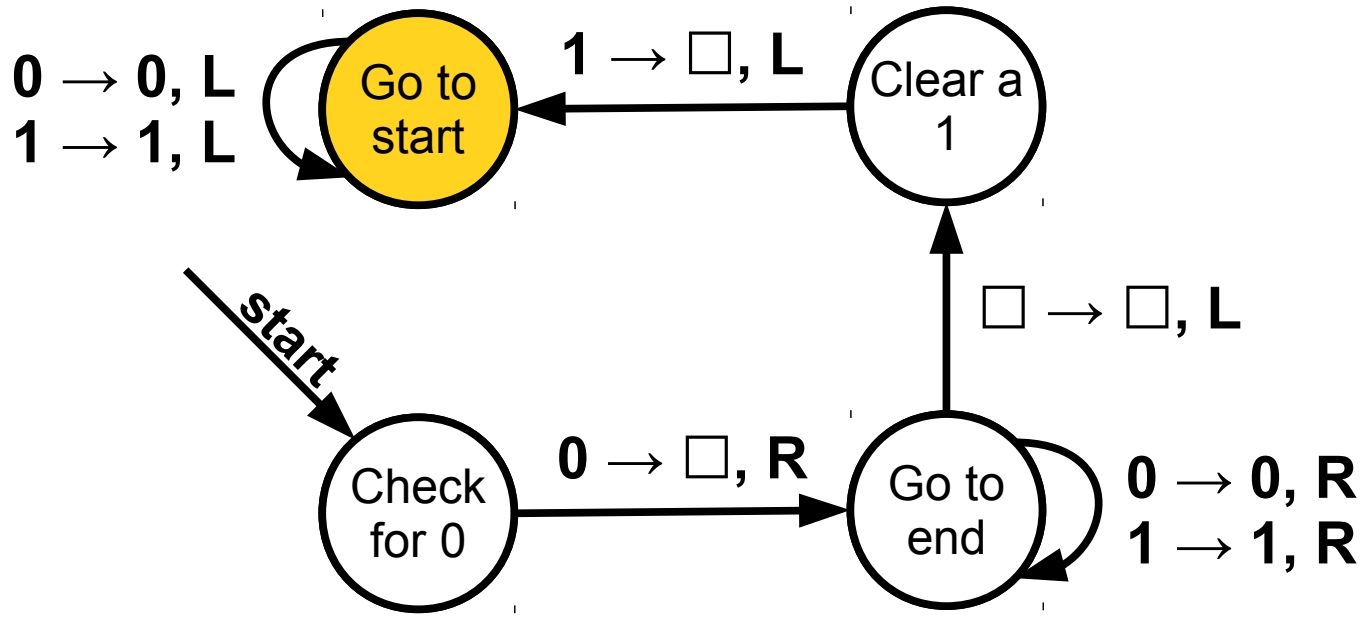

…
0
0
1
1
…

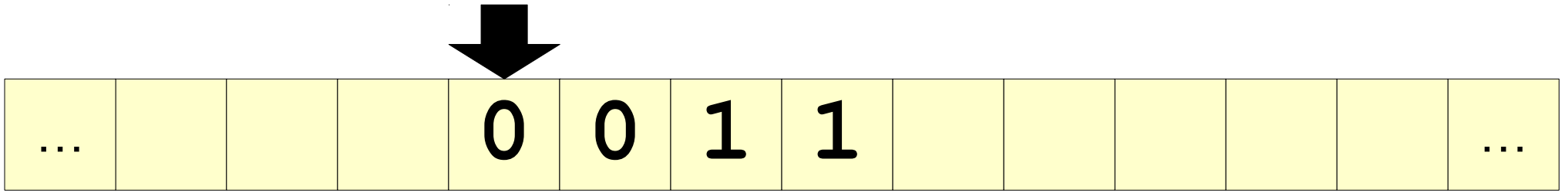

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

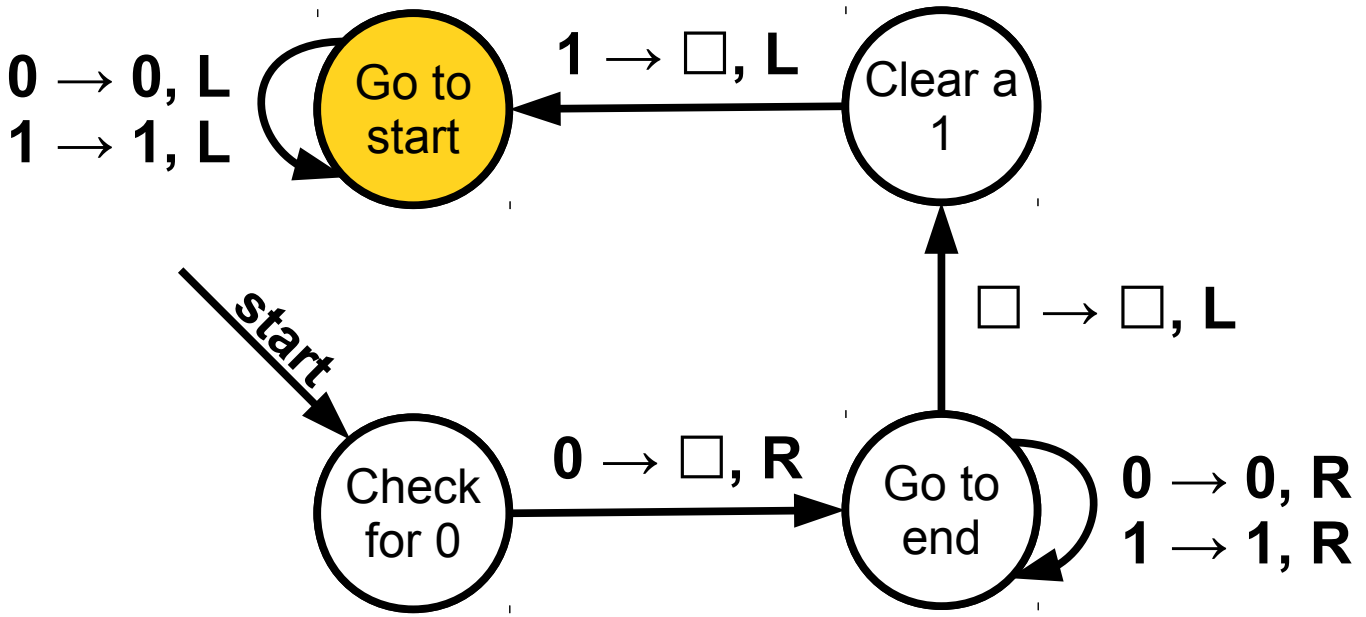

…
0
0
1
1
…

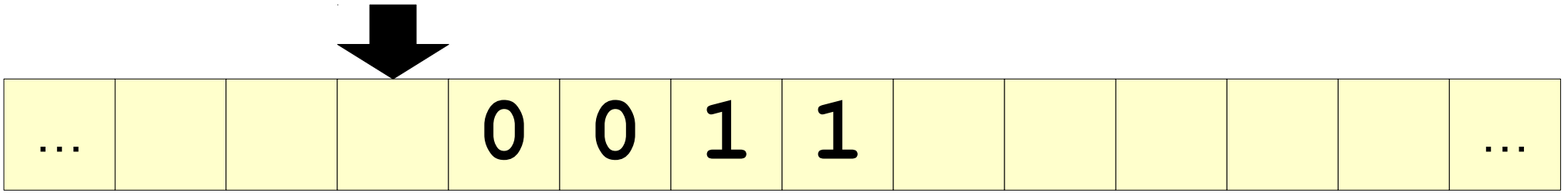

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

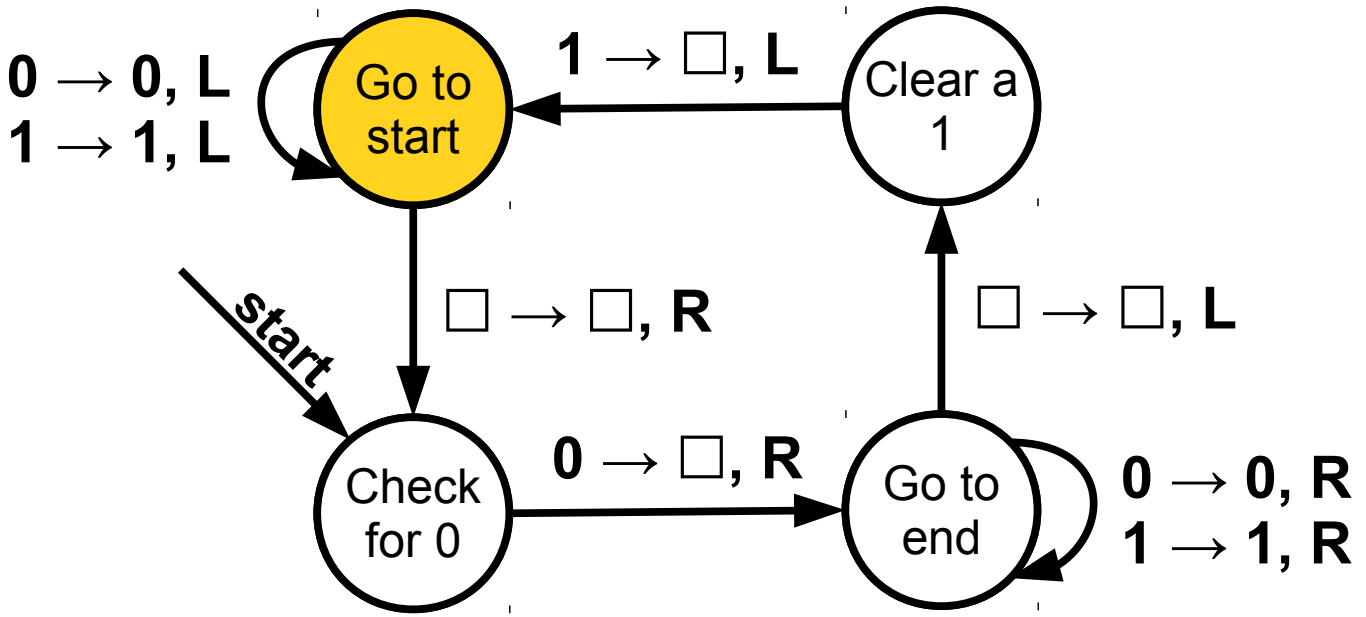

…
0
0
1
1
…

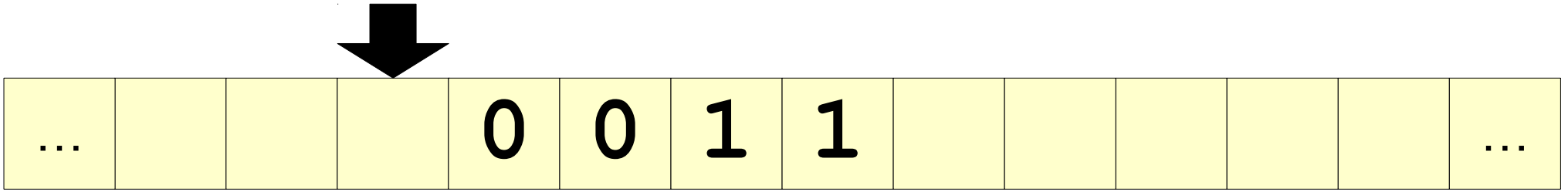

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

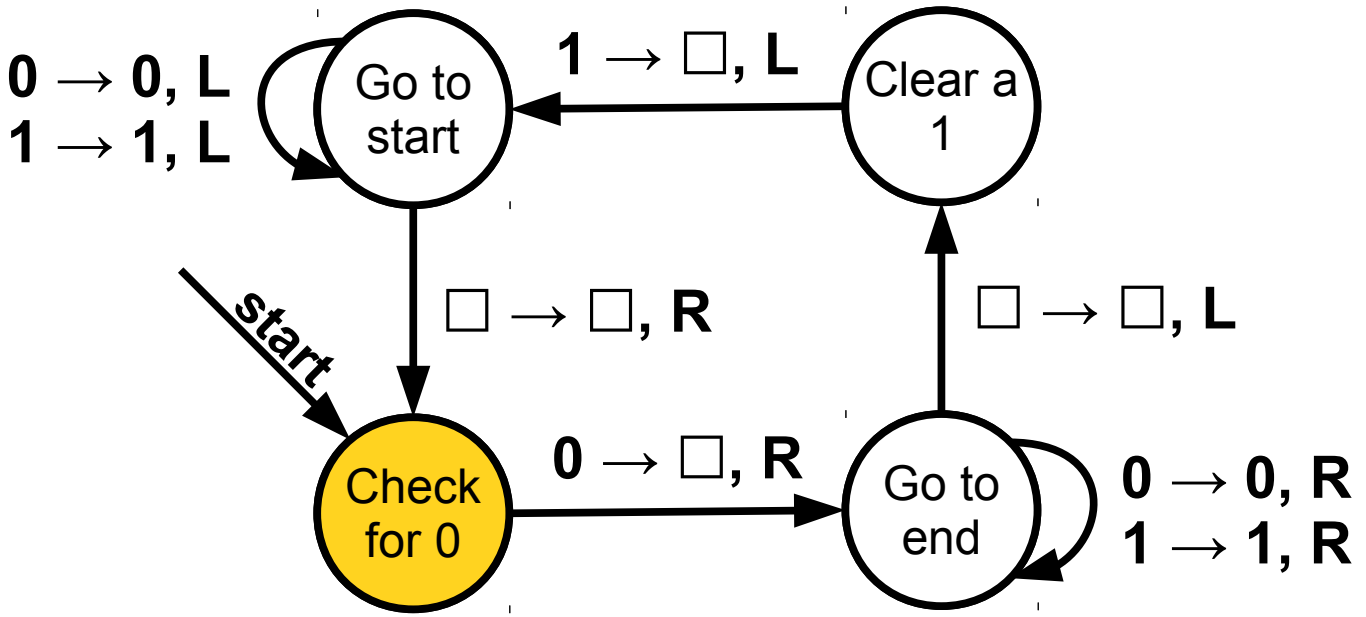

…
0
0
1
1
…

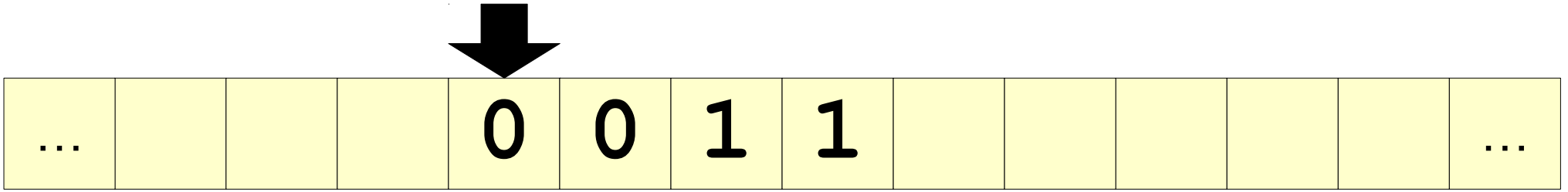

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

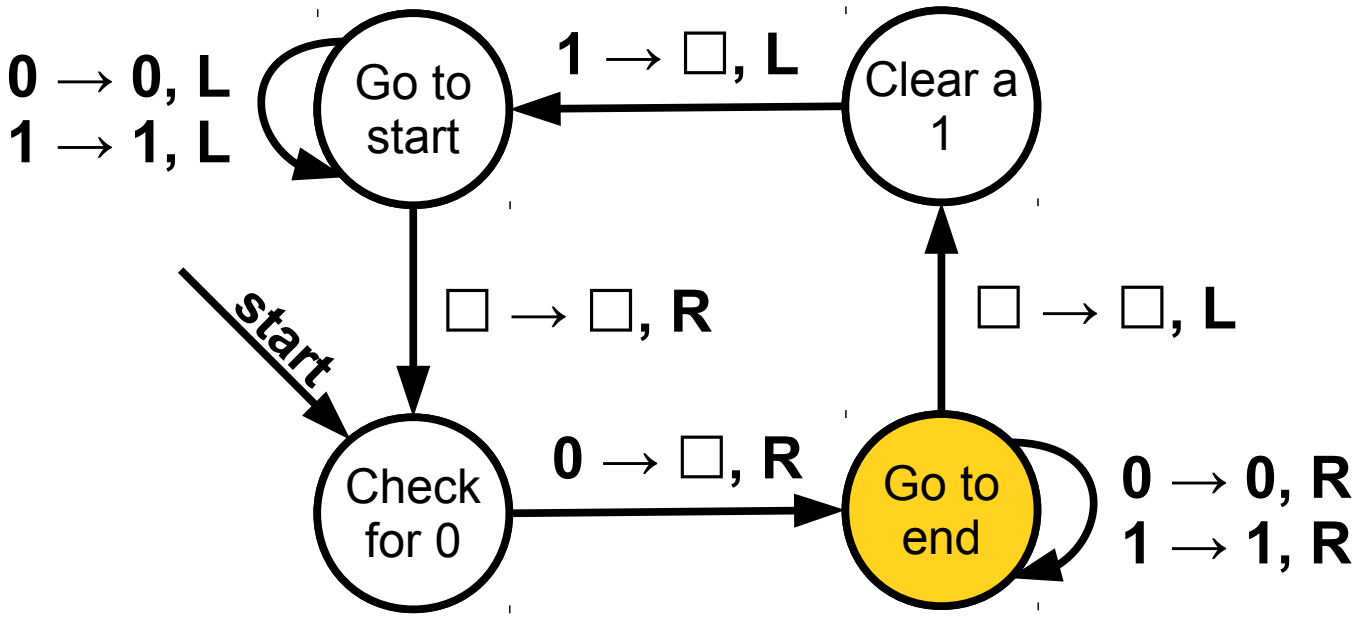

…
0
0
1
1
…

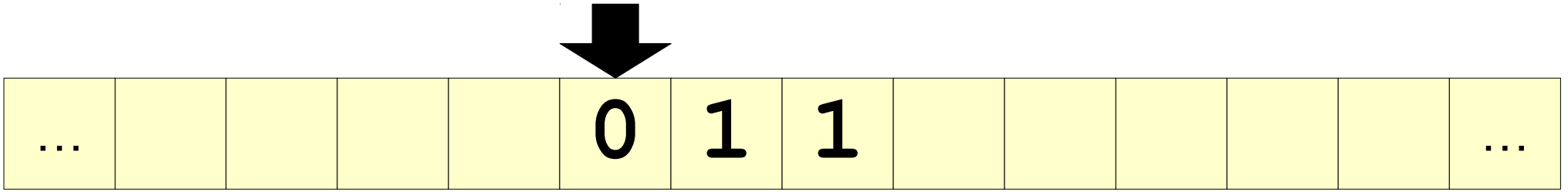

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

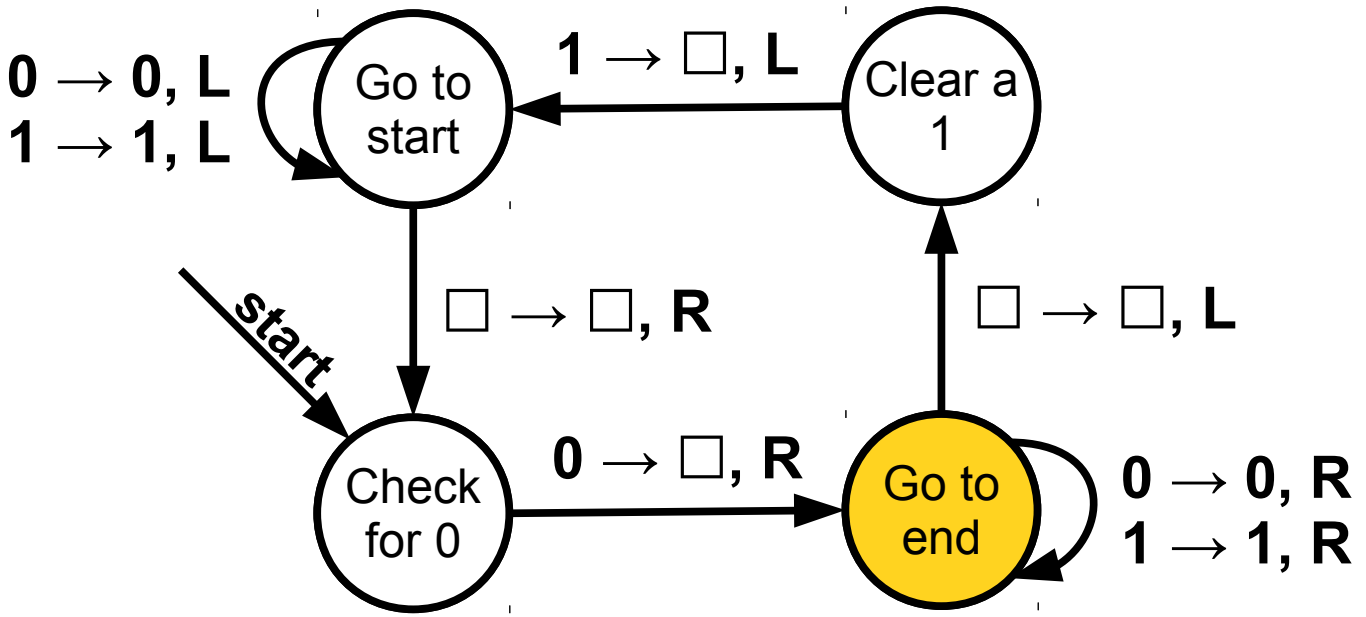

…
0 0 1 1
…

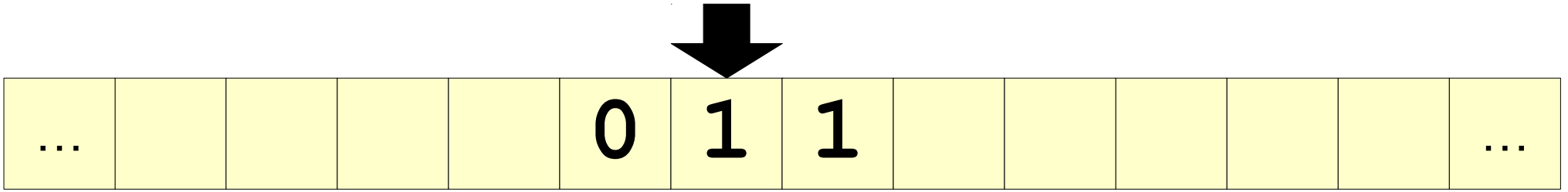

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

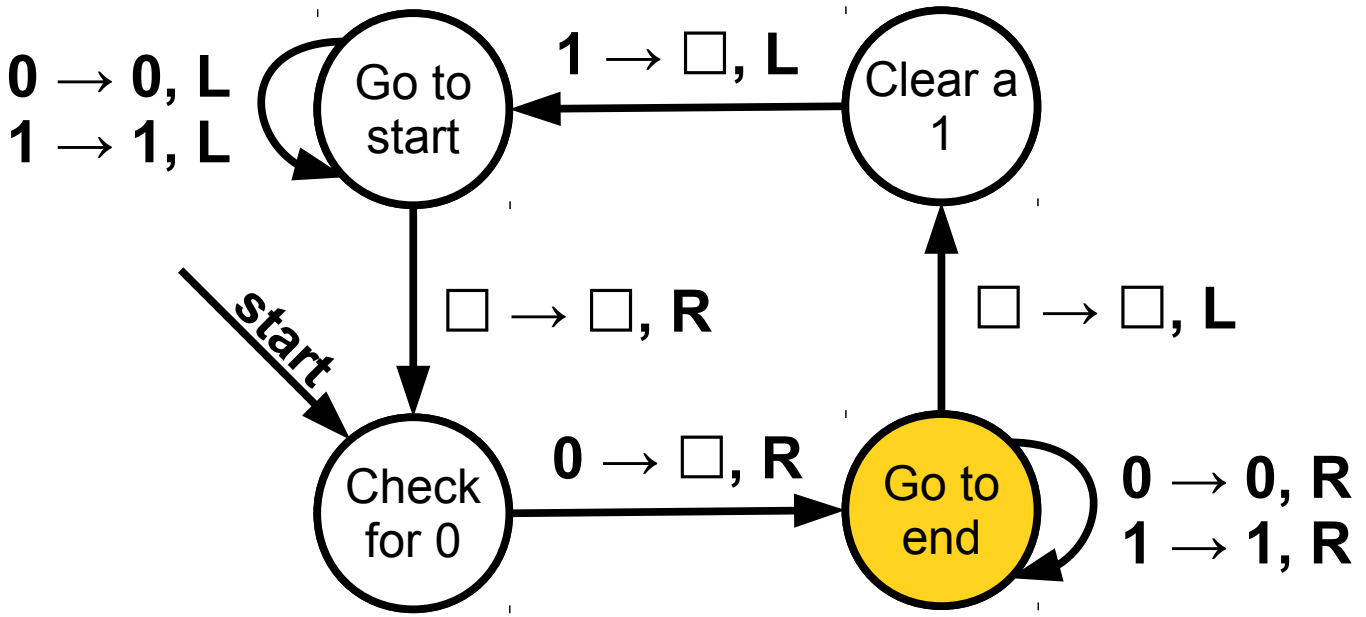

…
0
0
1
1
…

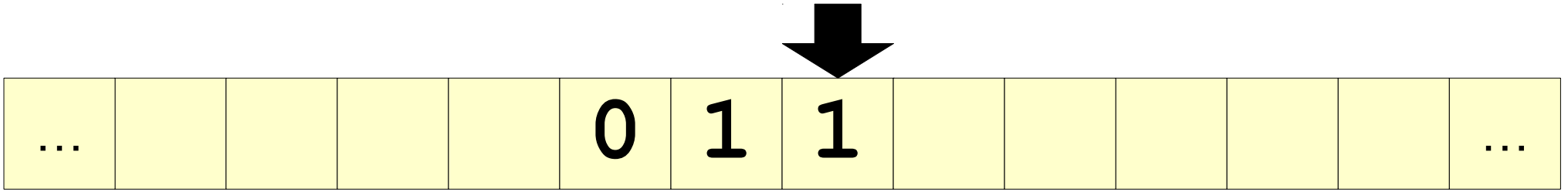

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

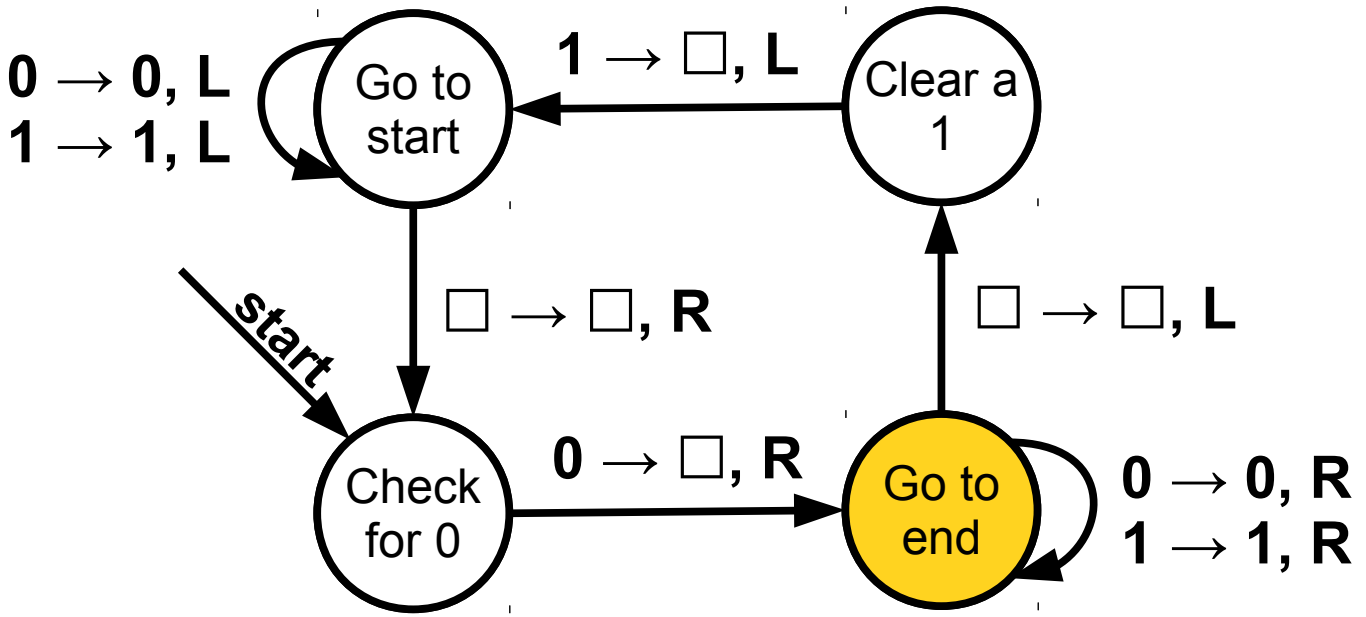

…
0
1
1
…

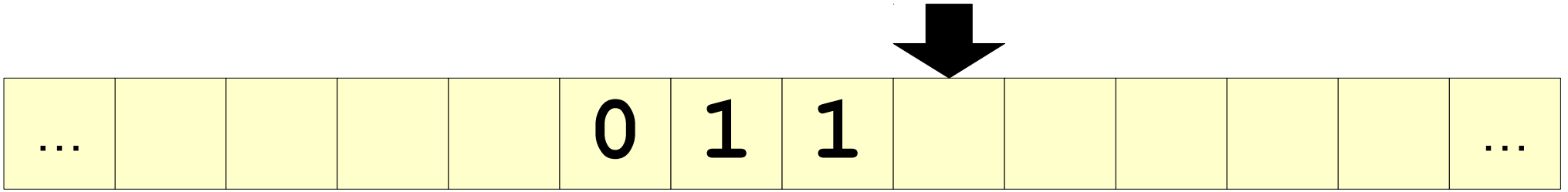

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

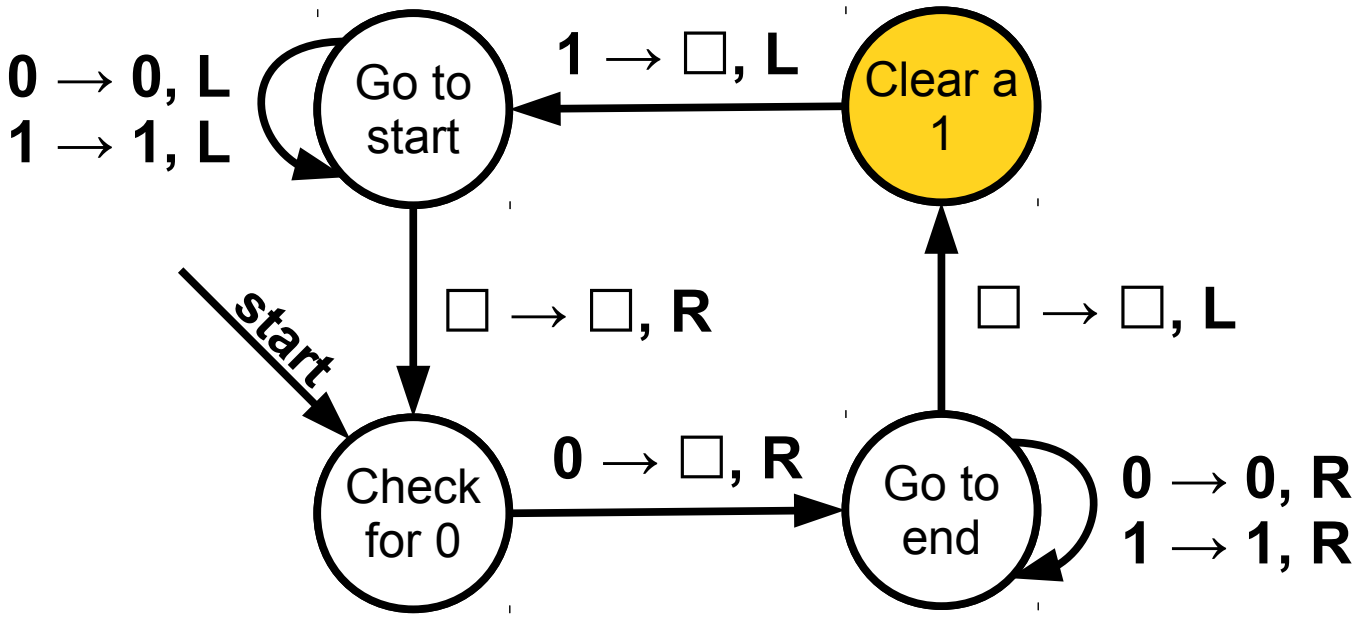

…
0
1
1
…

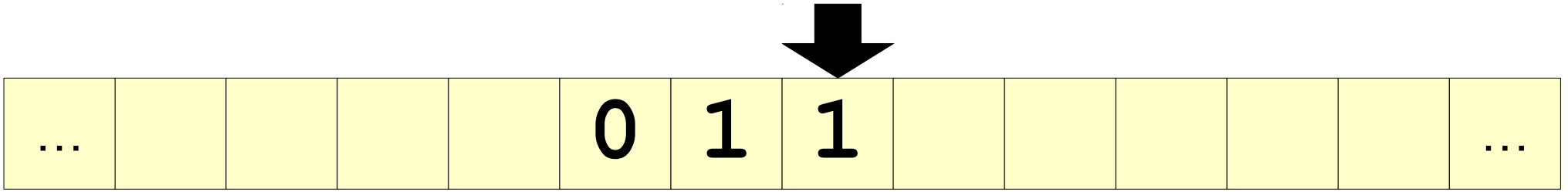

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

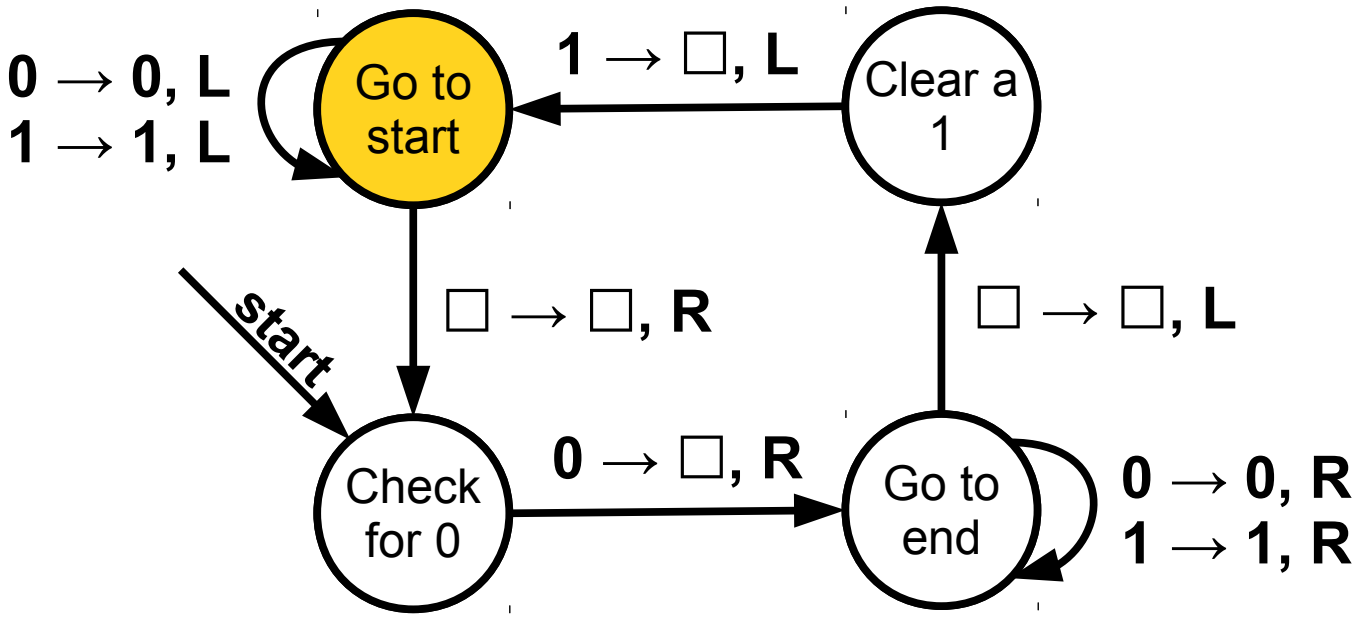

…
0
1
1
…

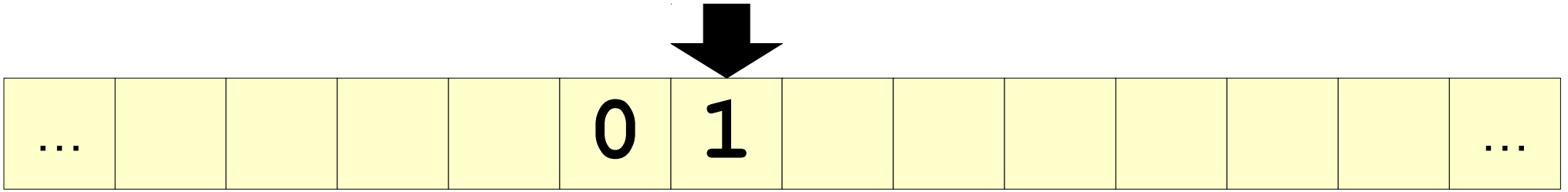

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

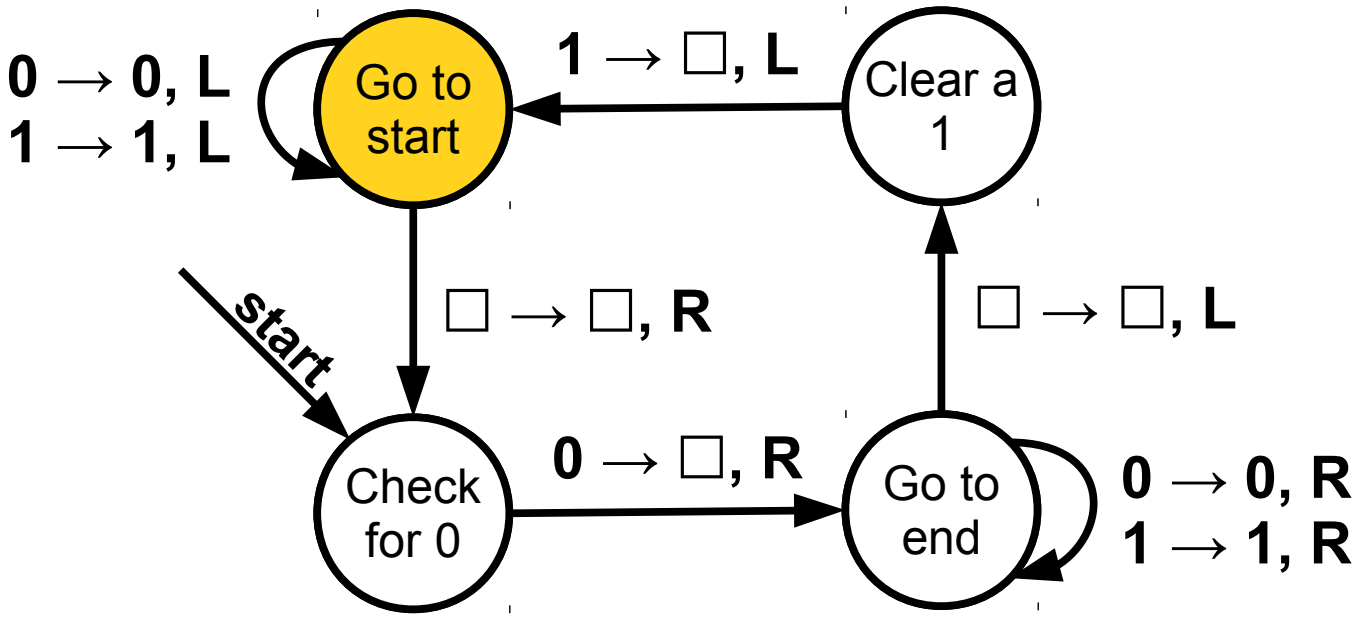

…
0
1
1
…

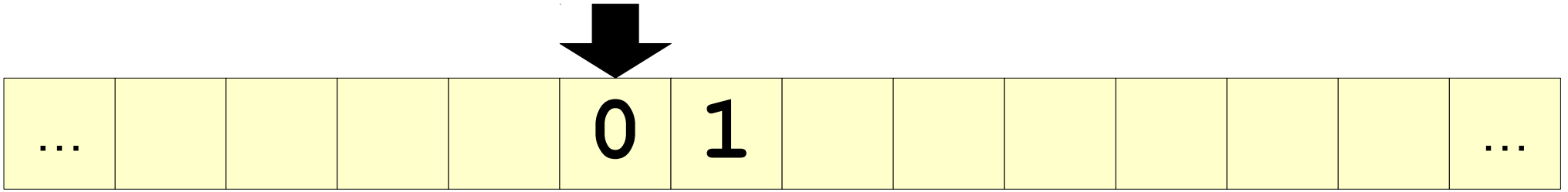

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

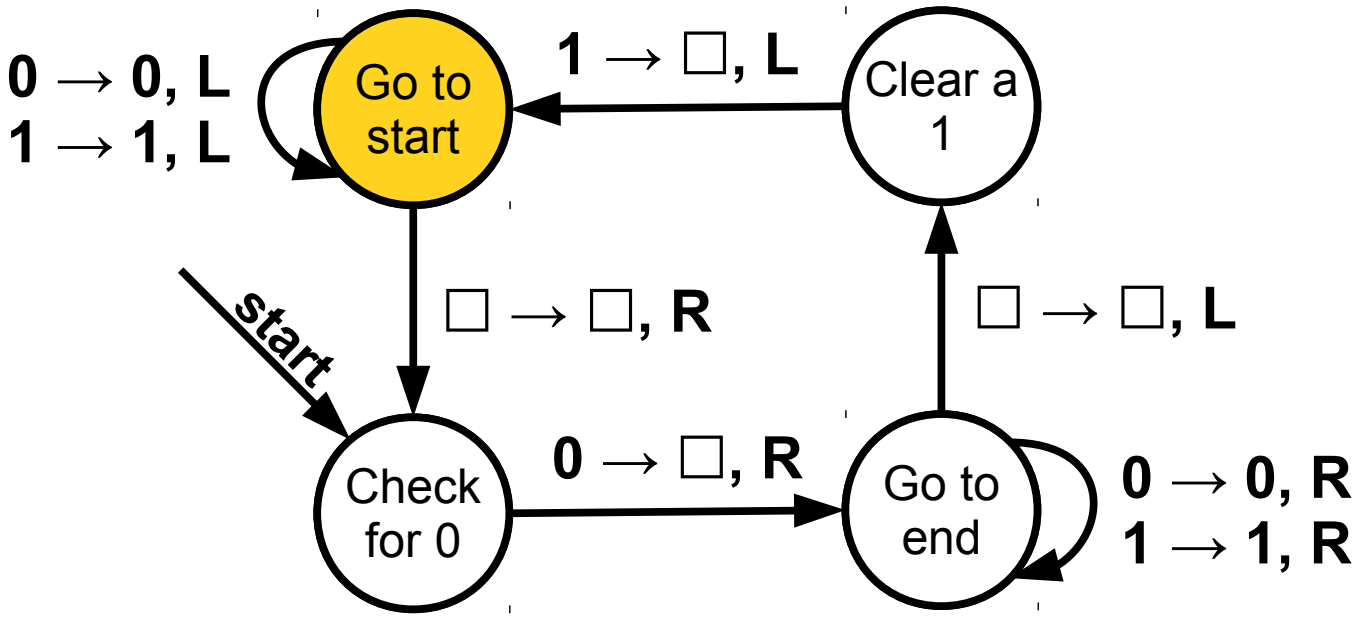

…
0
1
1
…

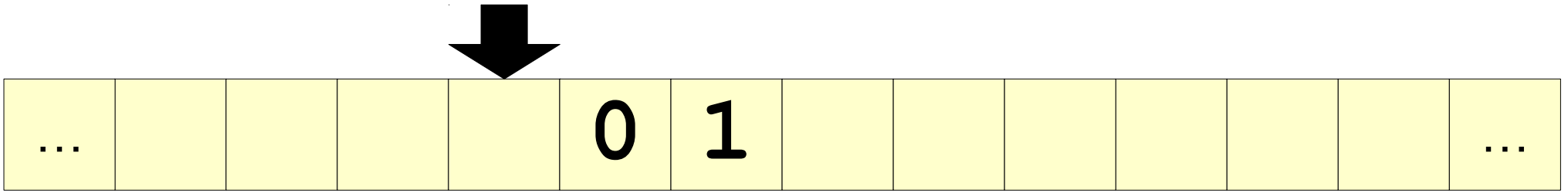

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

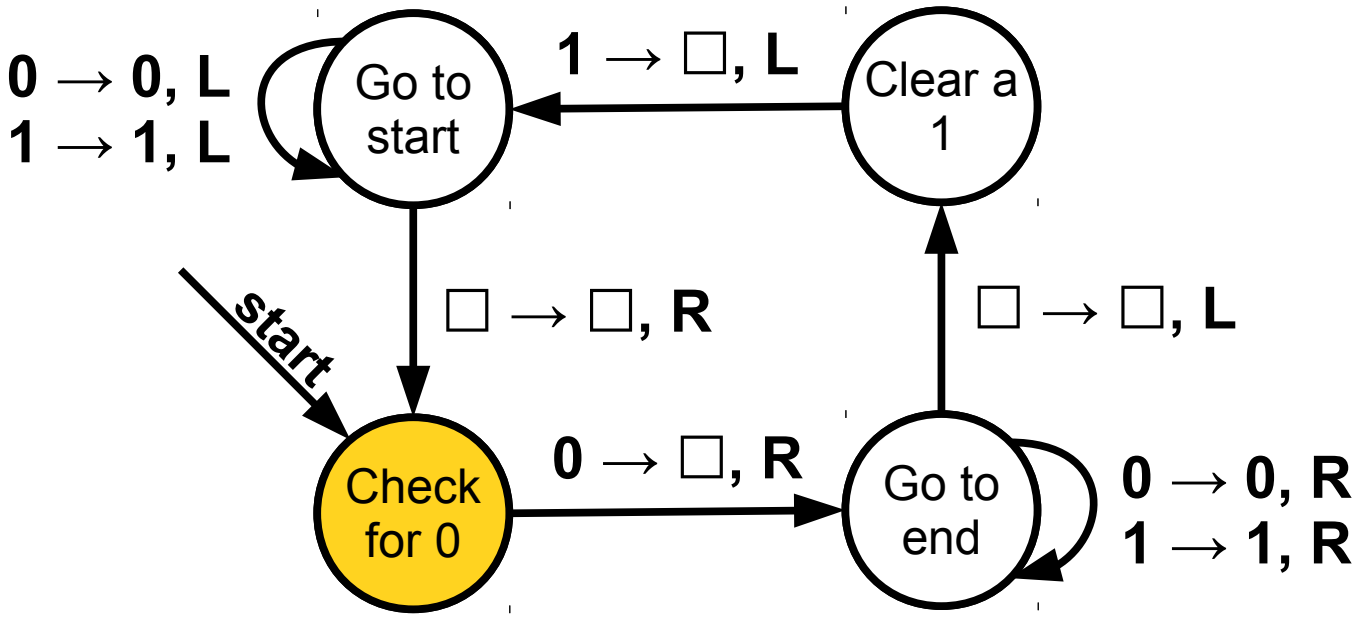

…
0
1
…

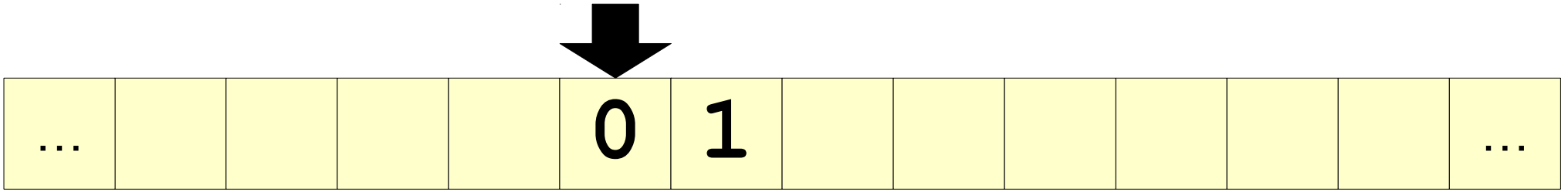

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

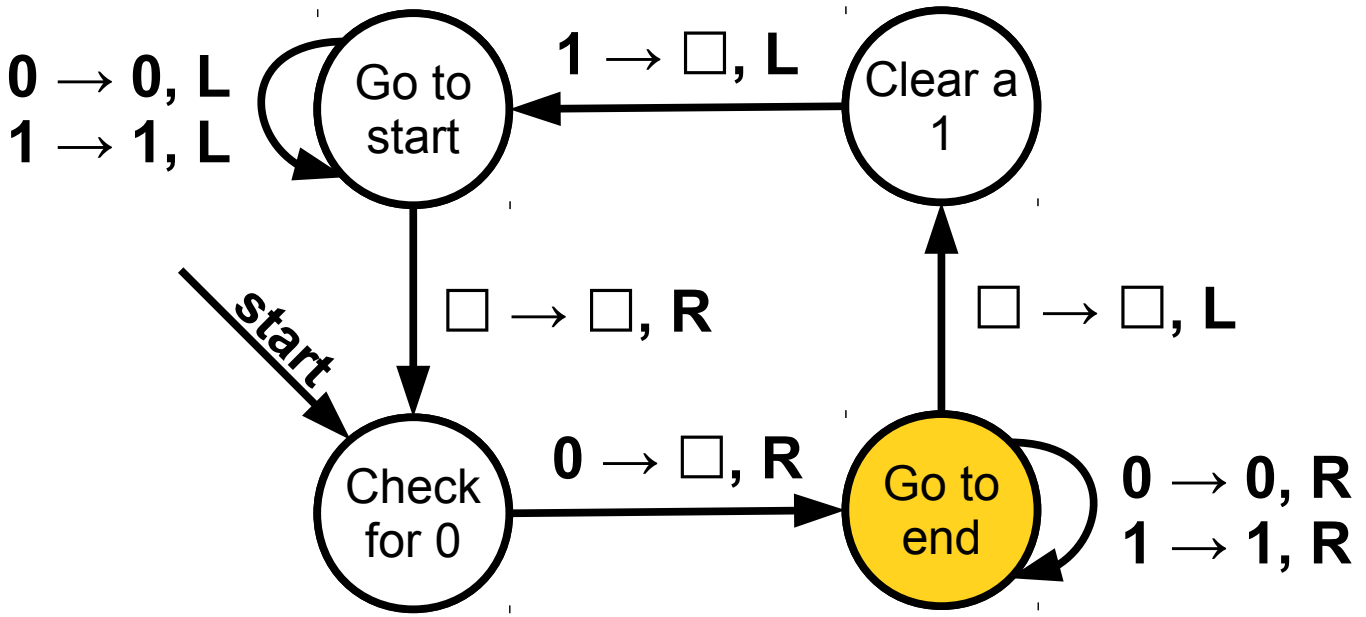

…
0
1
…

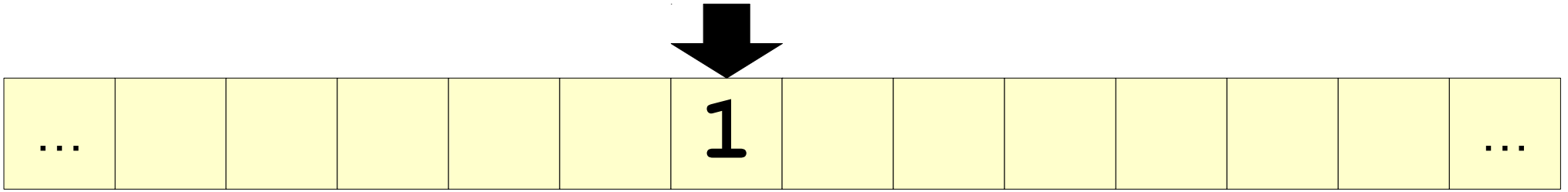

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

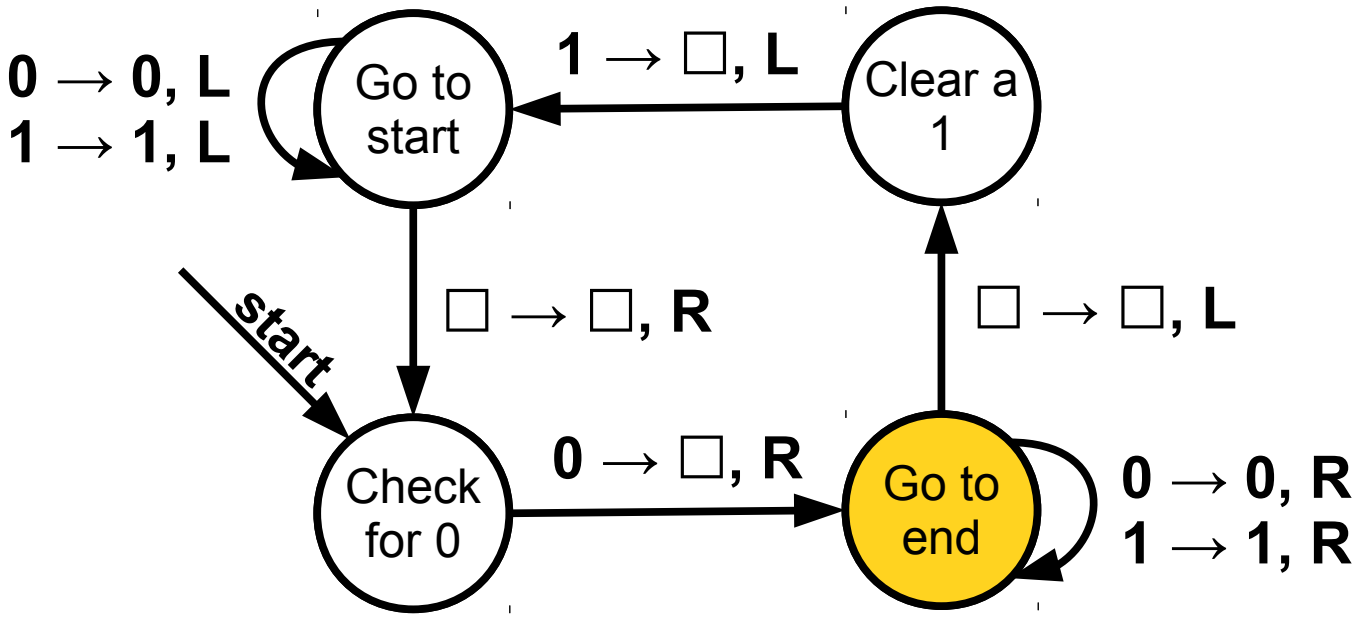

…
0
1
…

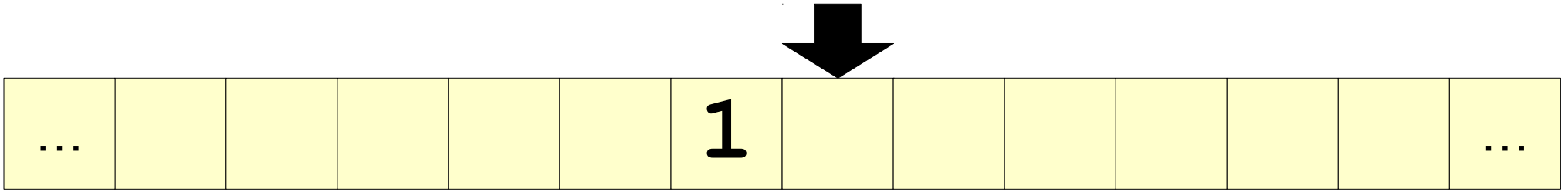

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

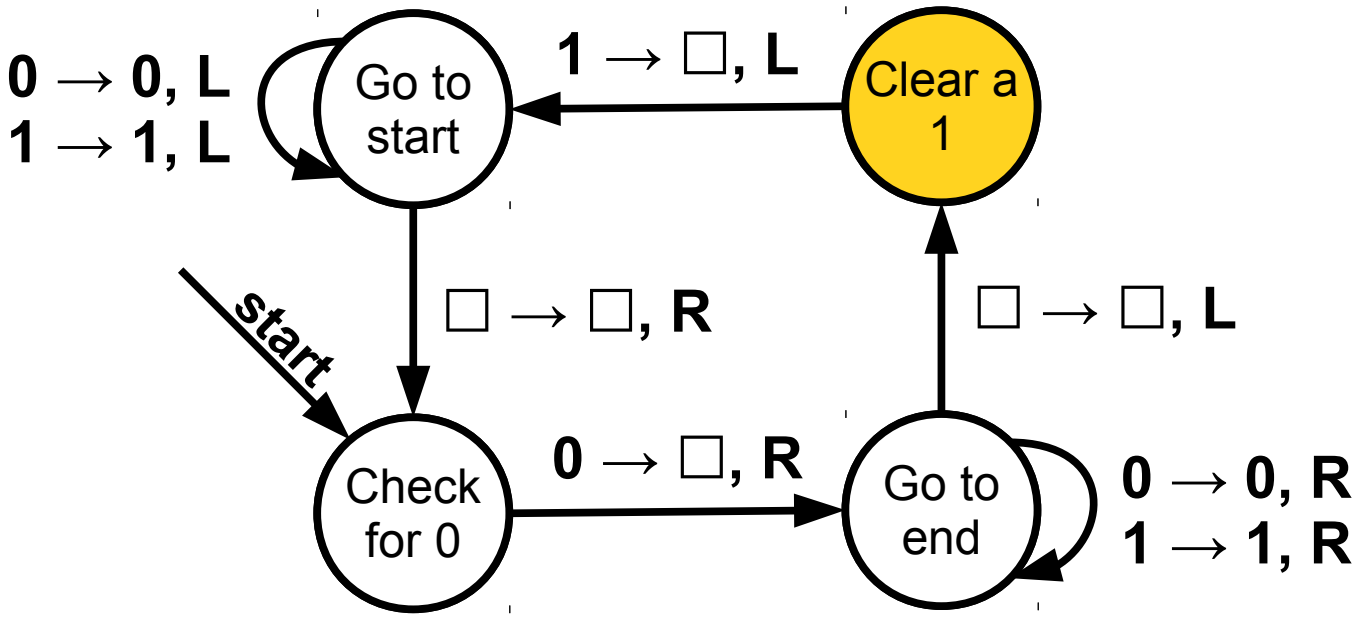

…
0
1
…

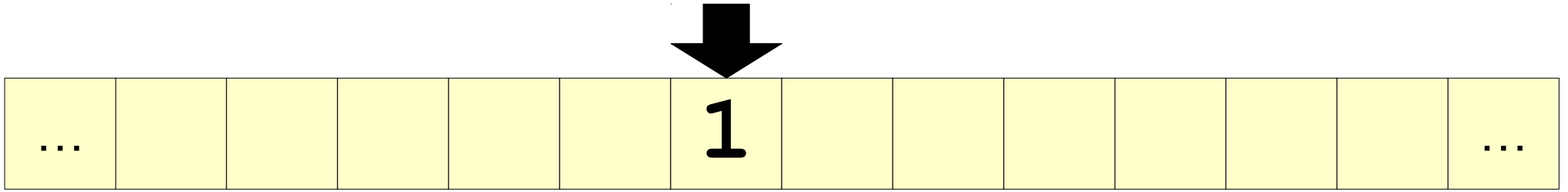

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

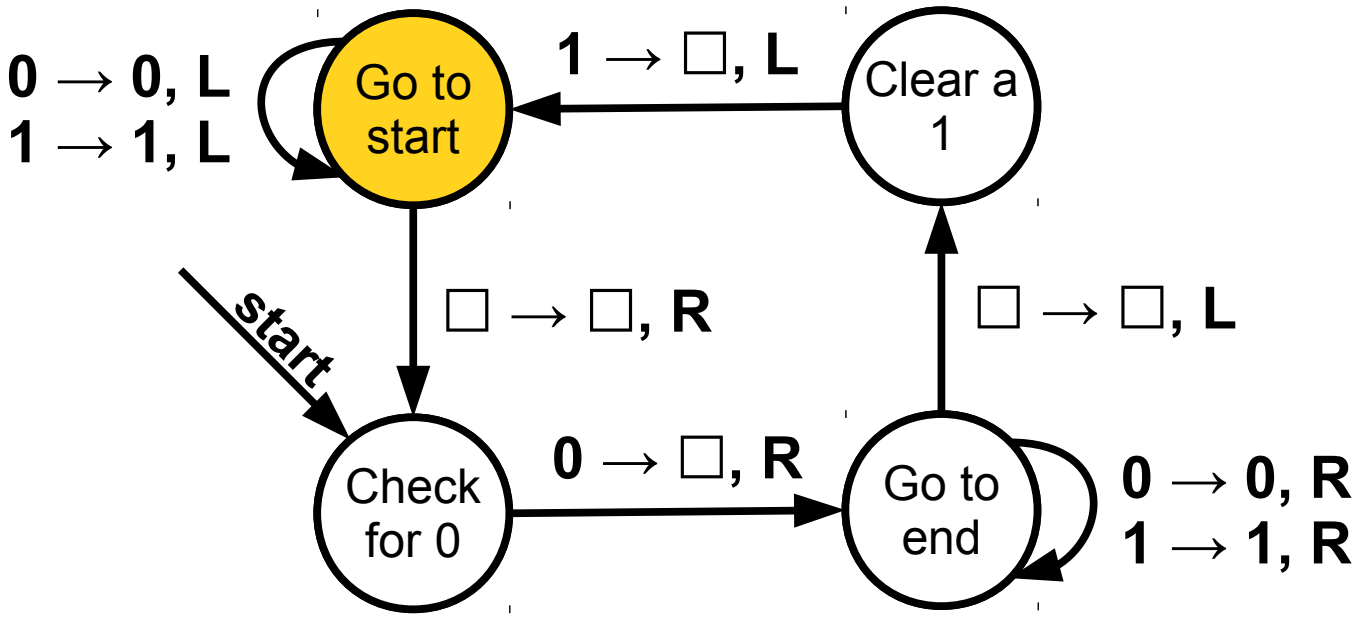

…
1
…

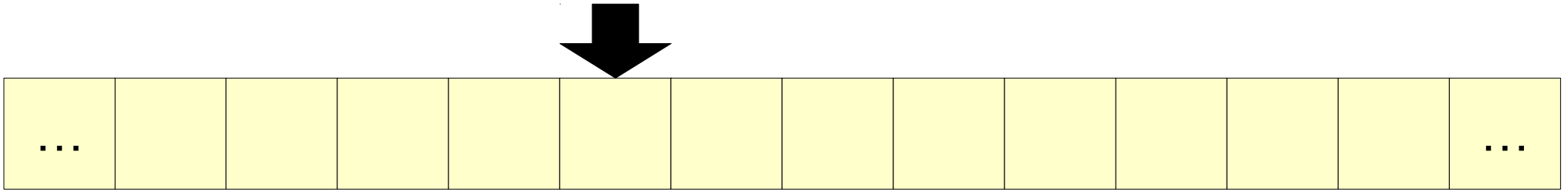

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

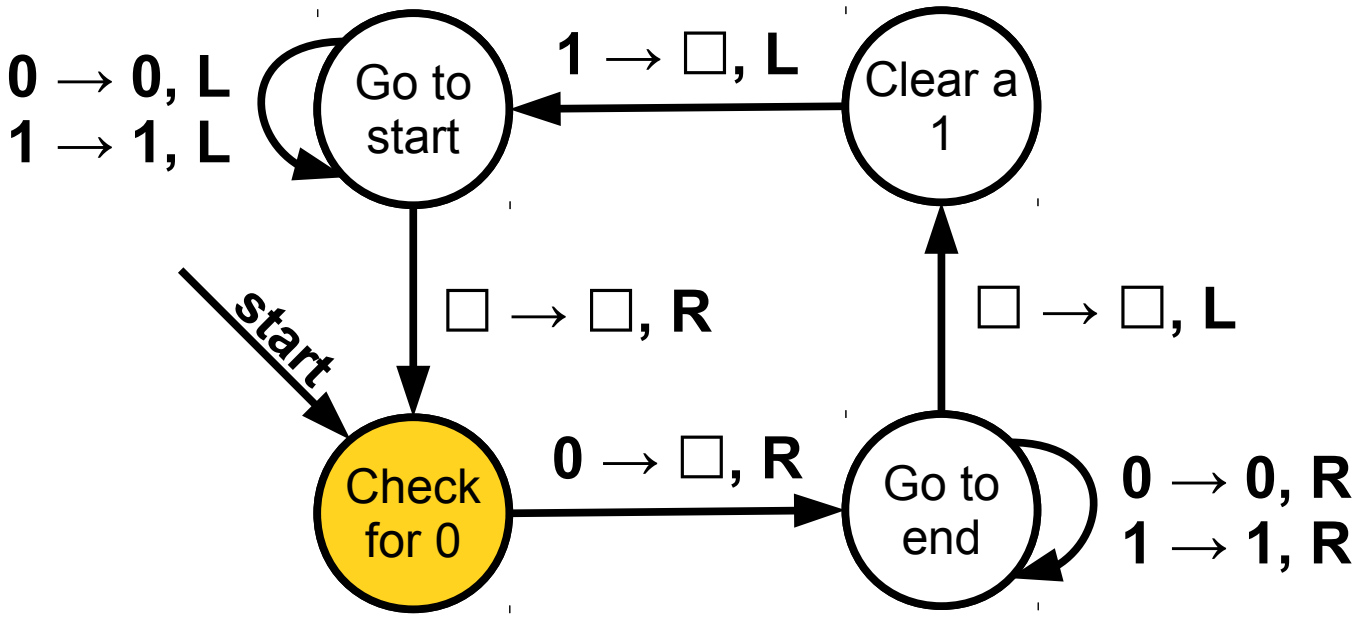

…
1
…

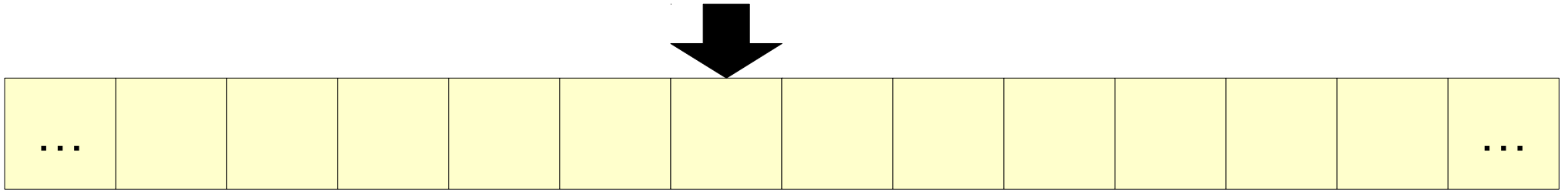

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

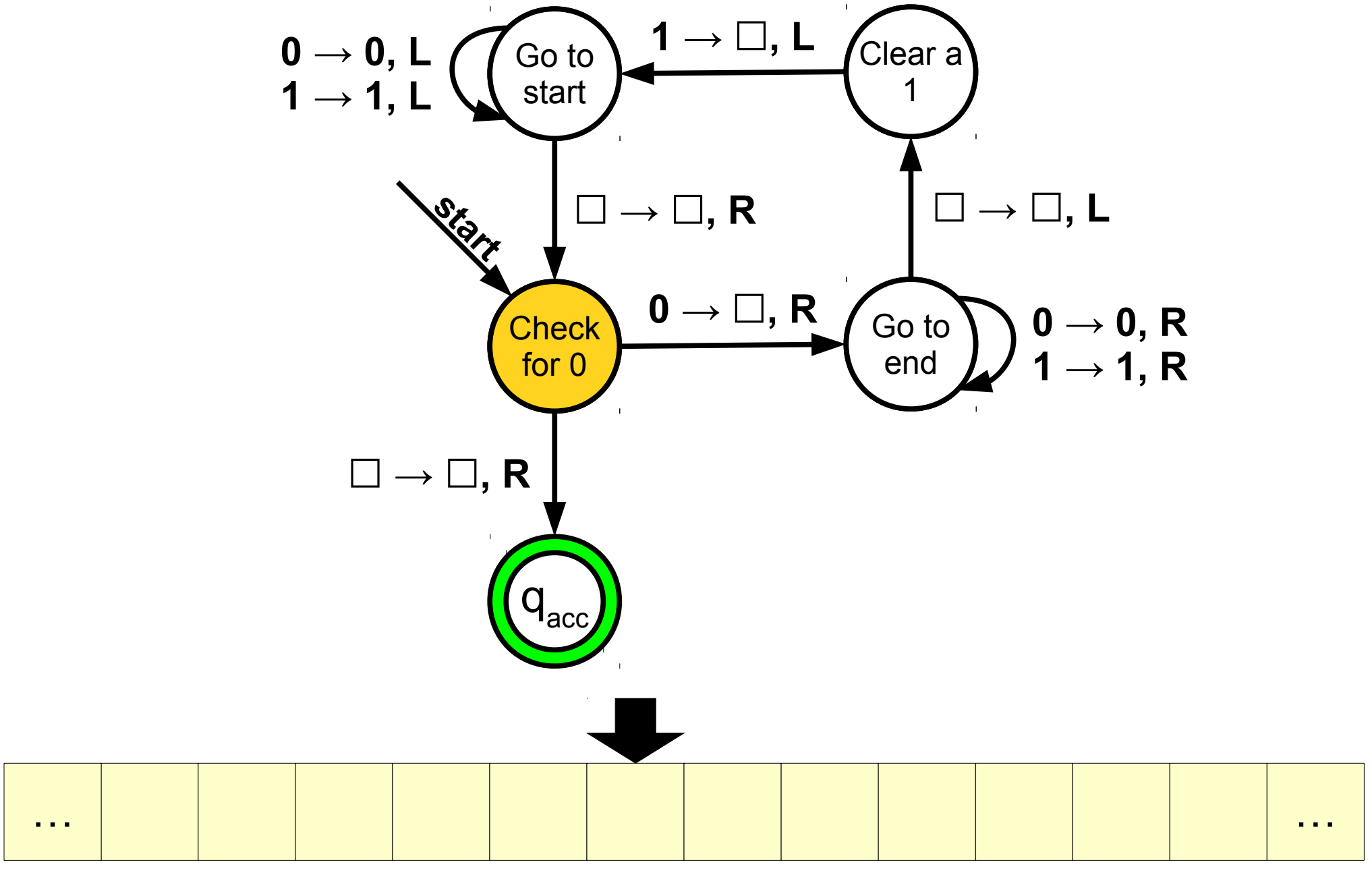

…
1
…

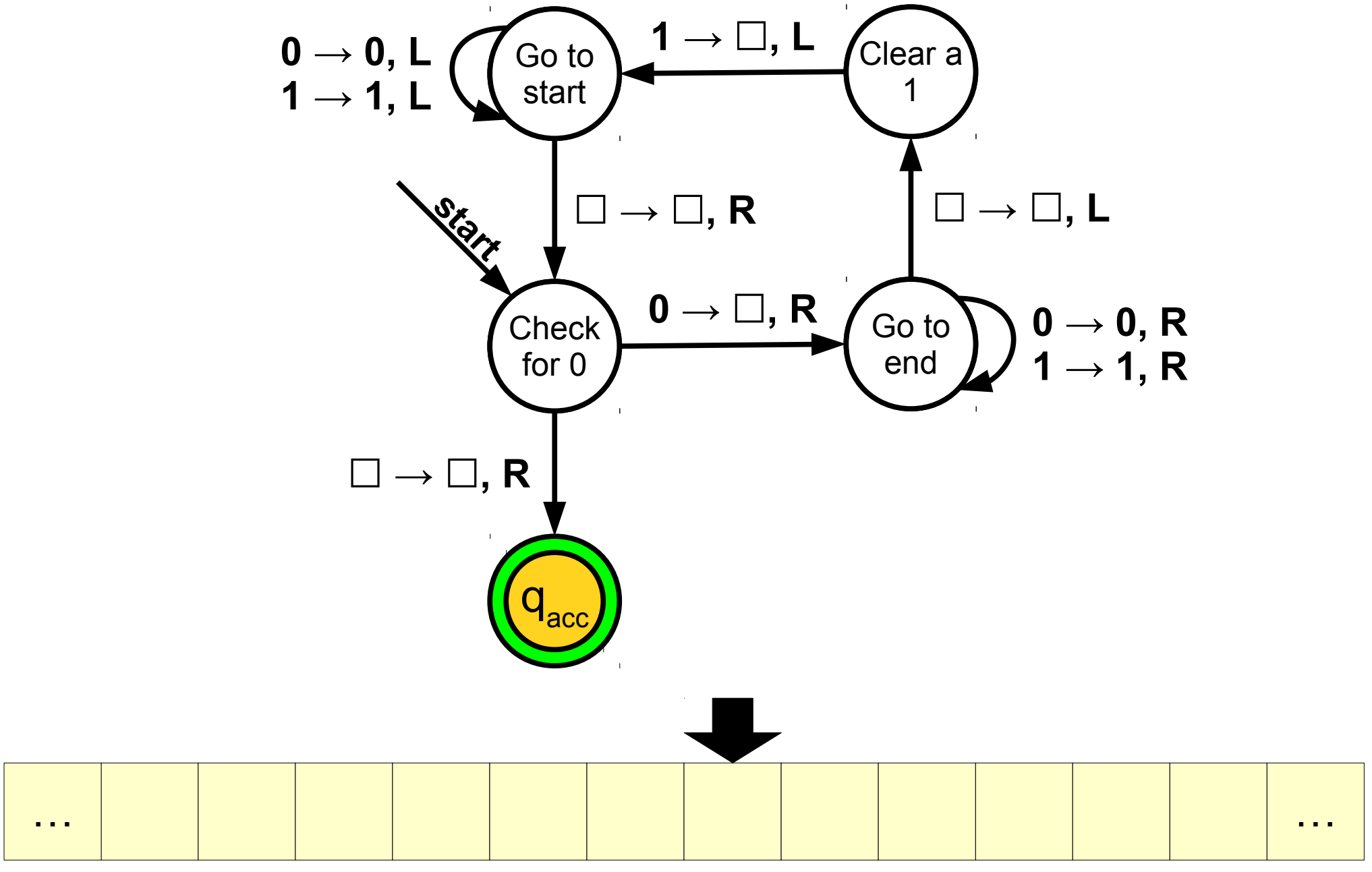

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

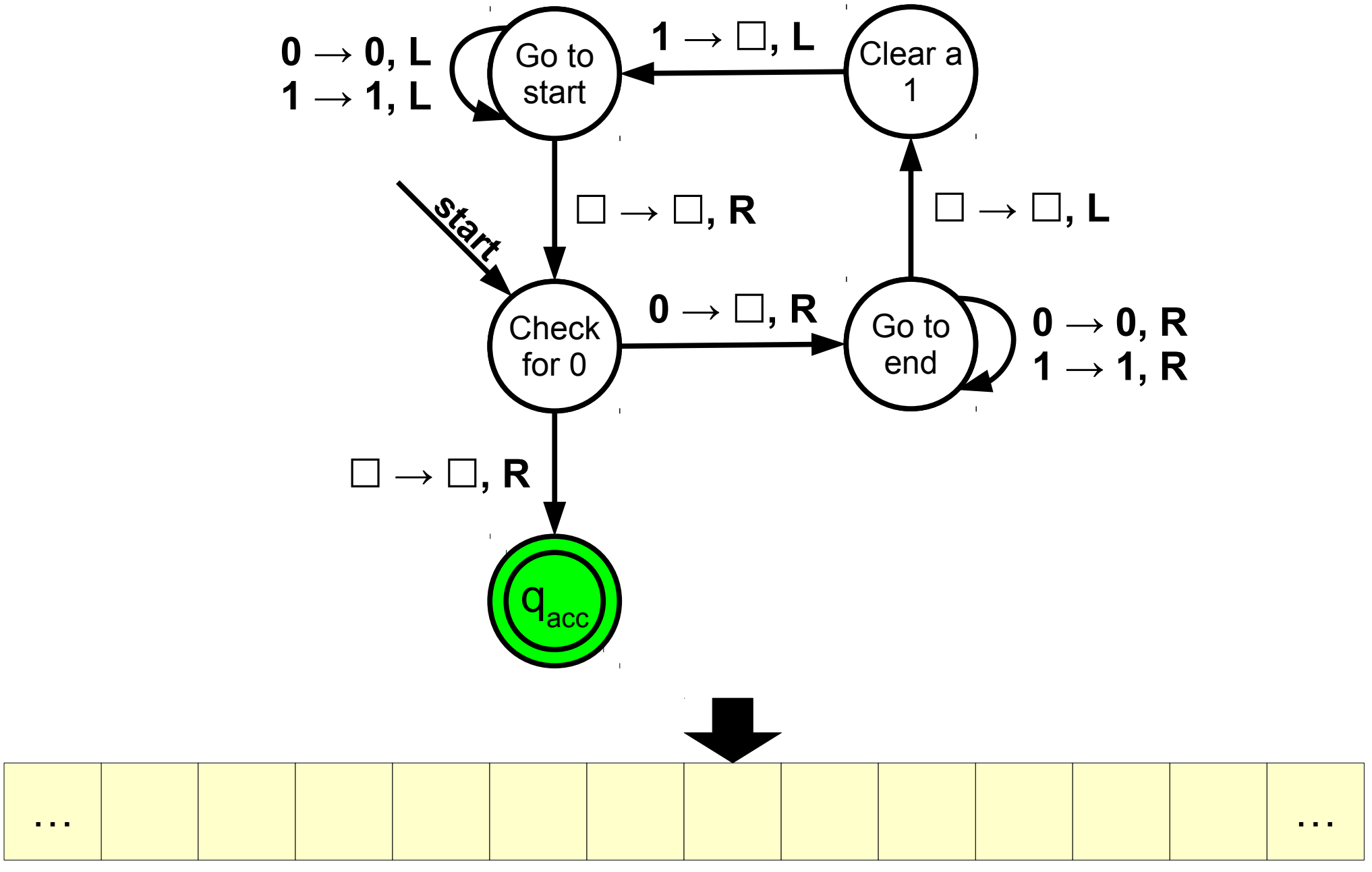

...
...

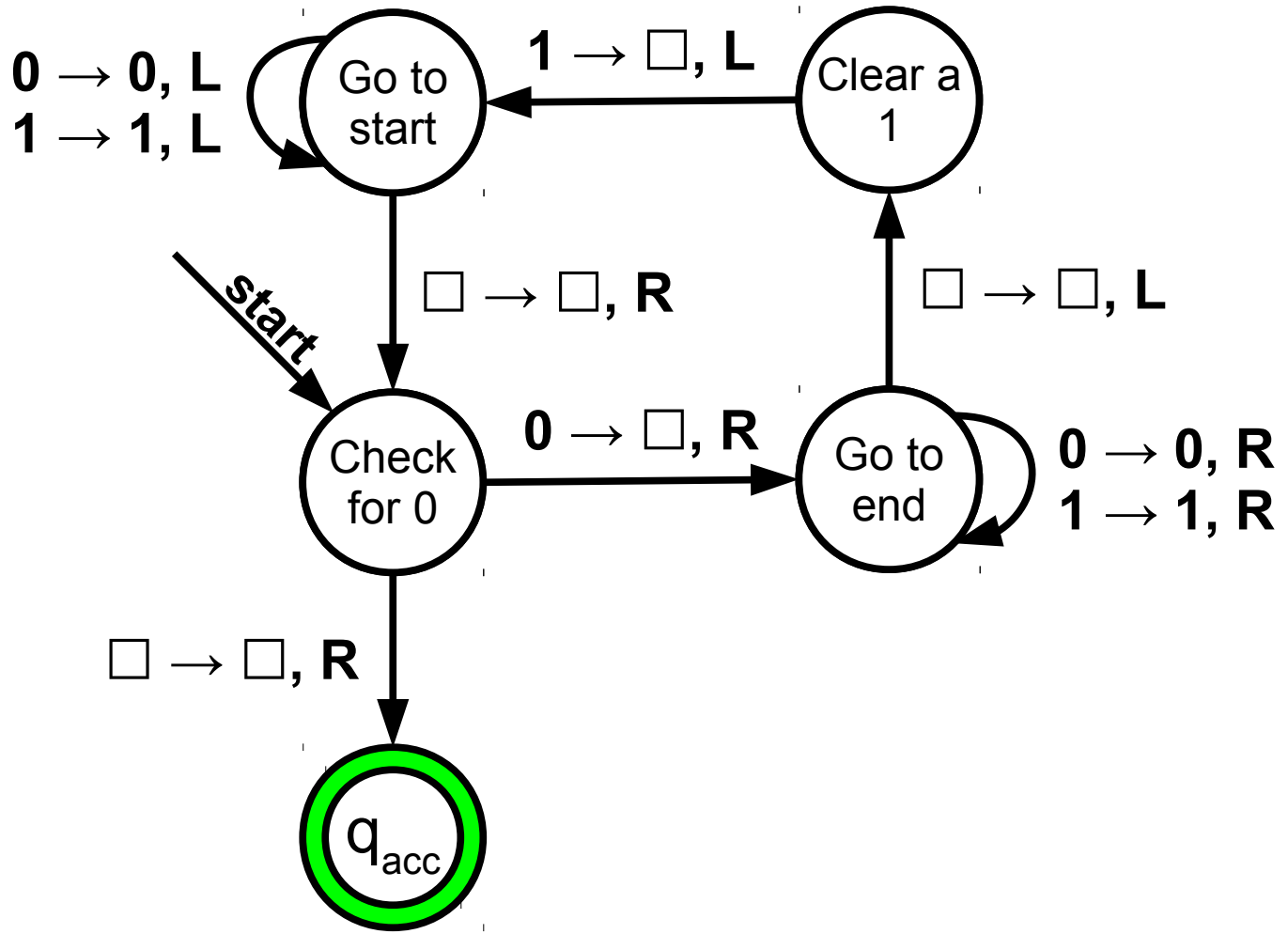

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R

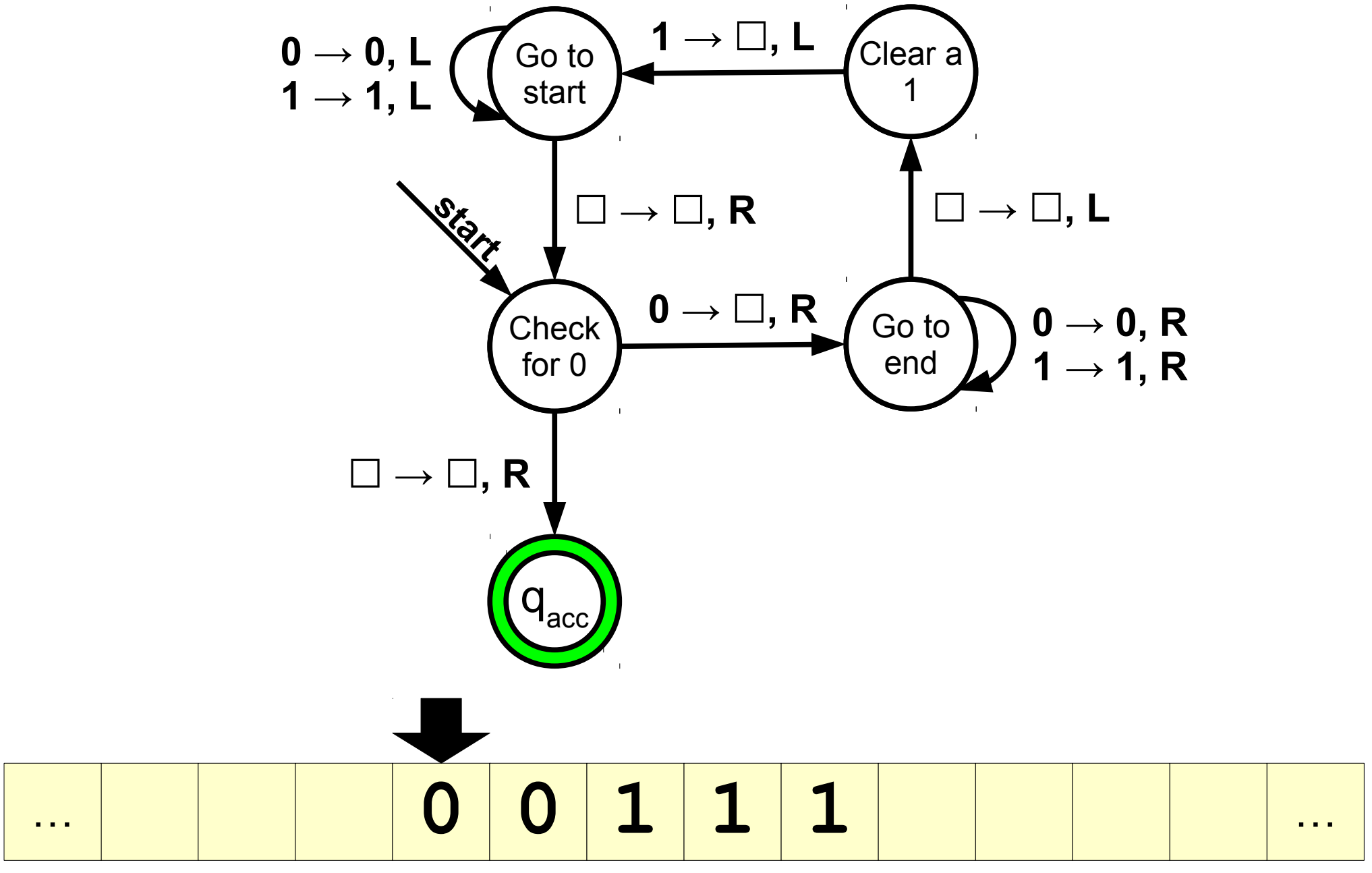

...
...

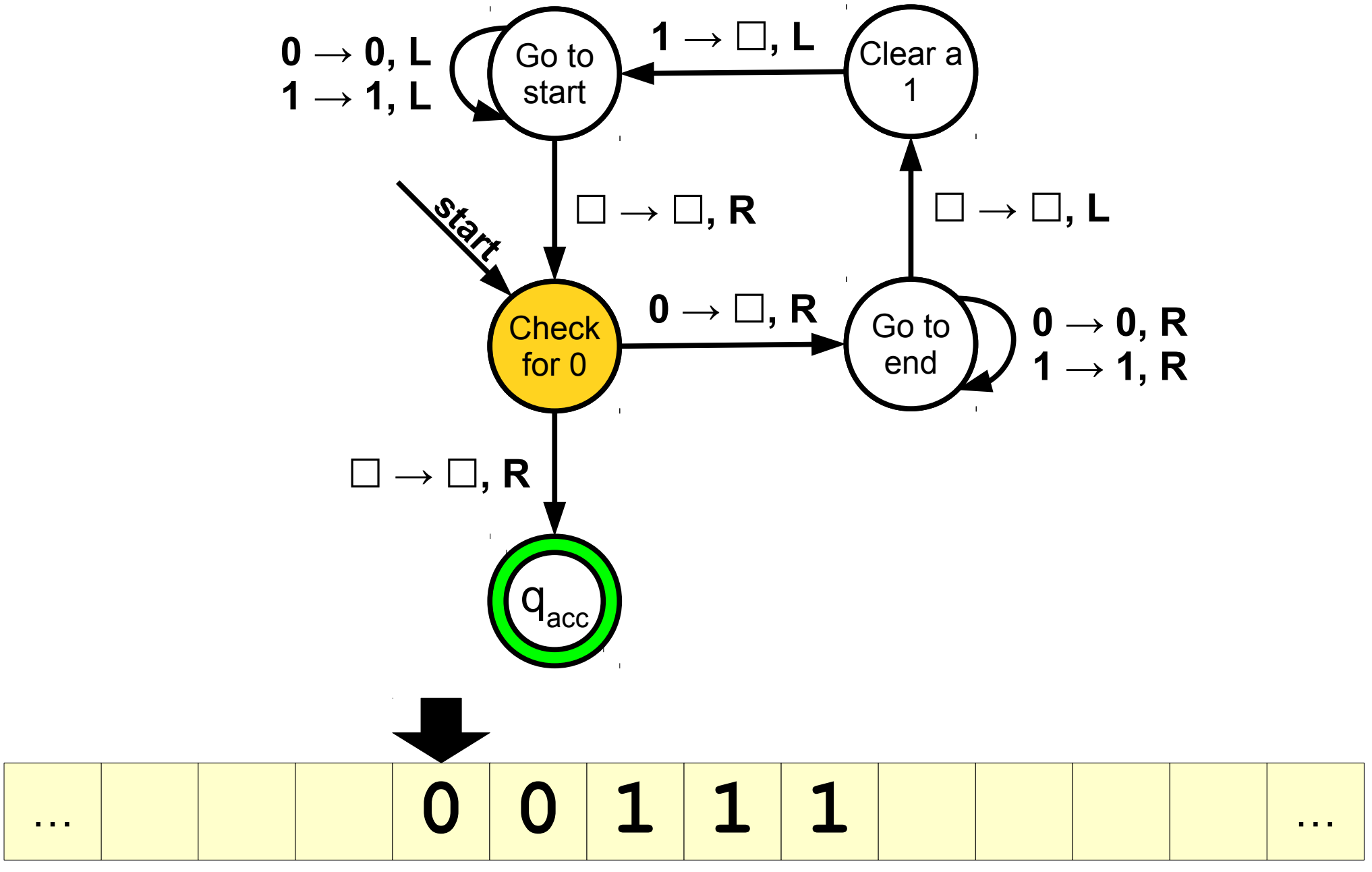

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
…

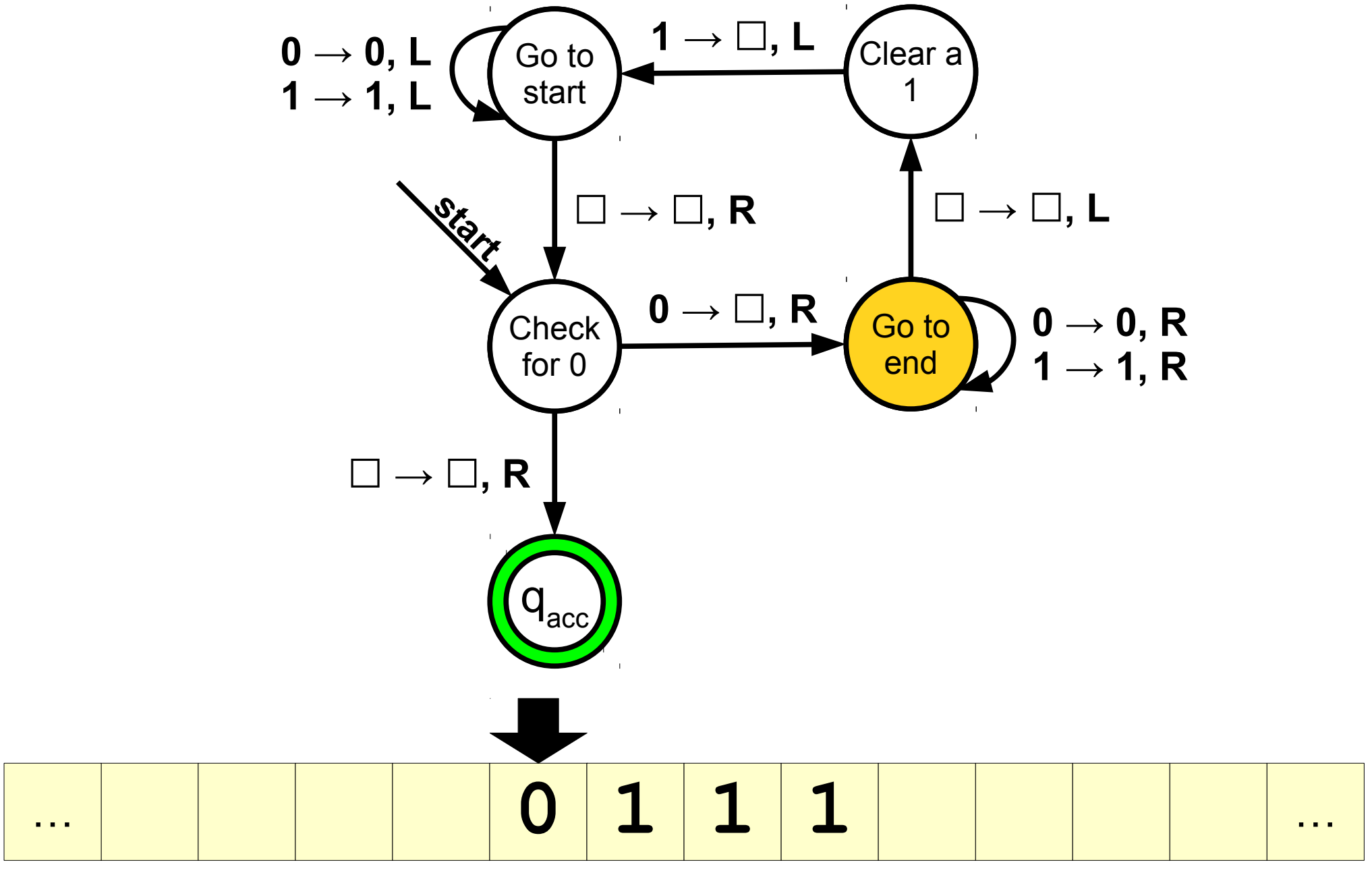

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
…

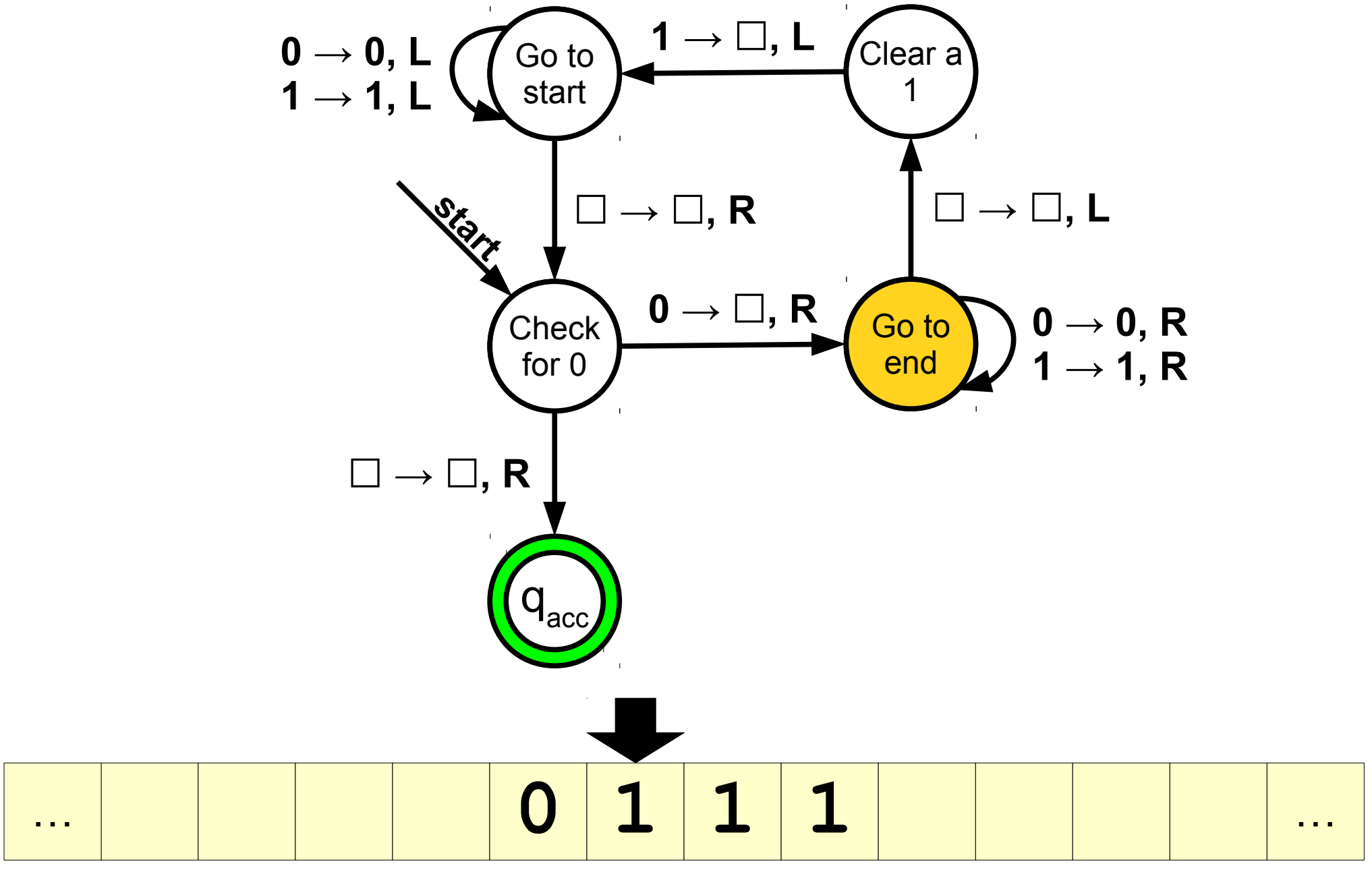

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
…

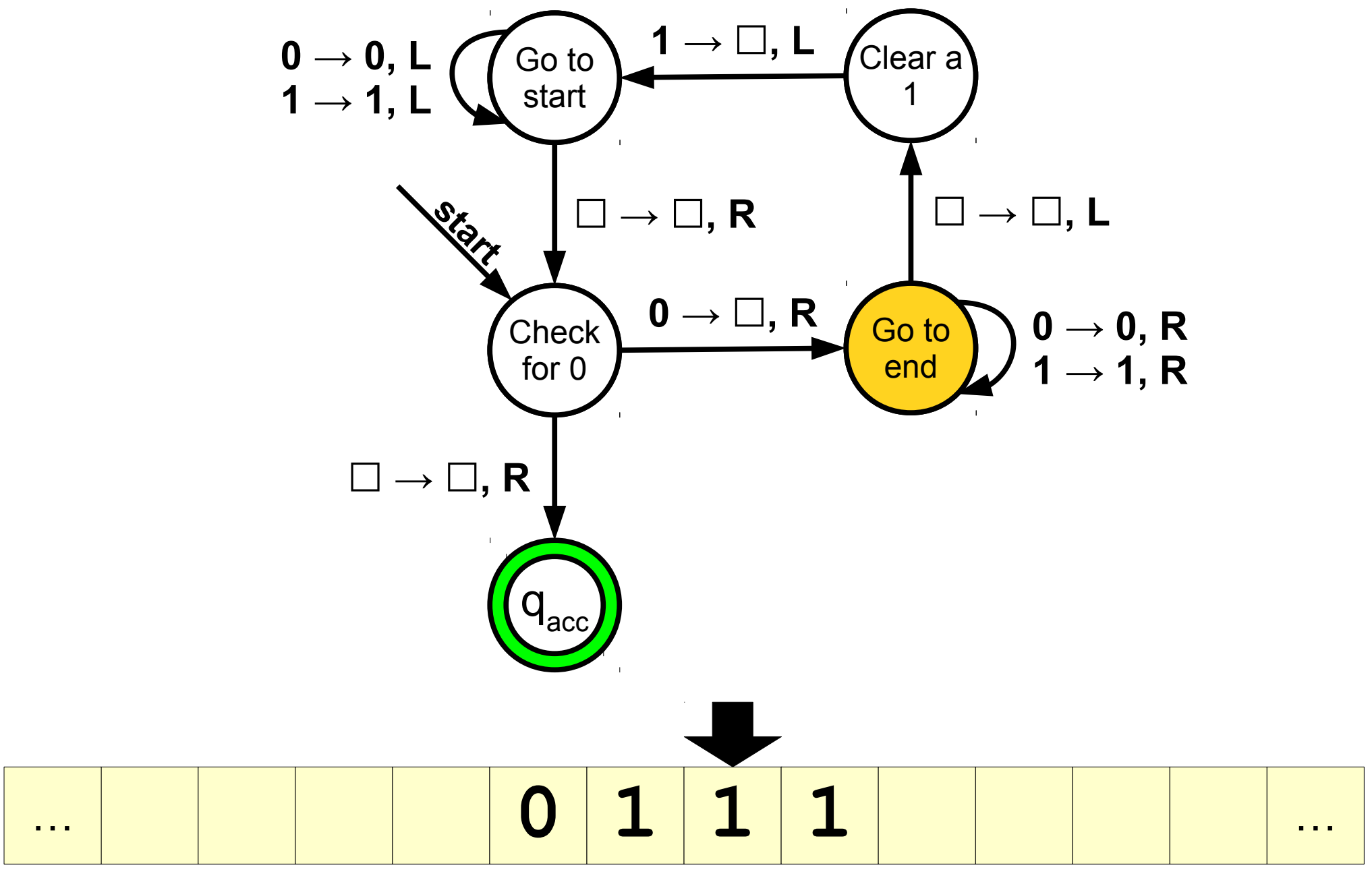

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}

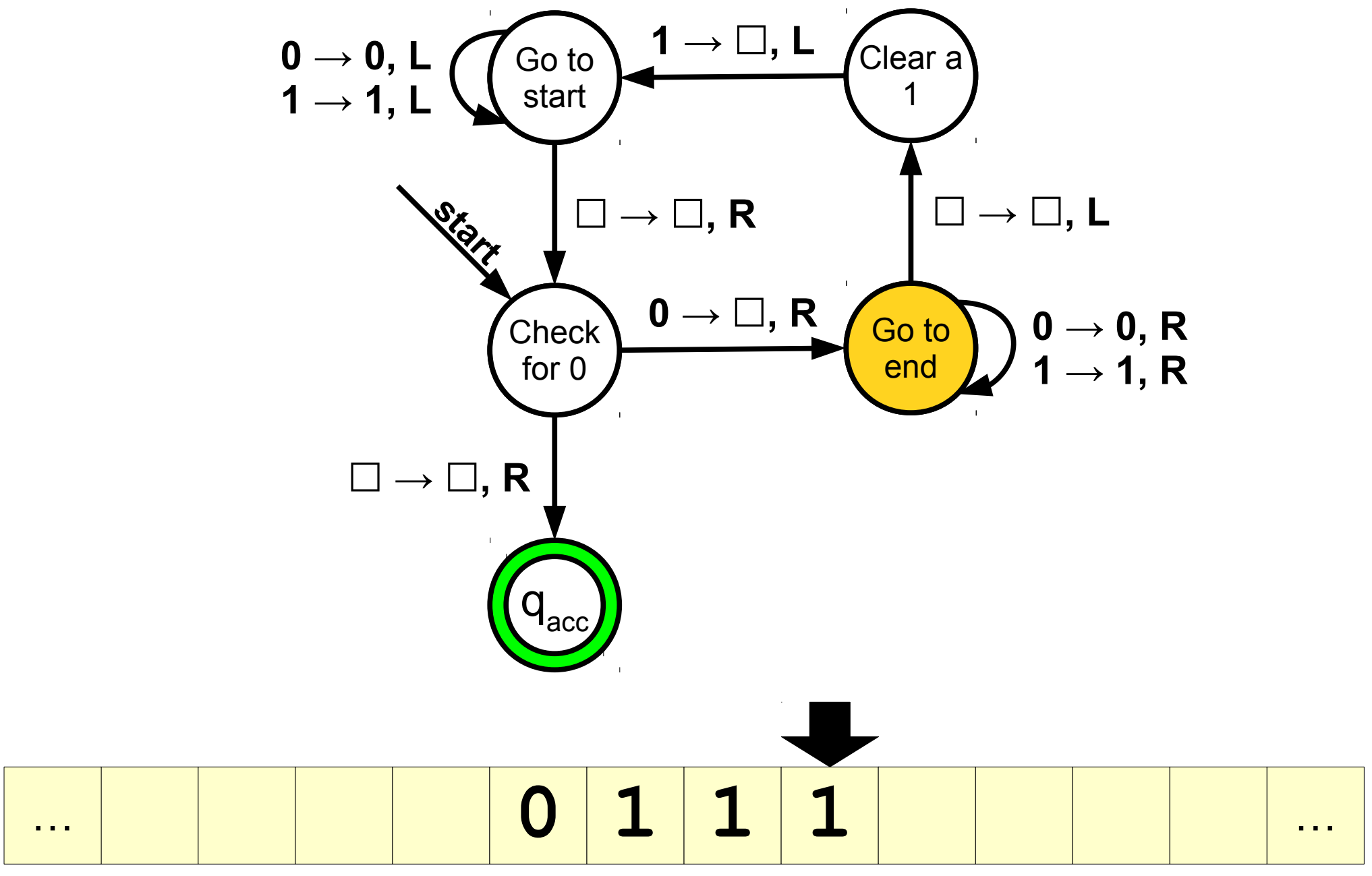

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
0
1
1
1
…

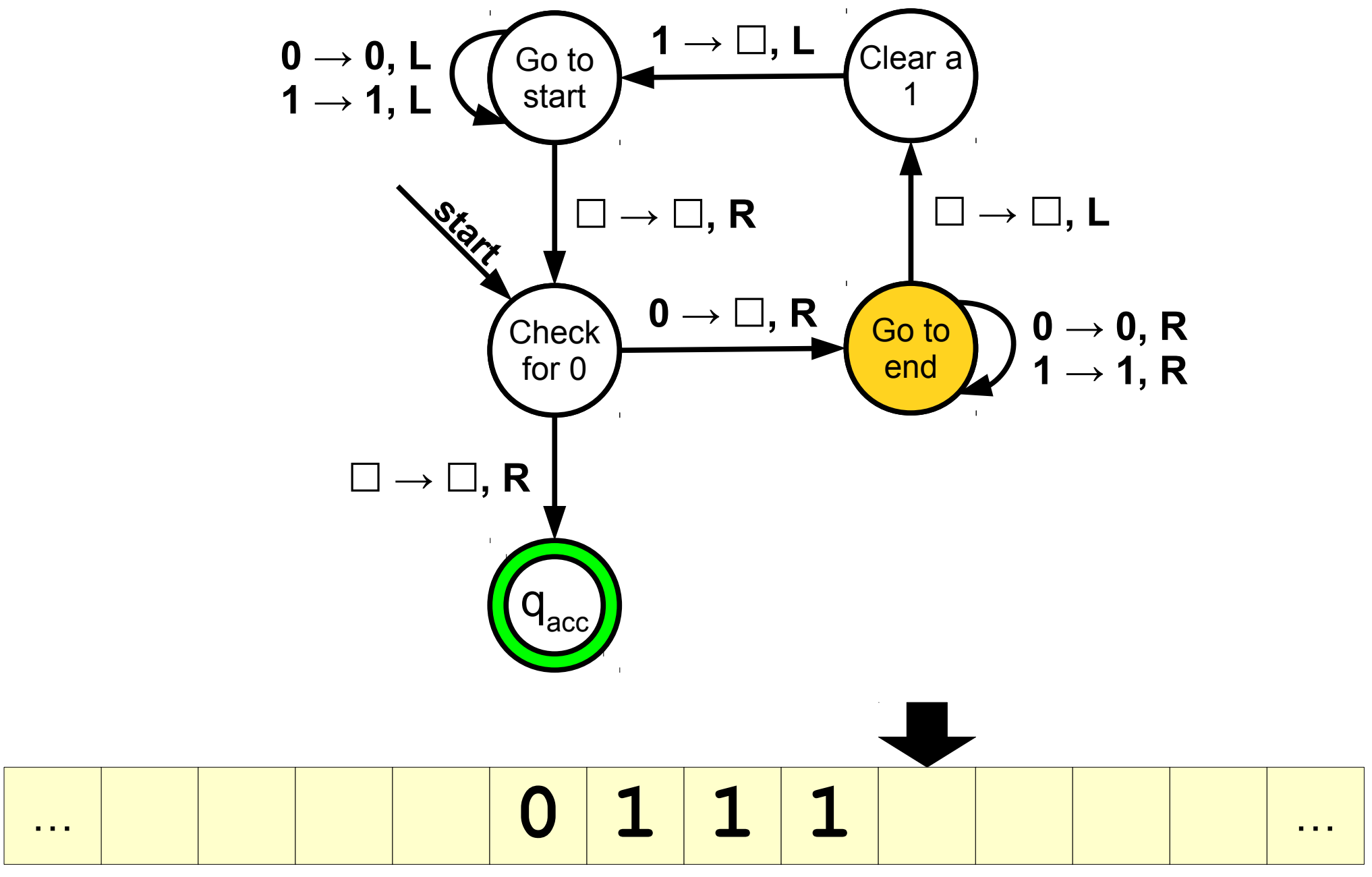

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
...
0
0
1
1
1
...

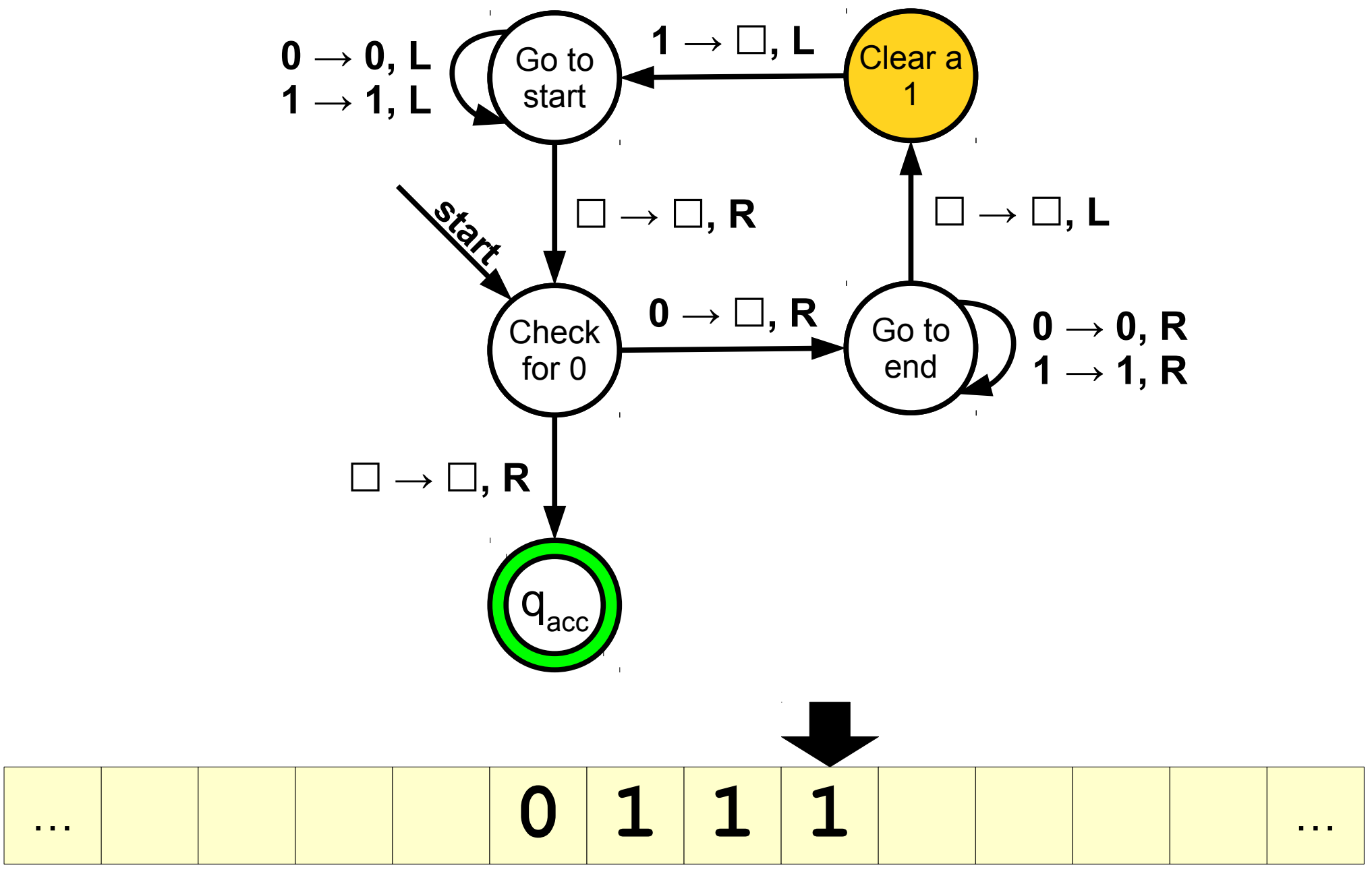

Go to start
Clear a 1
Check for 0
Go to end
q_{acc}
start
0 → 0, L
1 → 1, L
1 → □, L
□ → □, R
□ → □, L
0 → □, R
0 → 0, R
1 → 1, R
□ → □, R
…
0
1
1
1
…

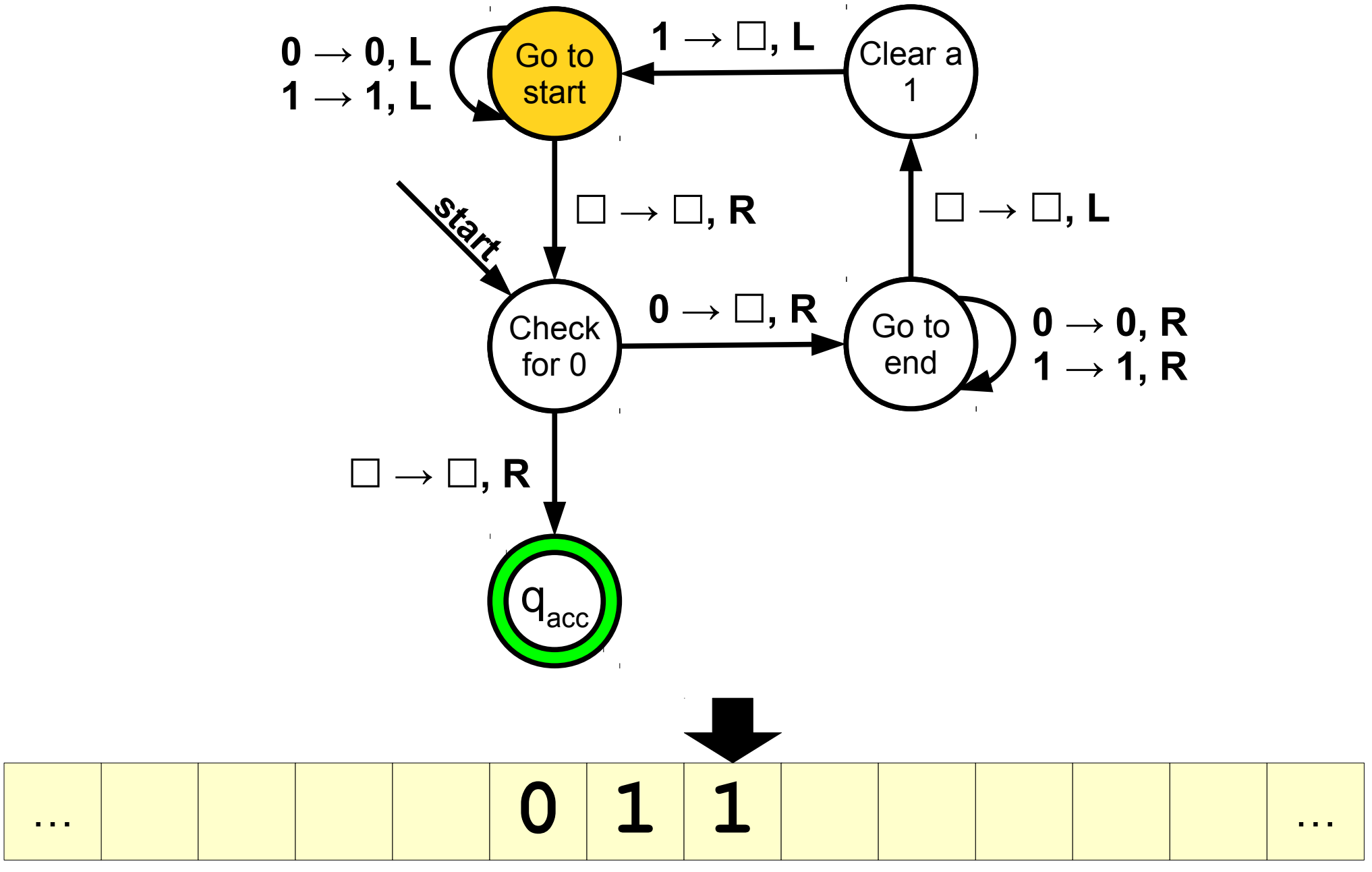

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
1
…

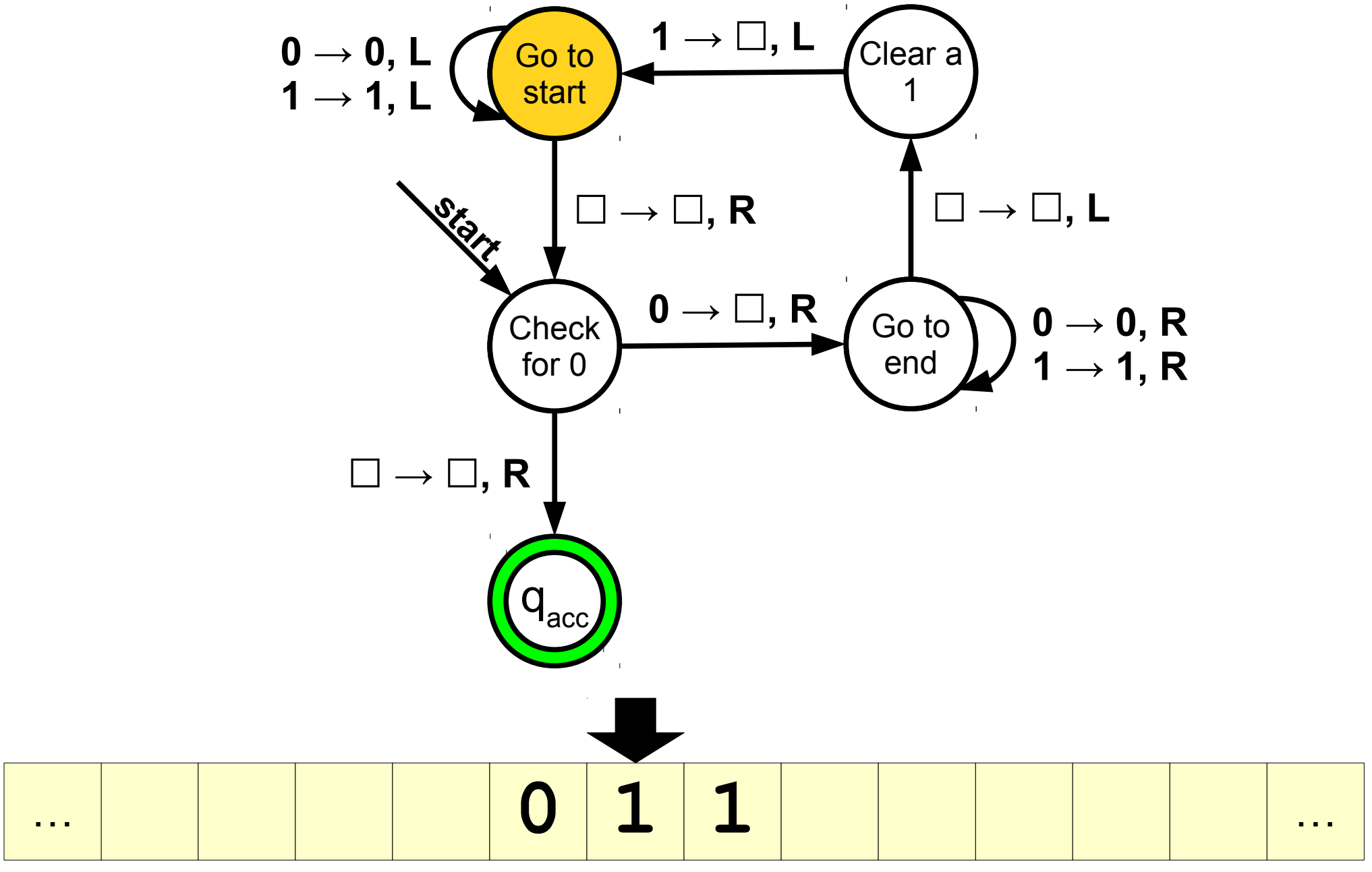

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
1
…

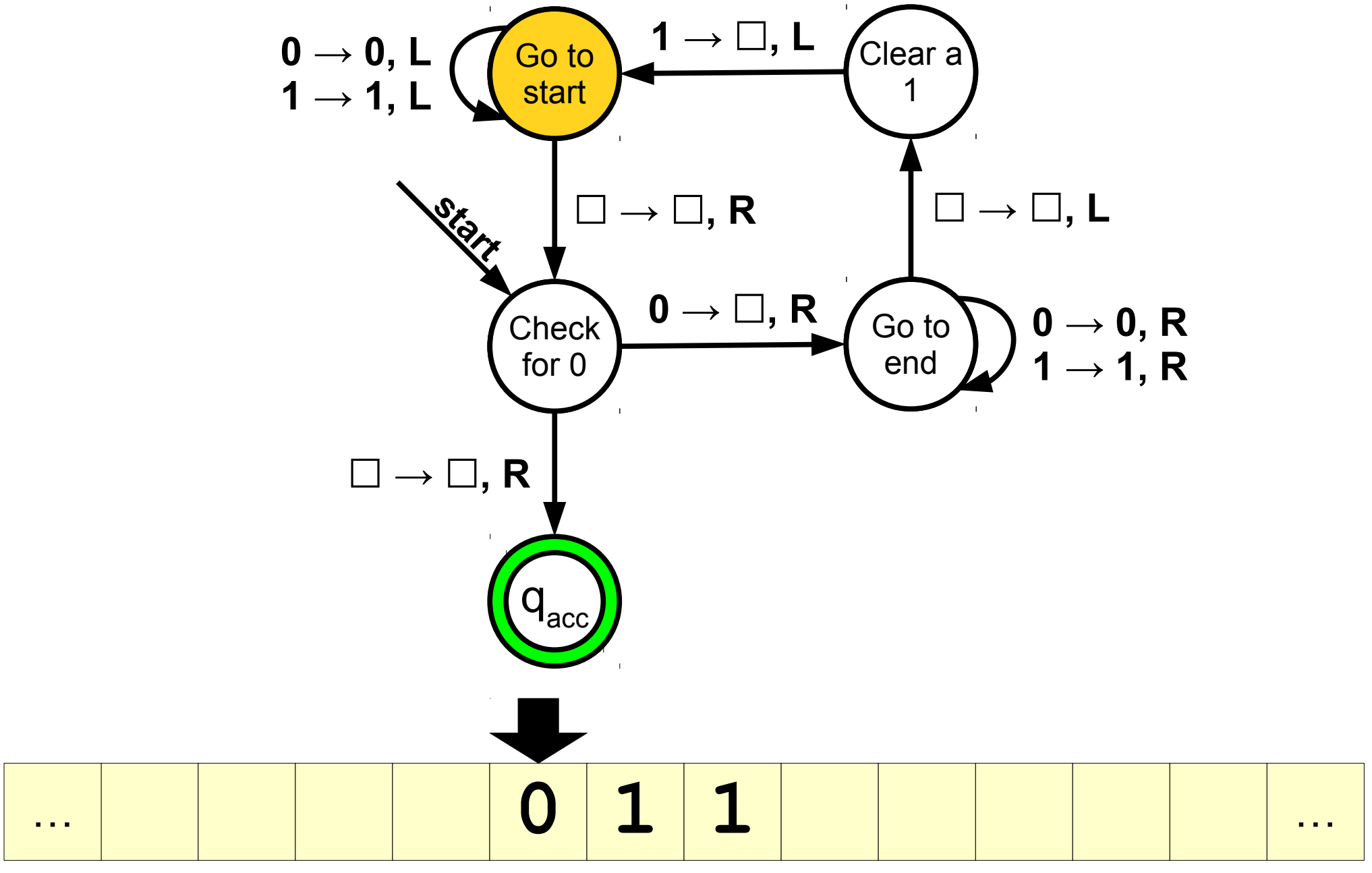

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
1
…

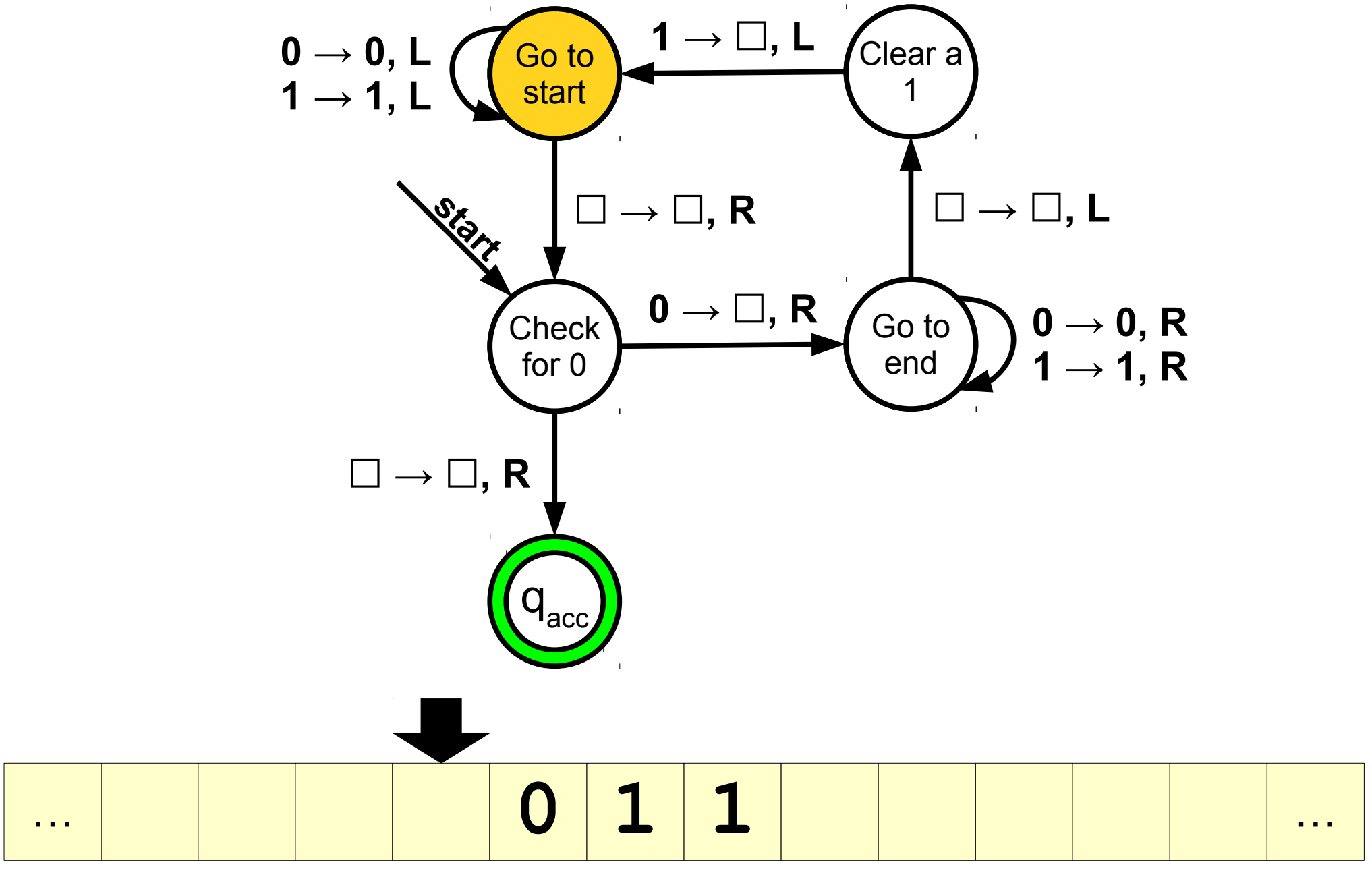

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0 1 1 1
…

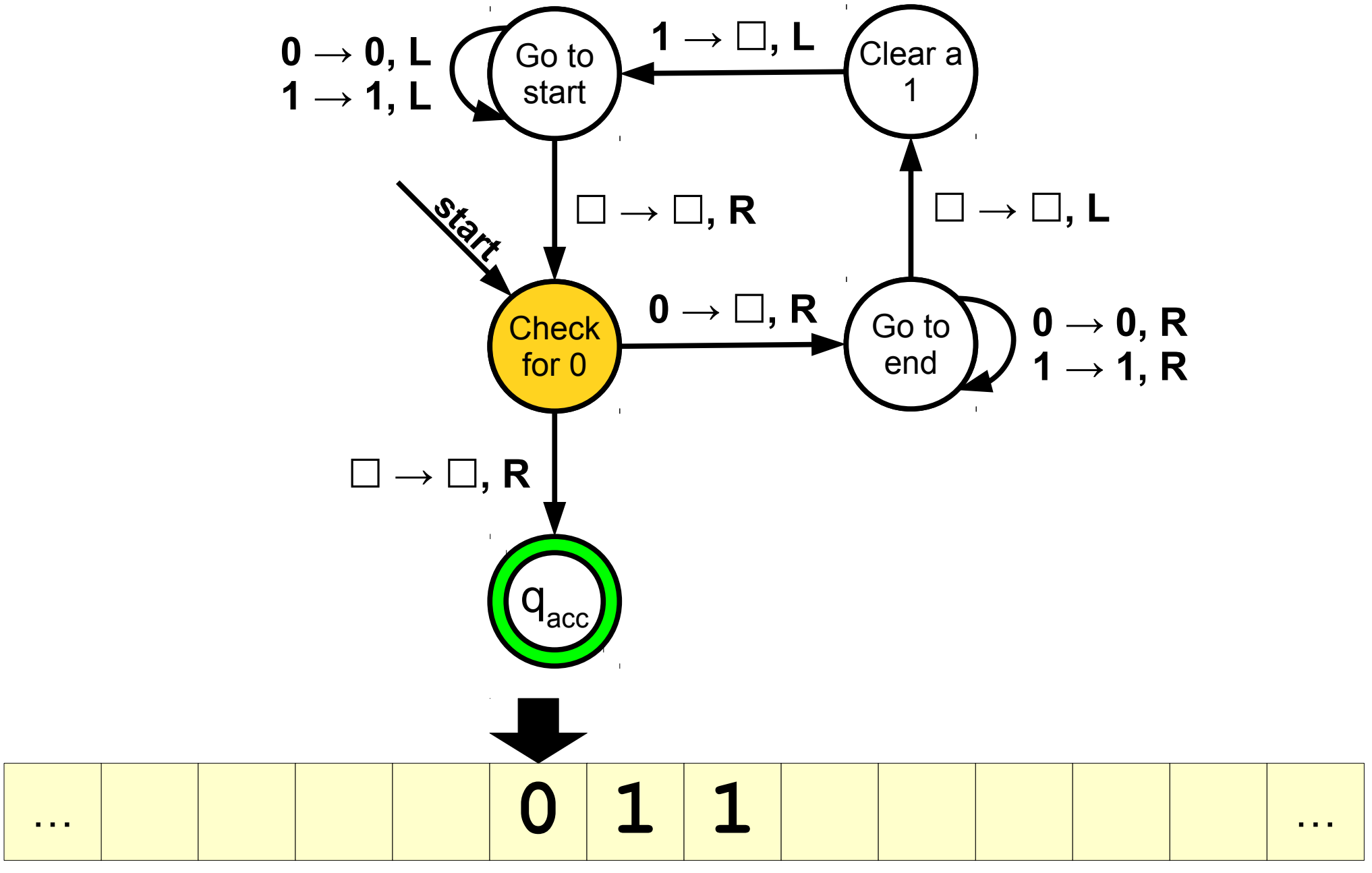

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
1
…

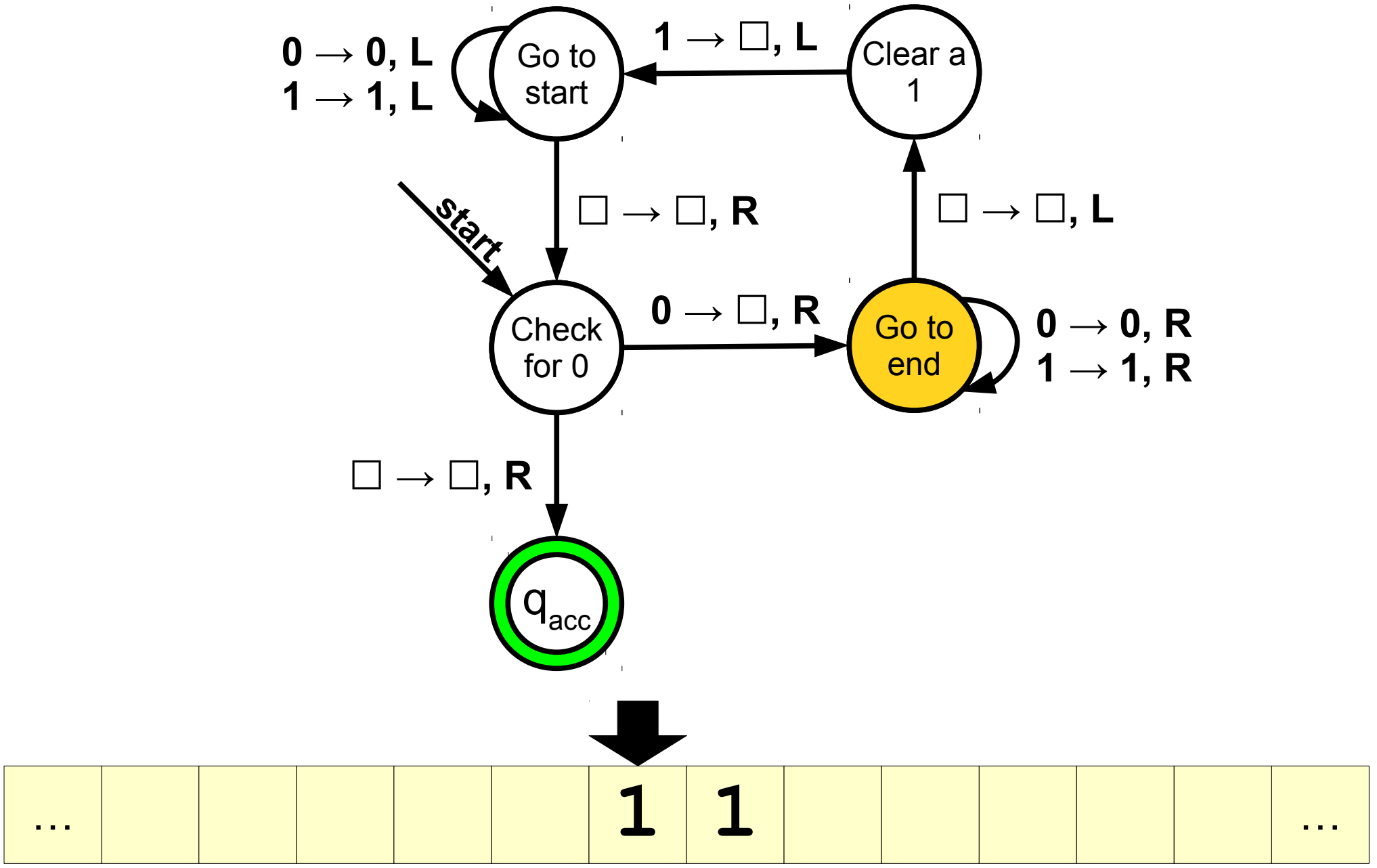

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
…

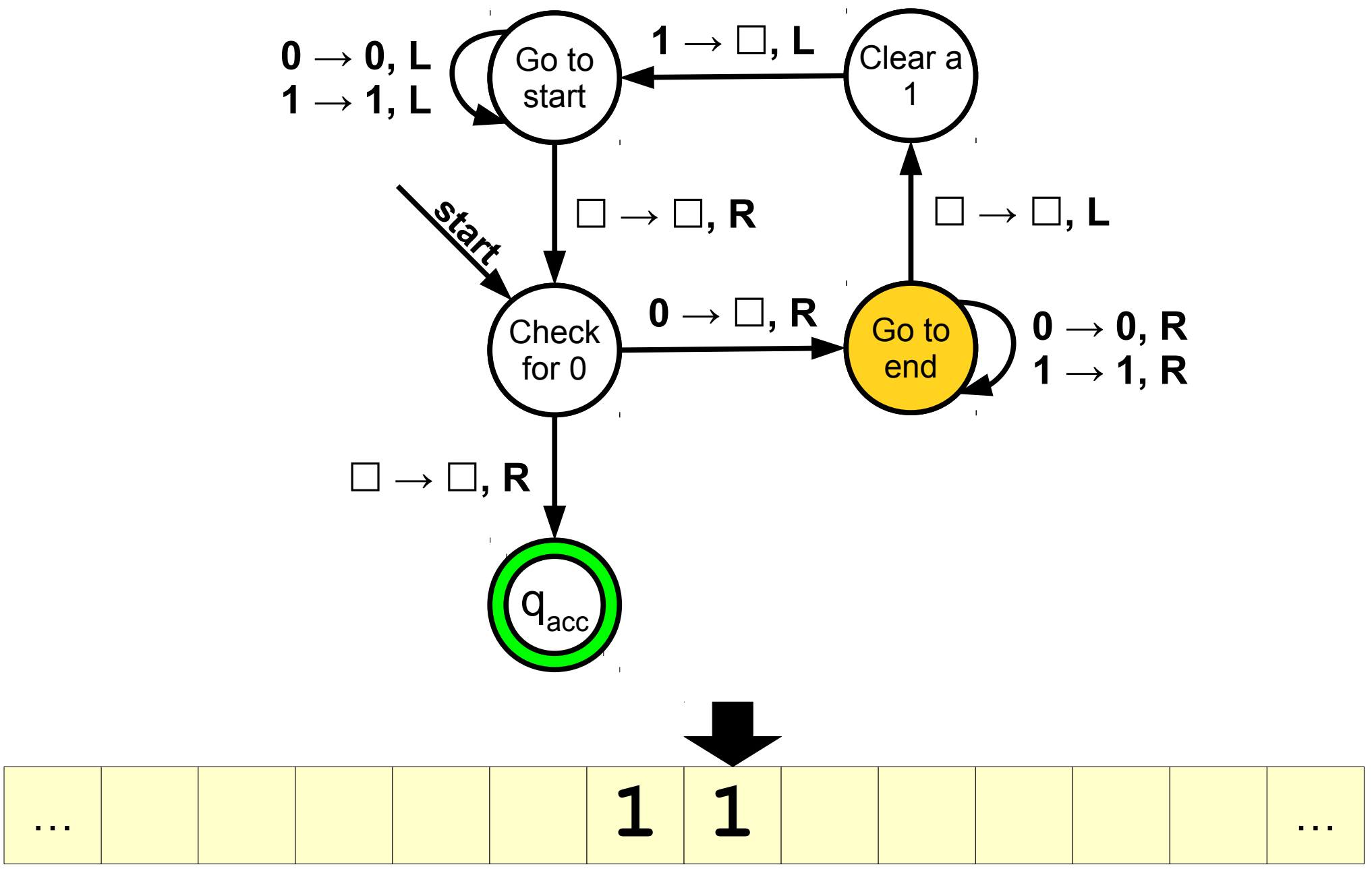

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
…

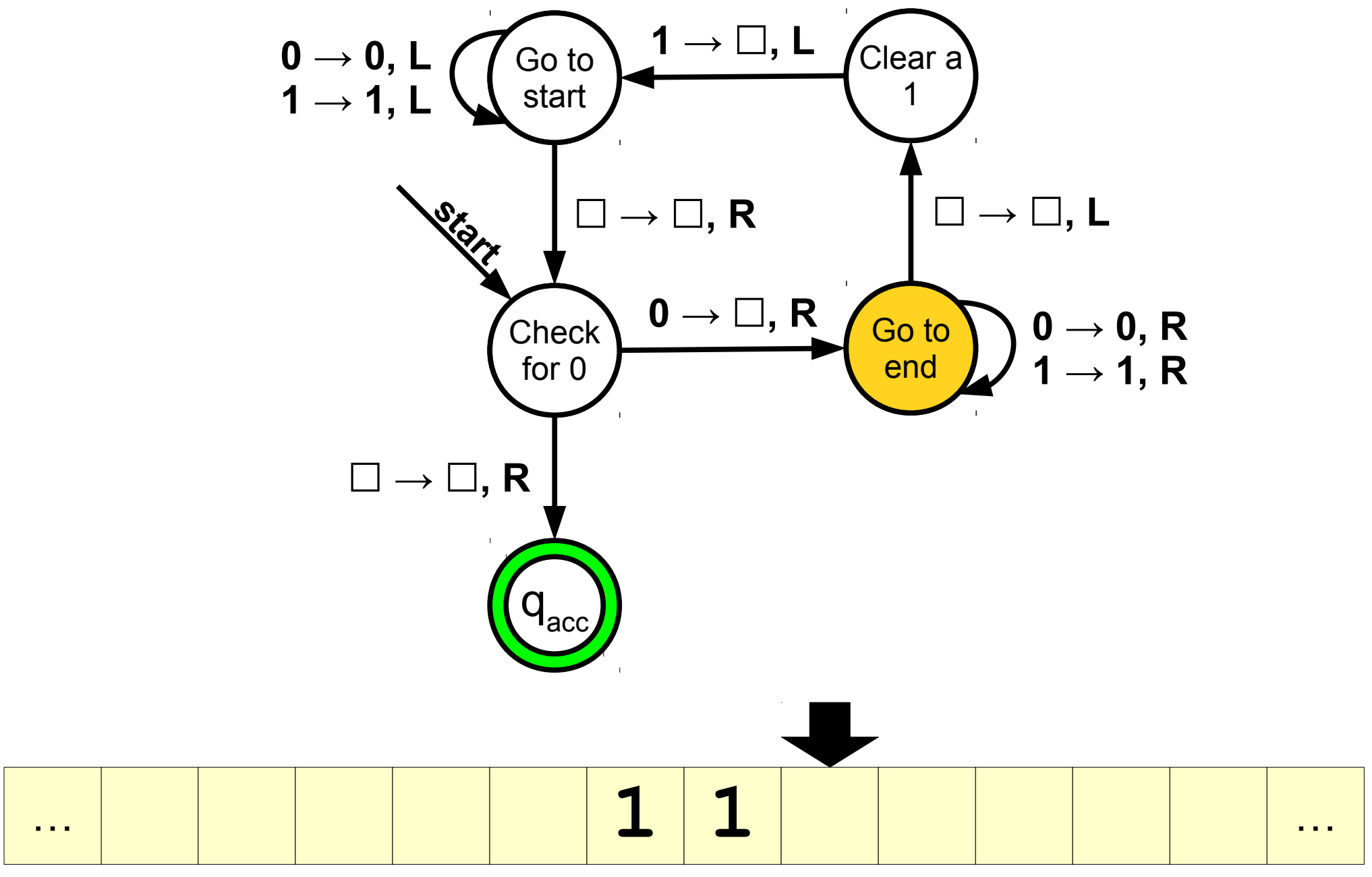

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
…

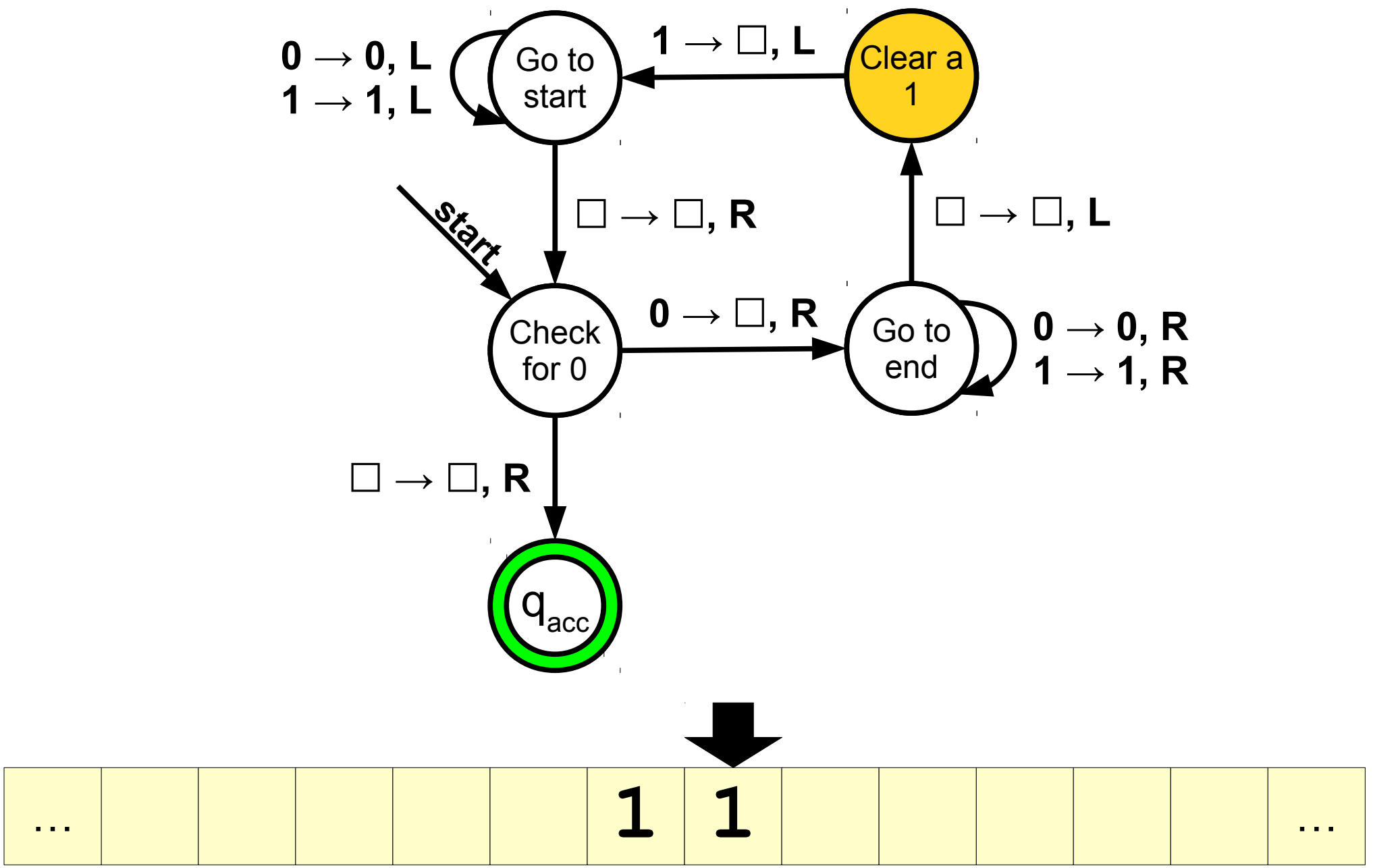

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
0
1
1
…

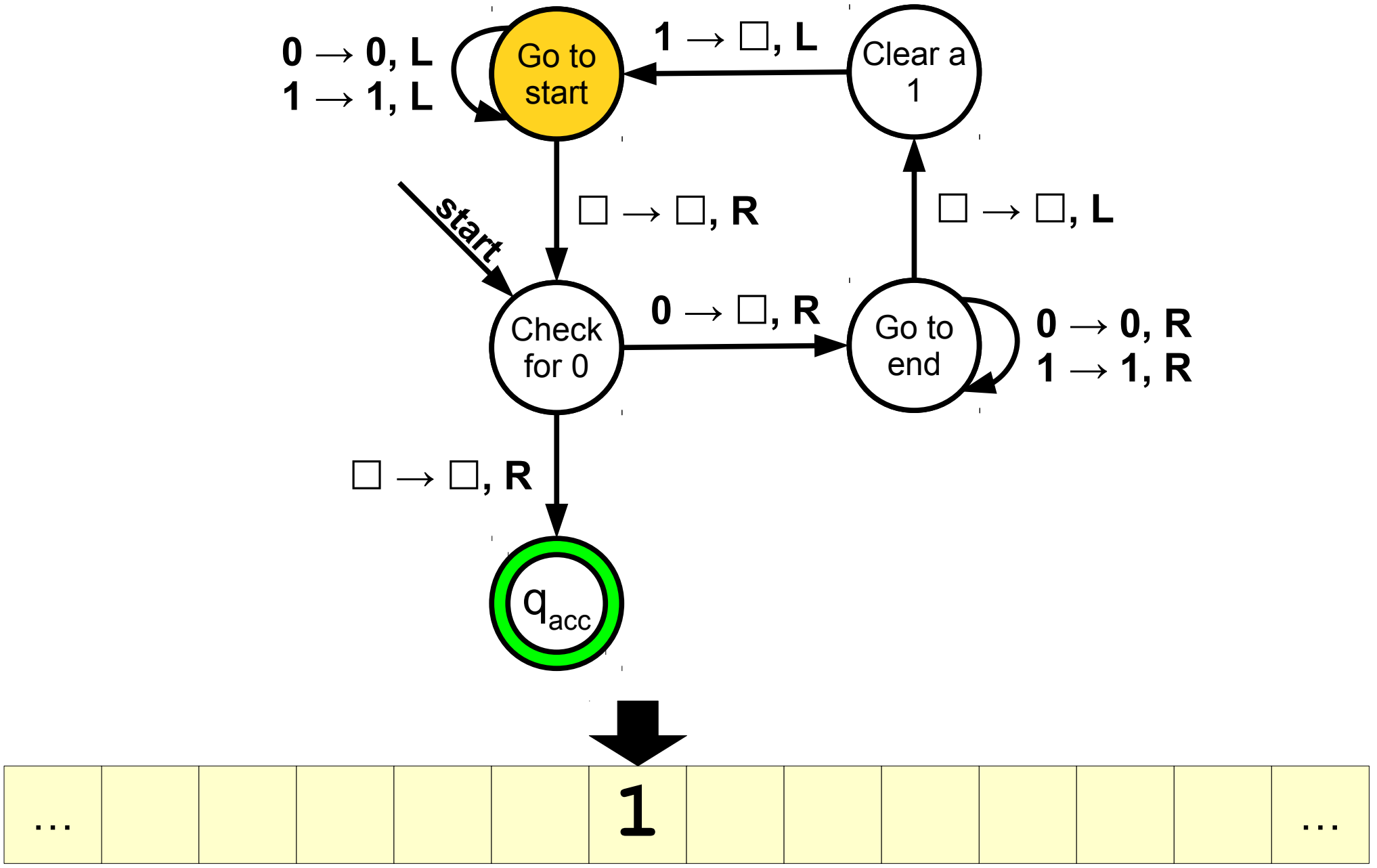

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_acc
...
0
1
1
...

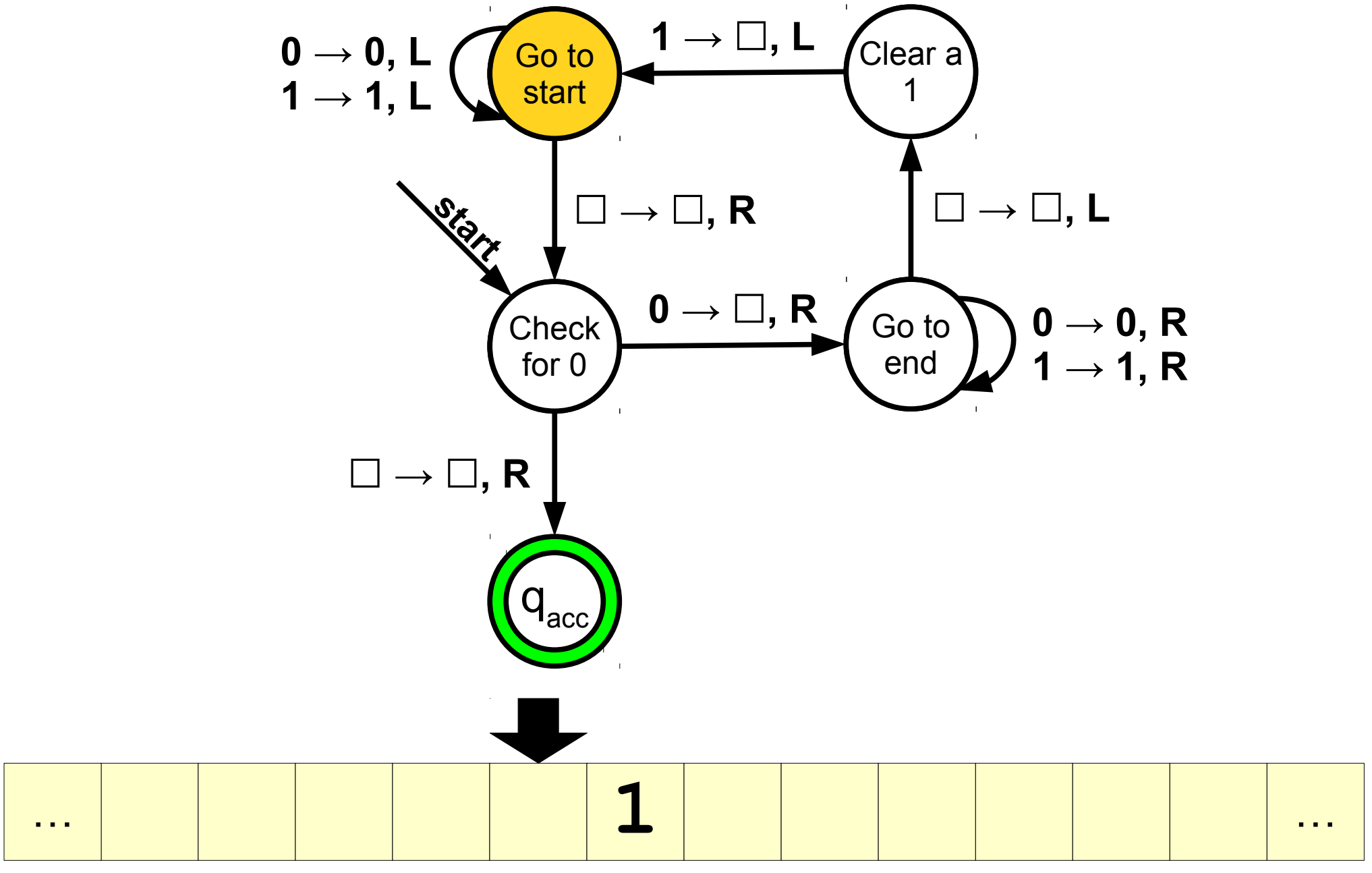

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_acc
...
1
1
...

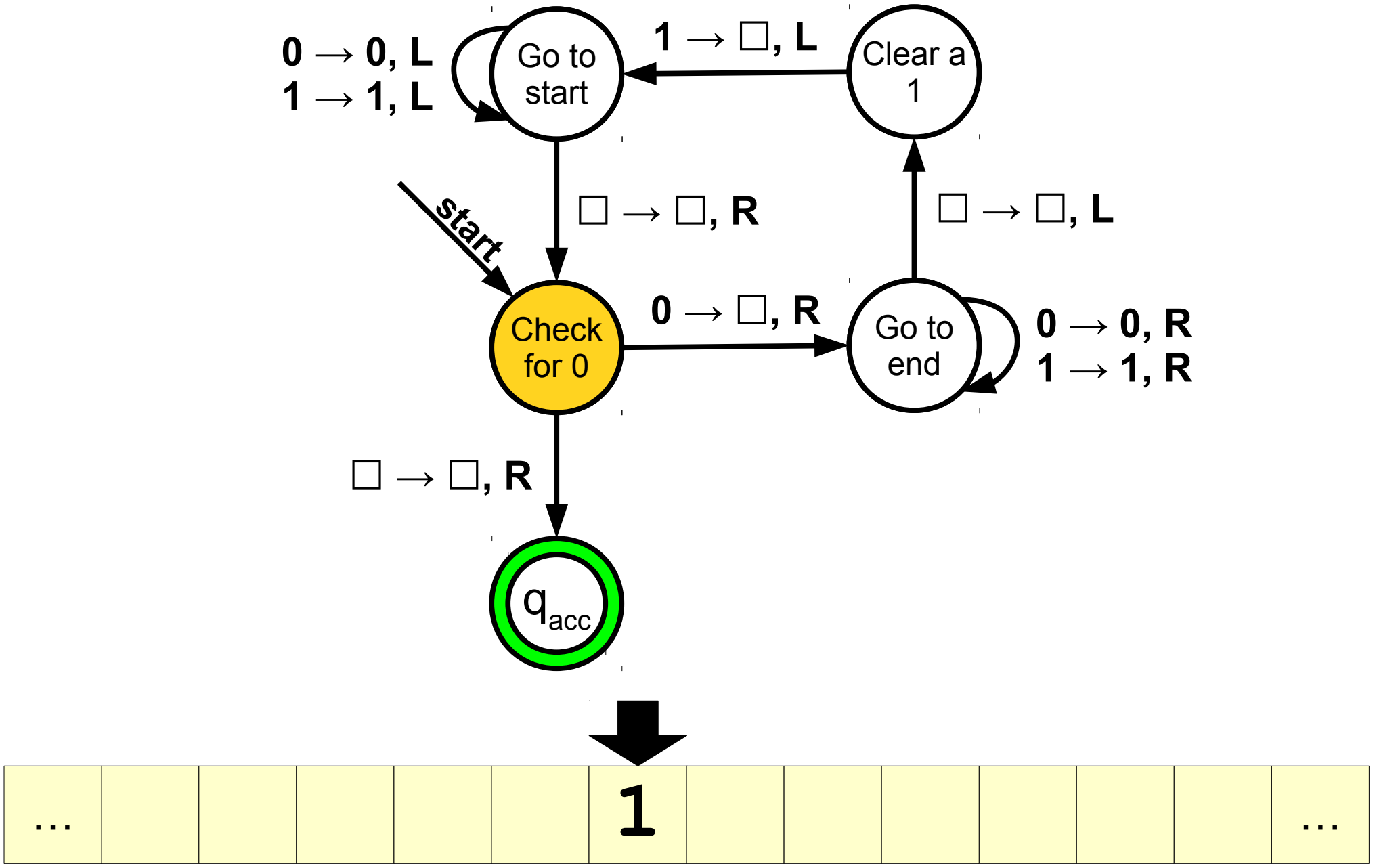

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
1
1
…

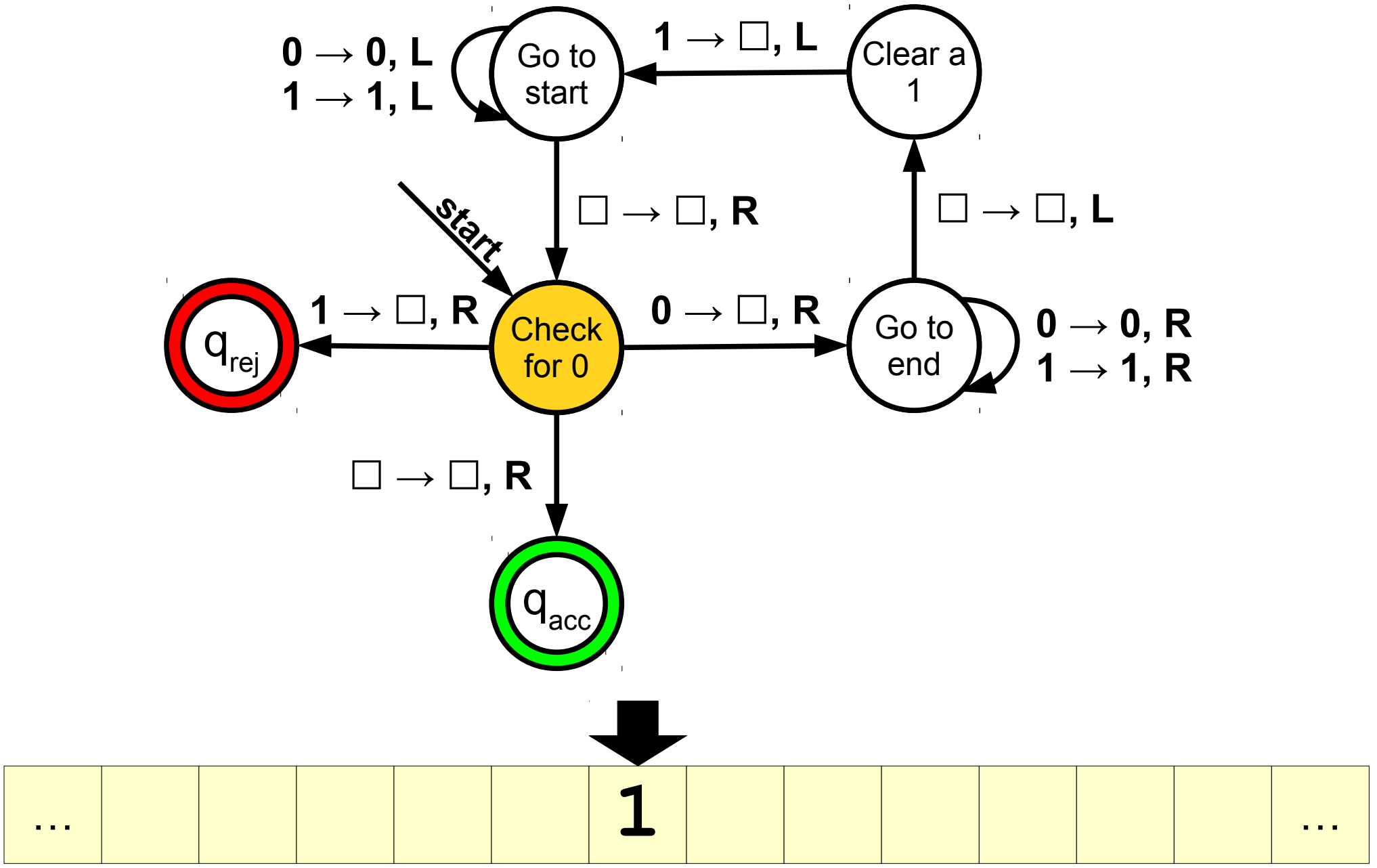

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
…
1
1
…

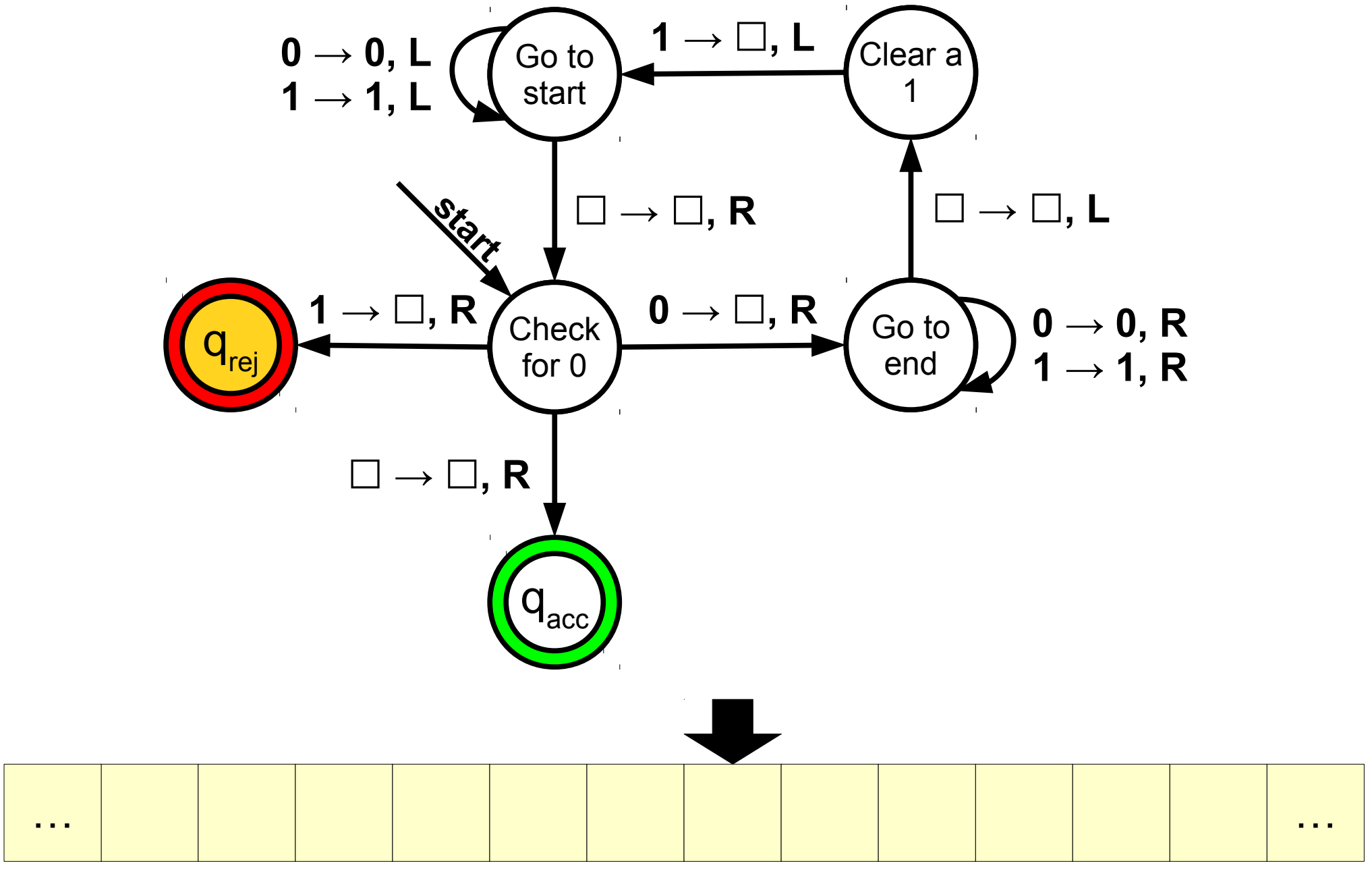

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{acc}
...
1
1
...

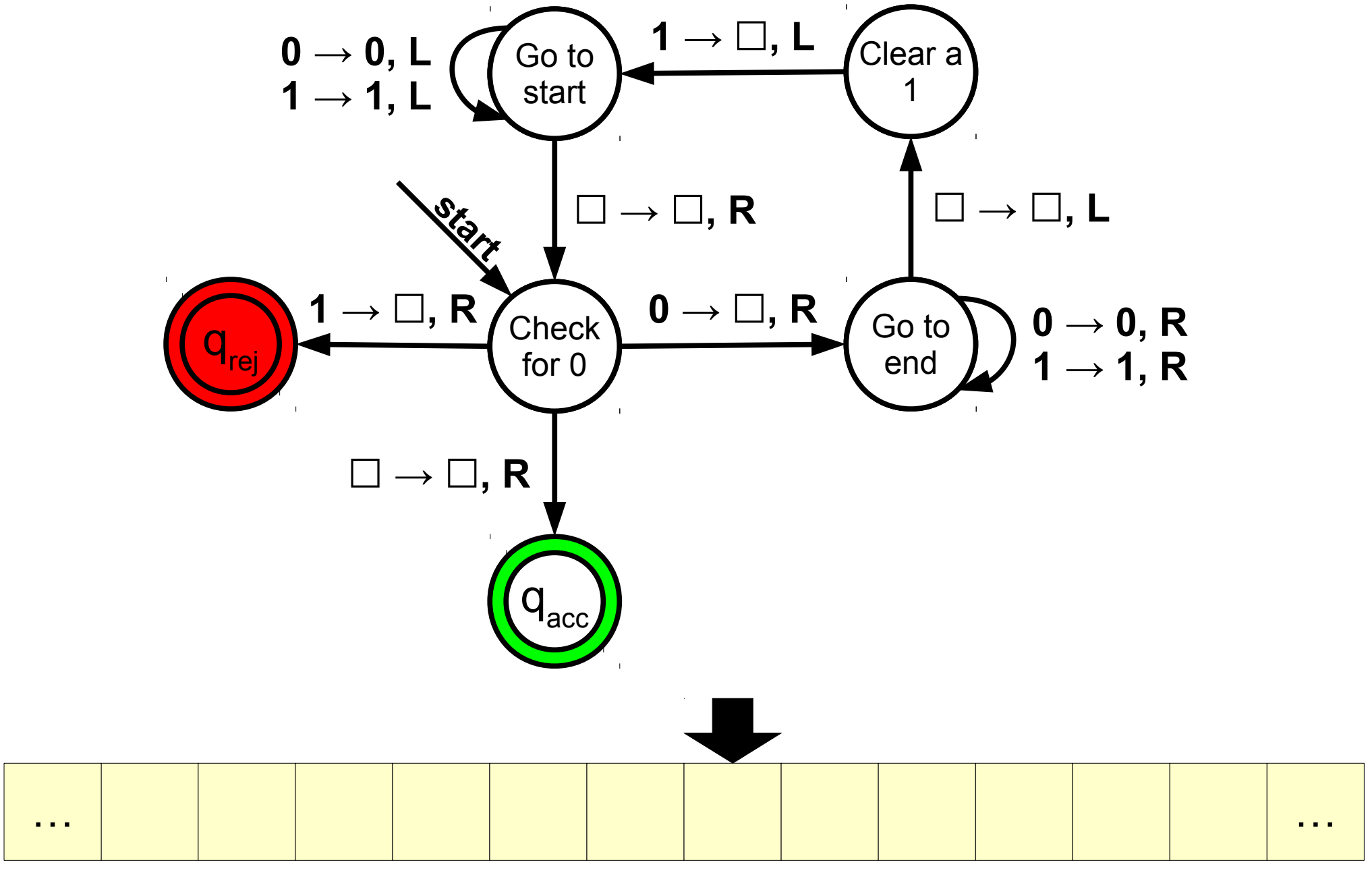

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_acc
...
1
...

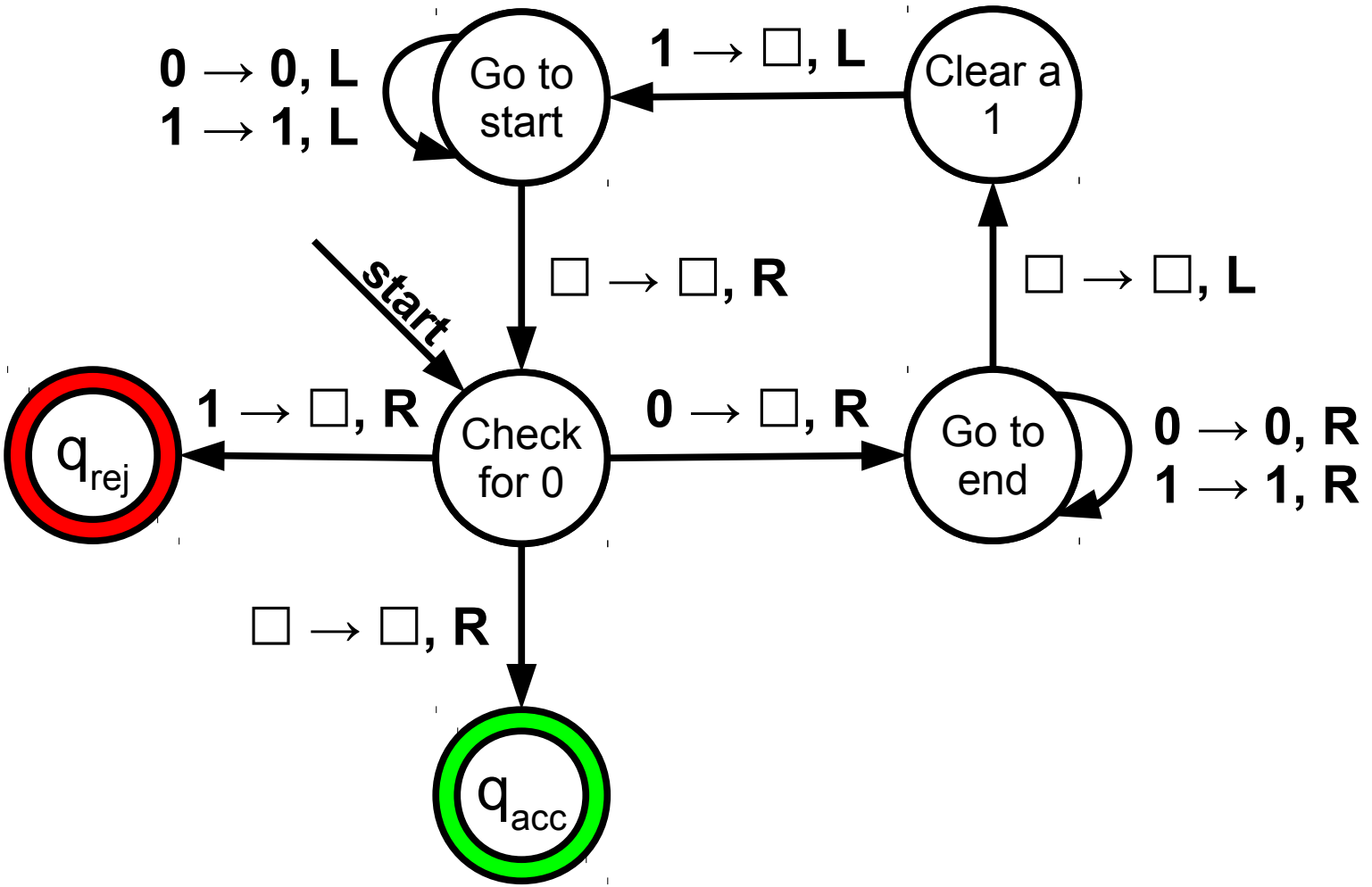

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_acc
...
1
...

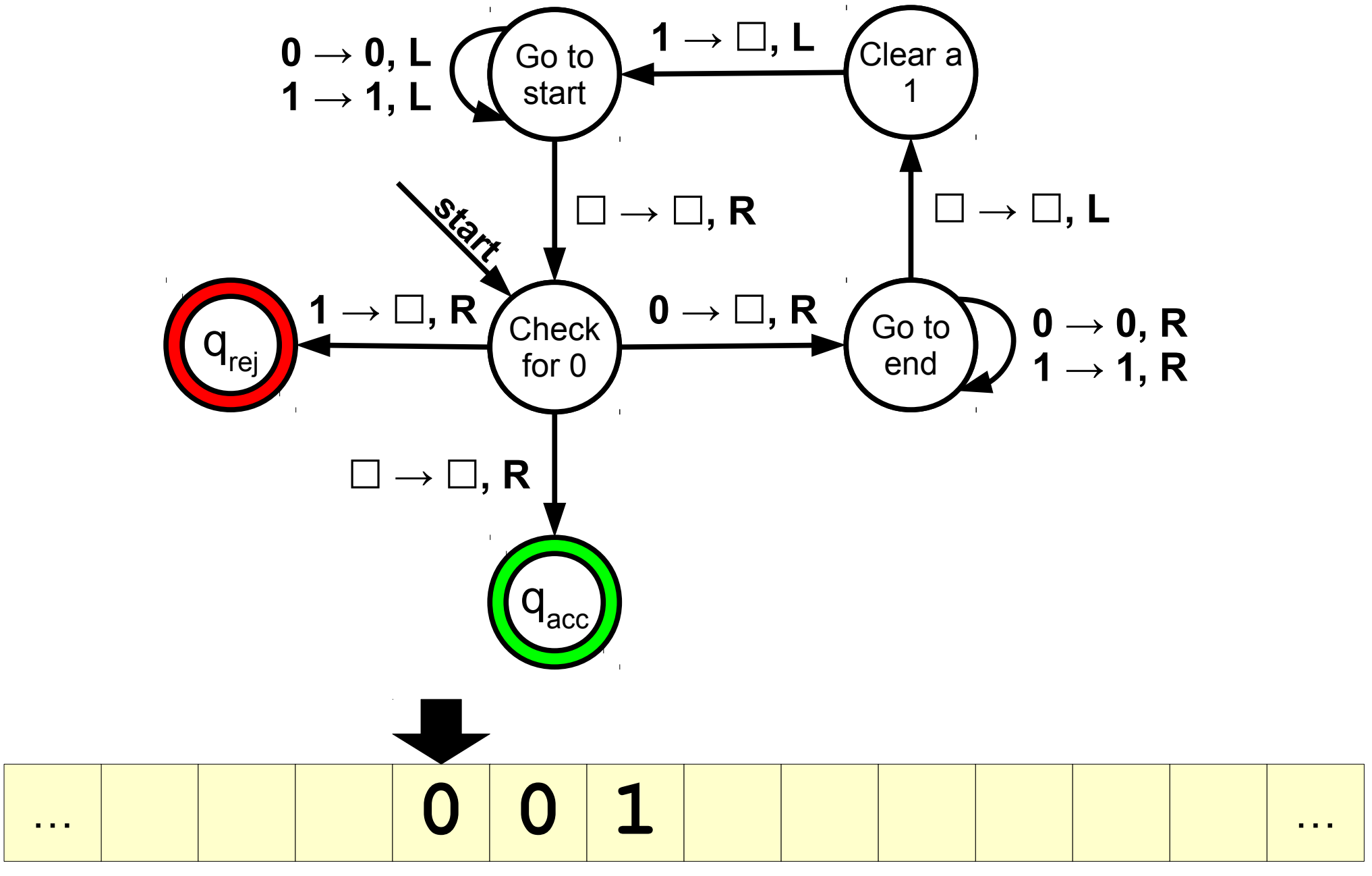

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_acc
...
1
...

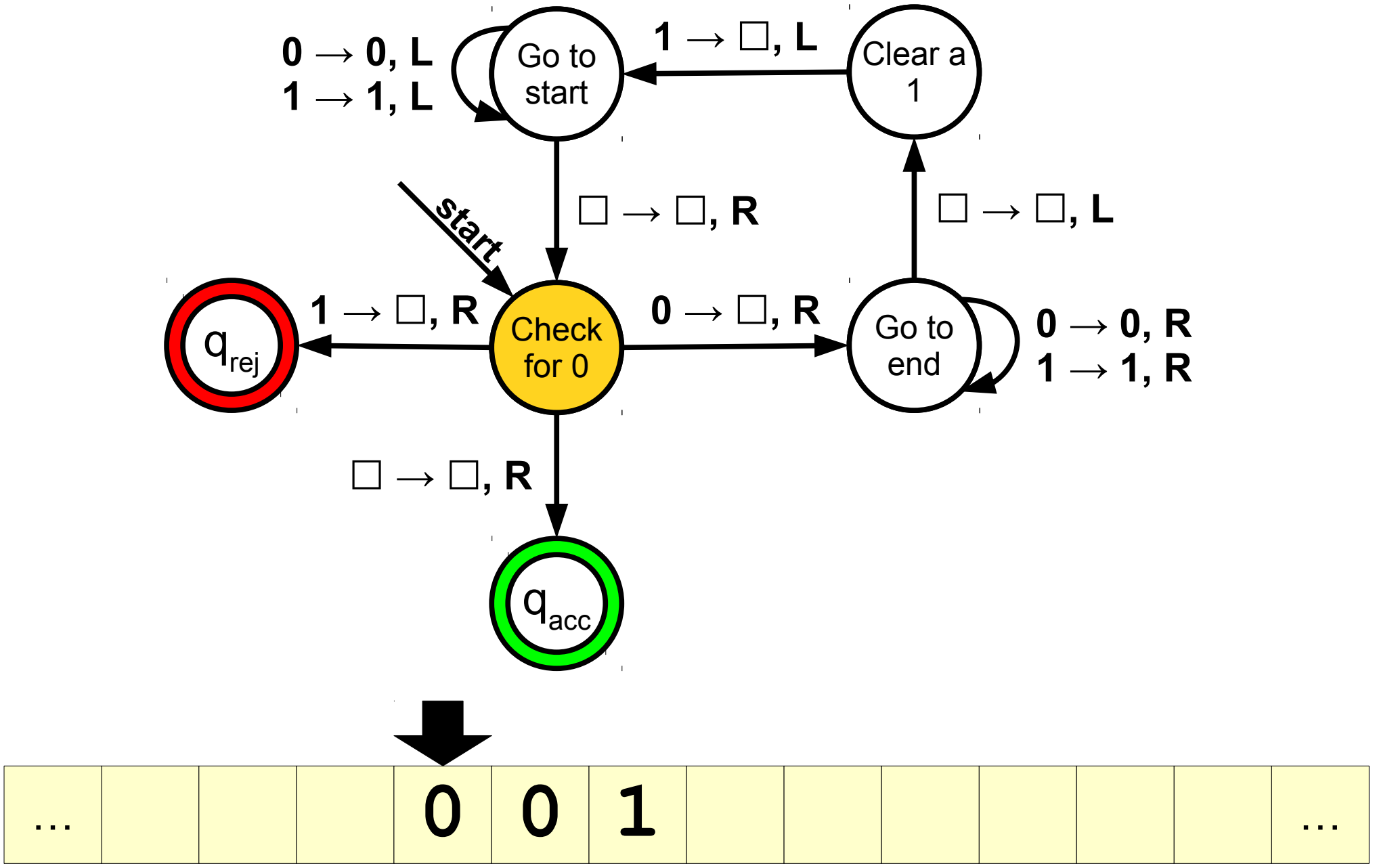

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
...

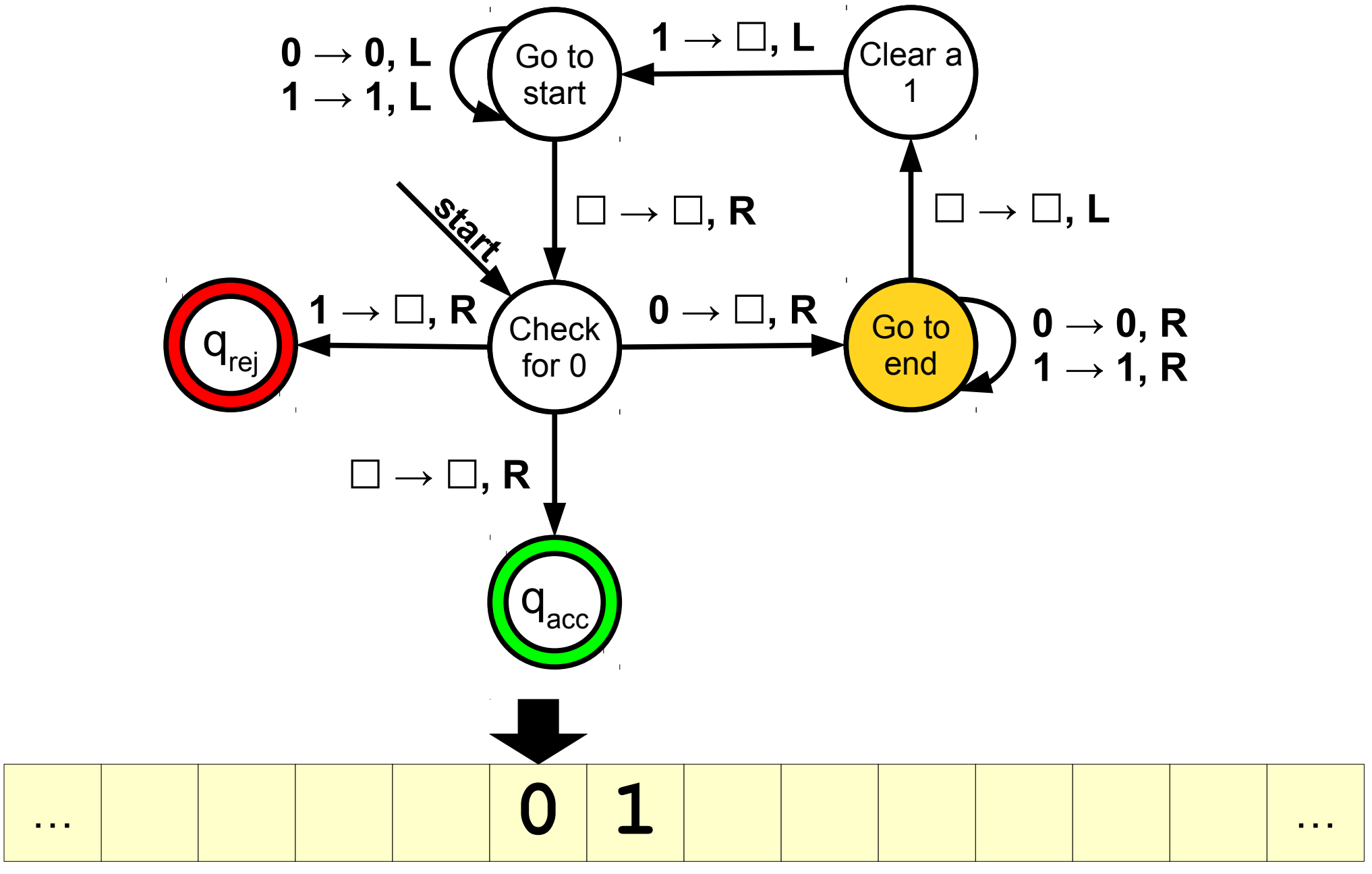

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

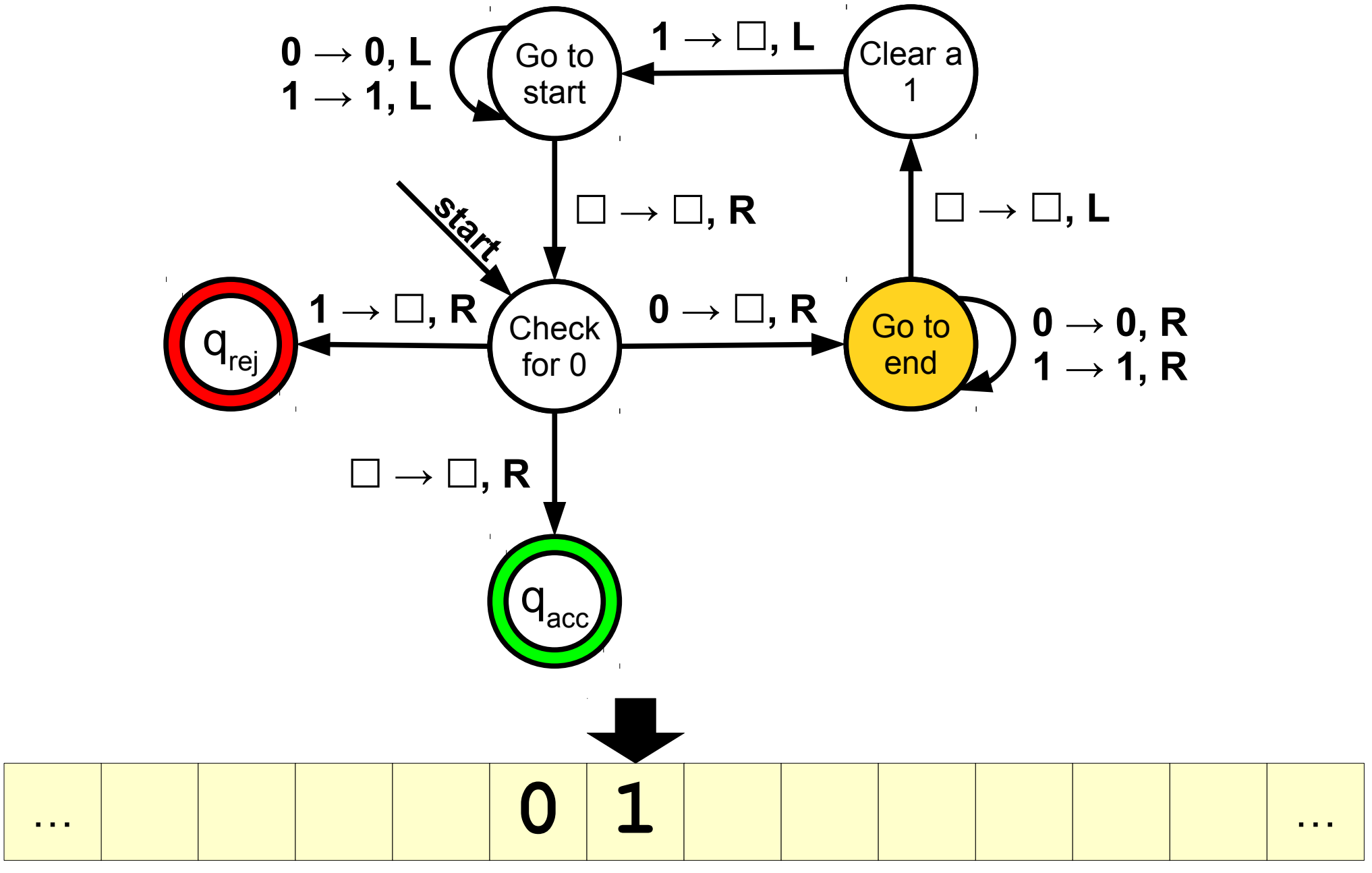

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

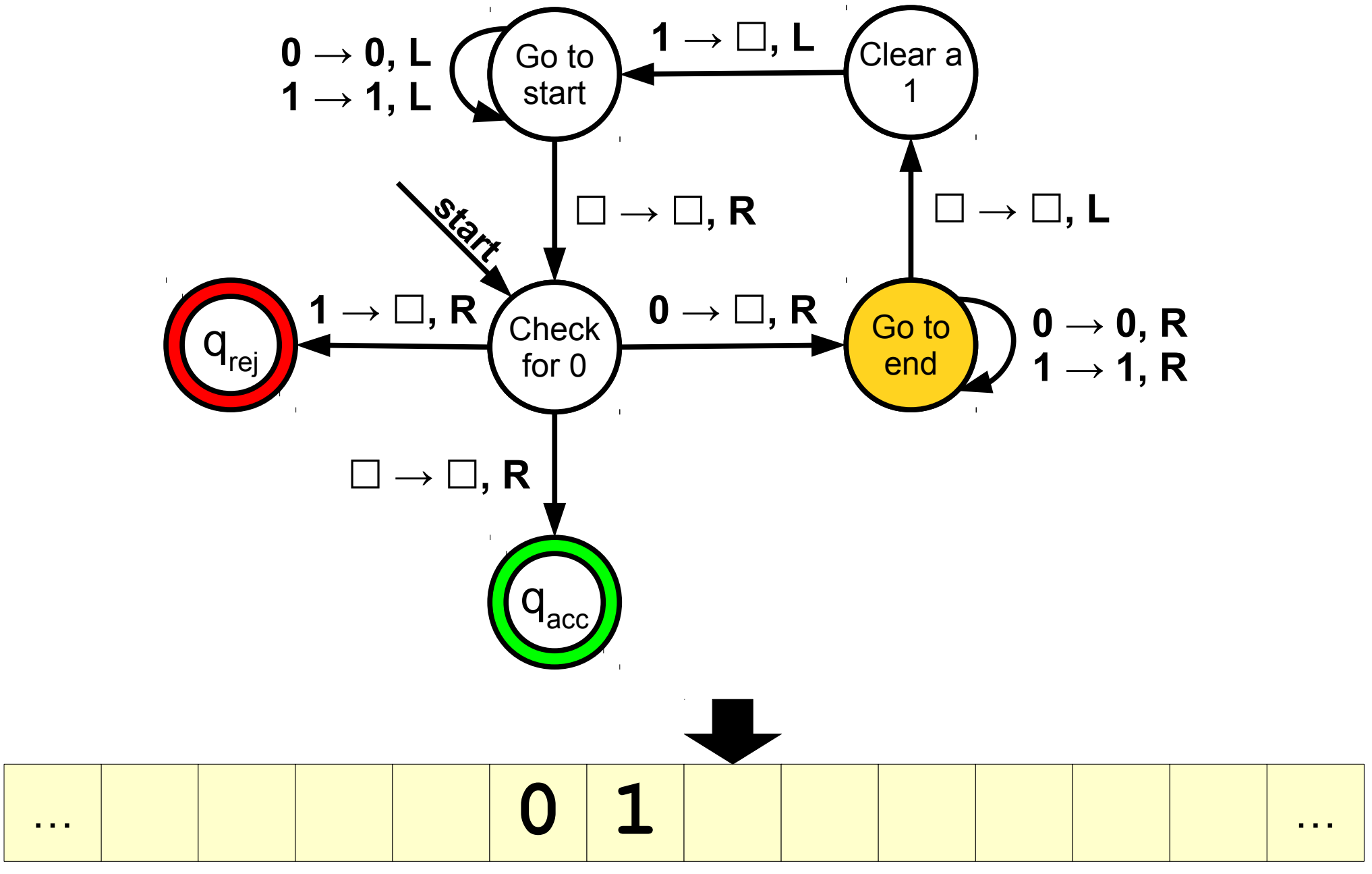

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc

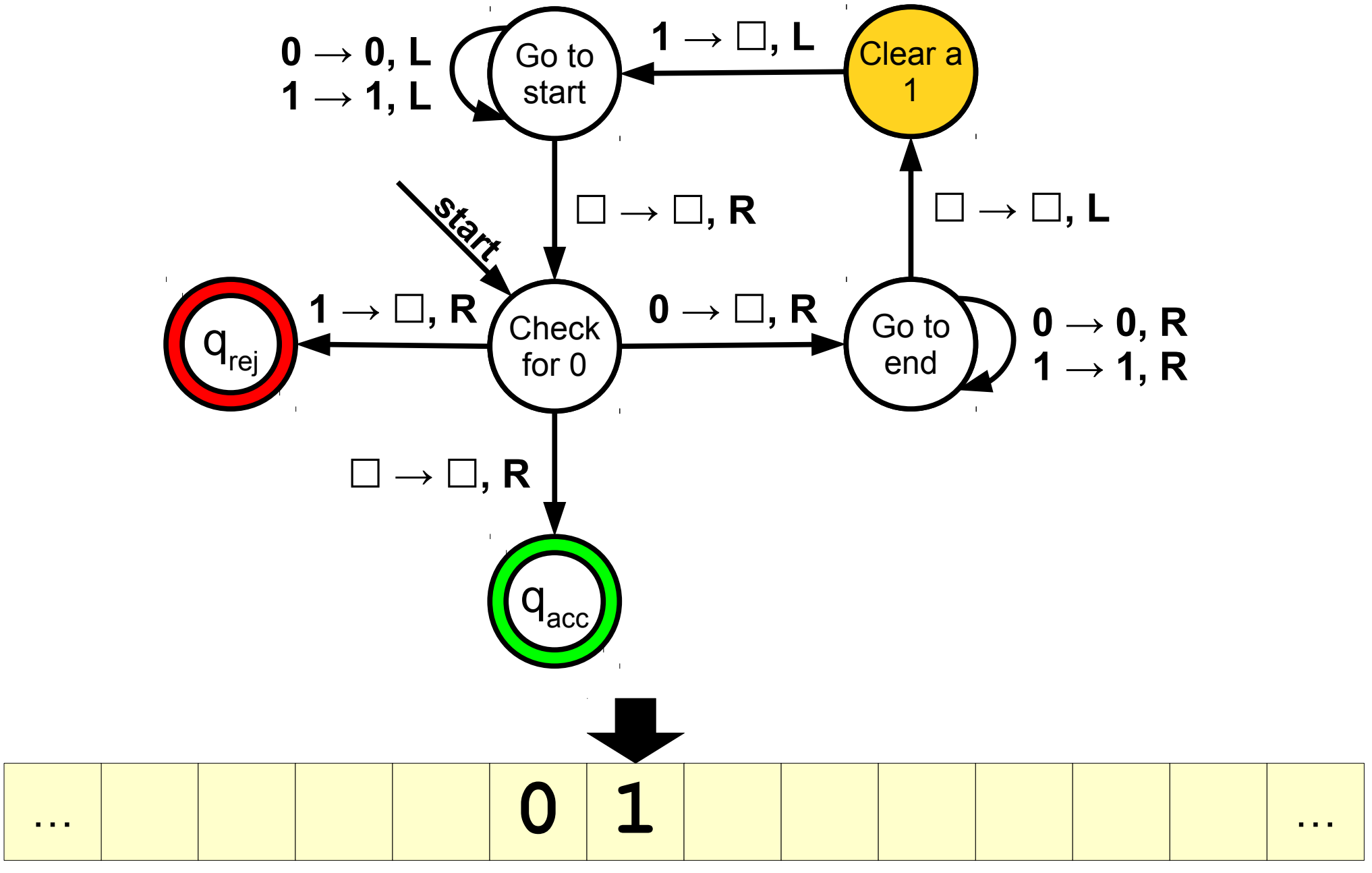

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
0
1
...

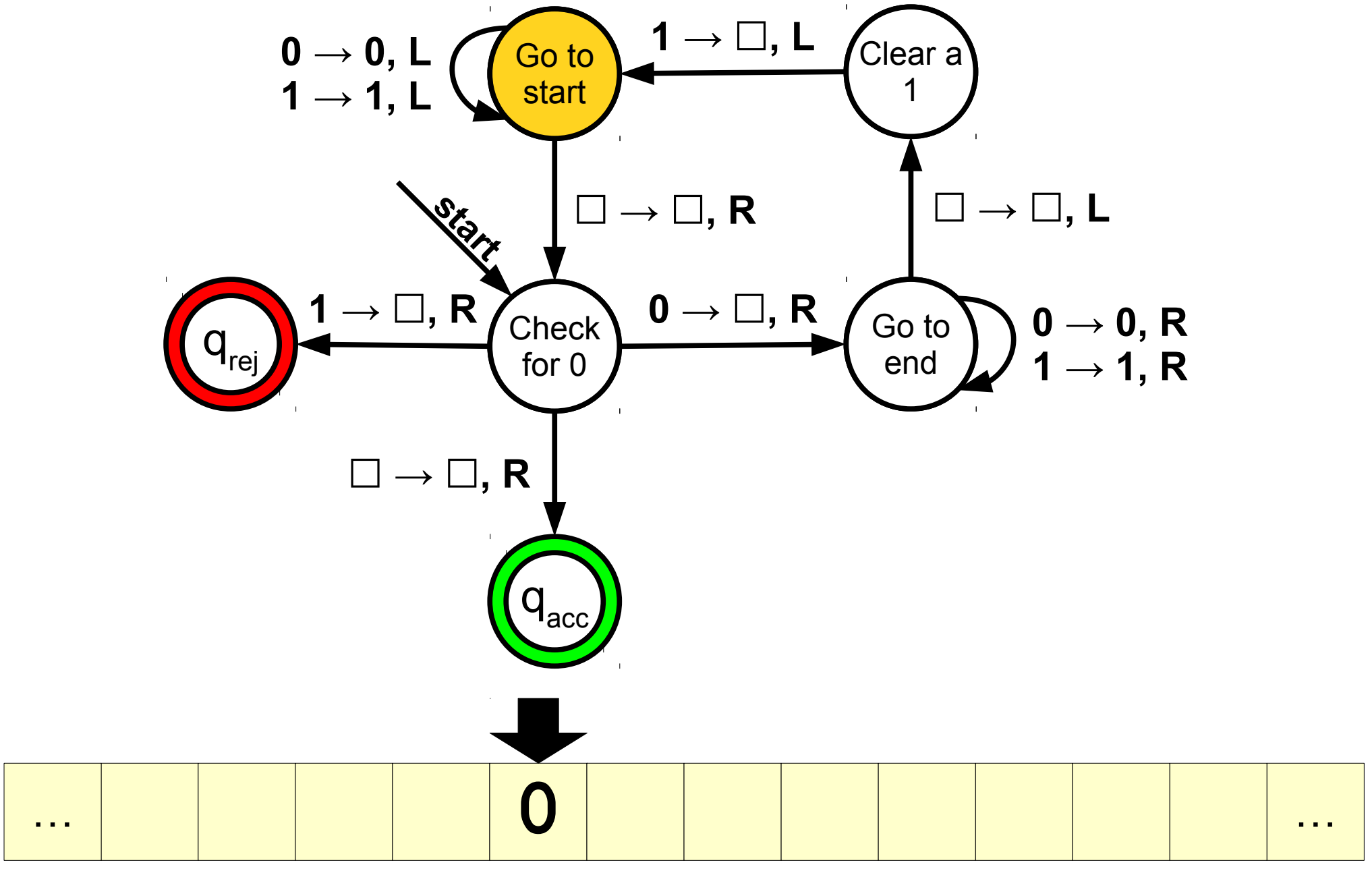

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
0
1
...

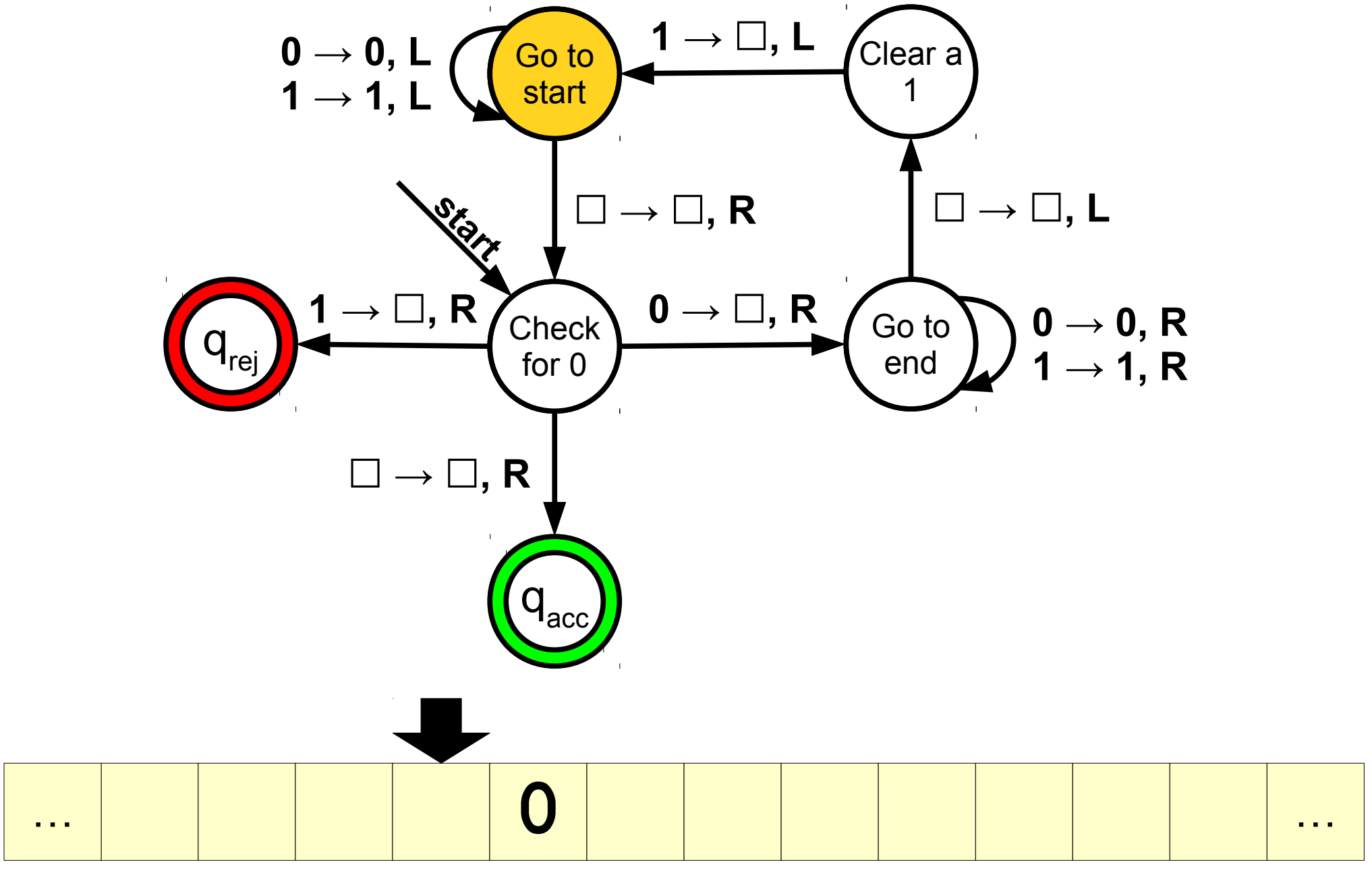

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
1
...

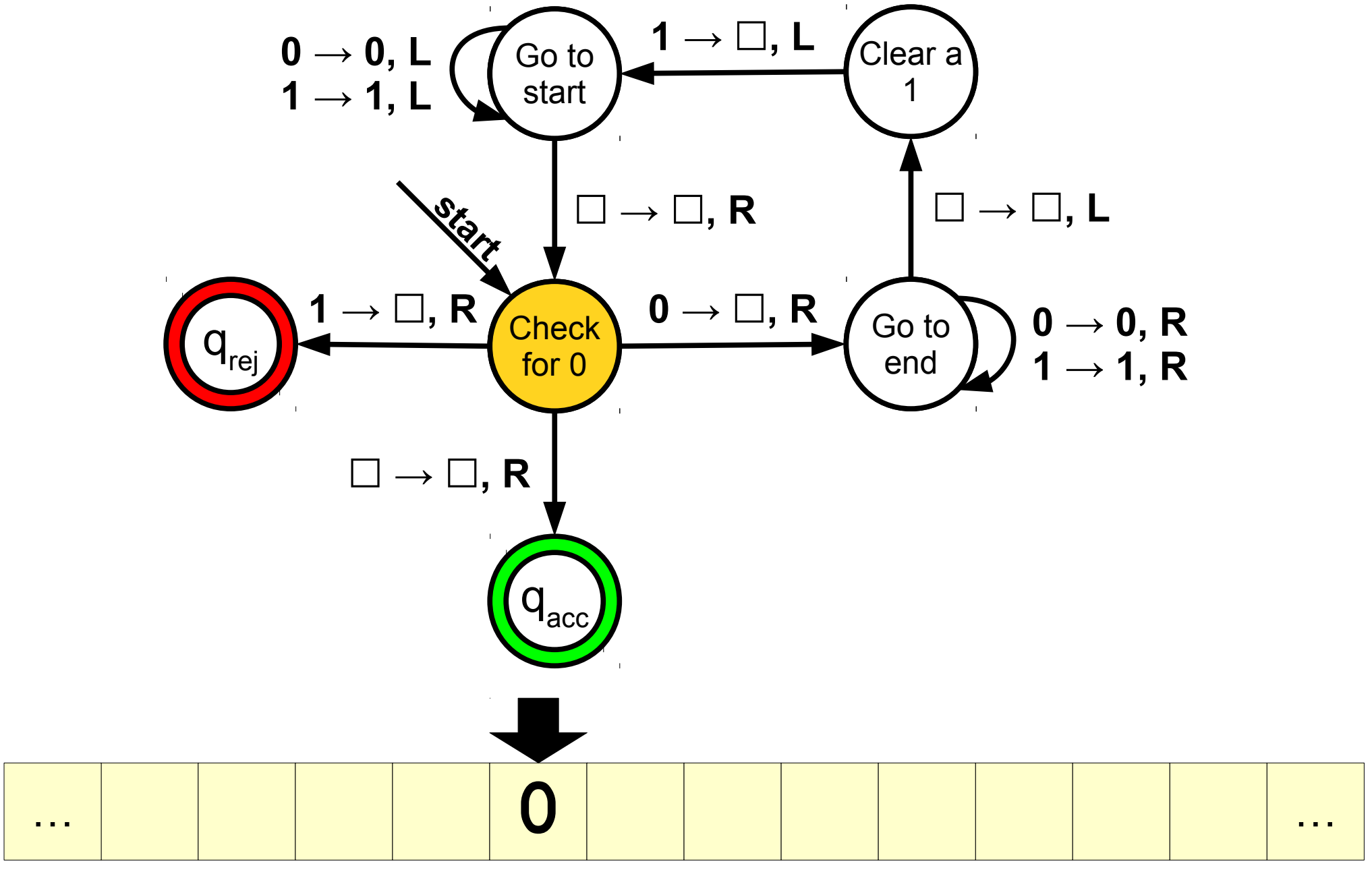

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
1
...

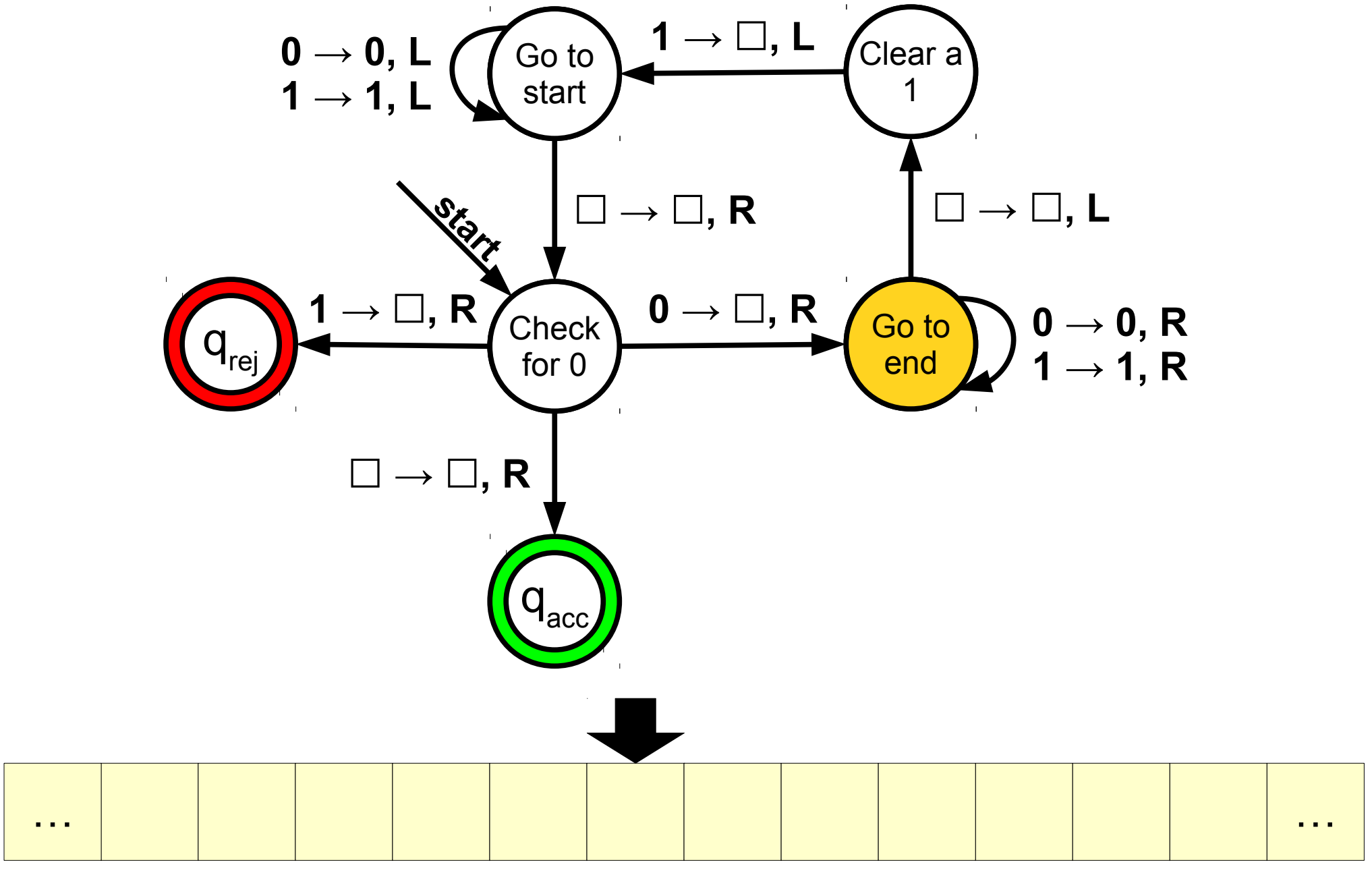

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
1
...

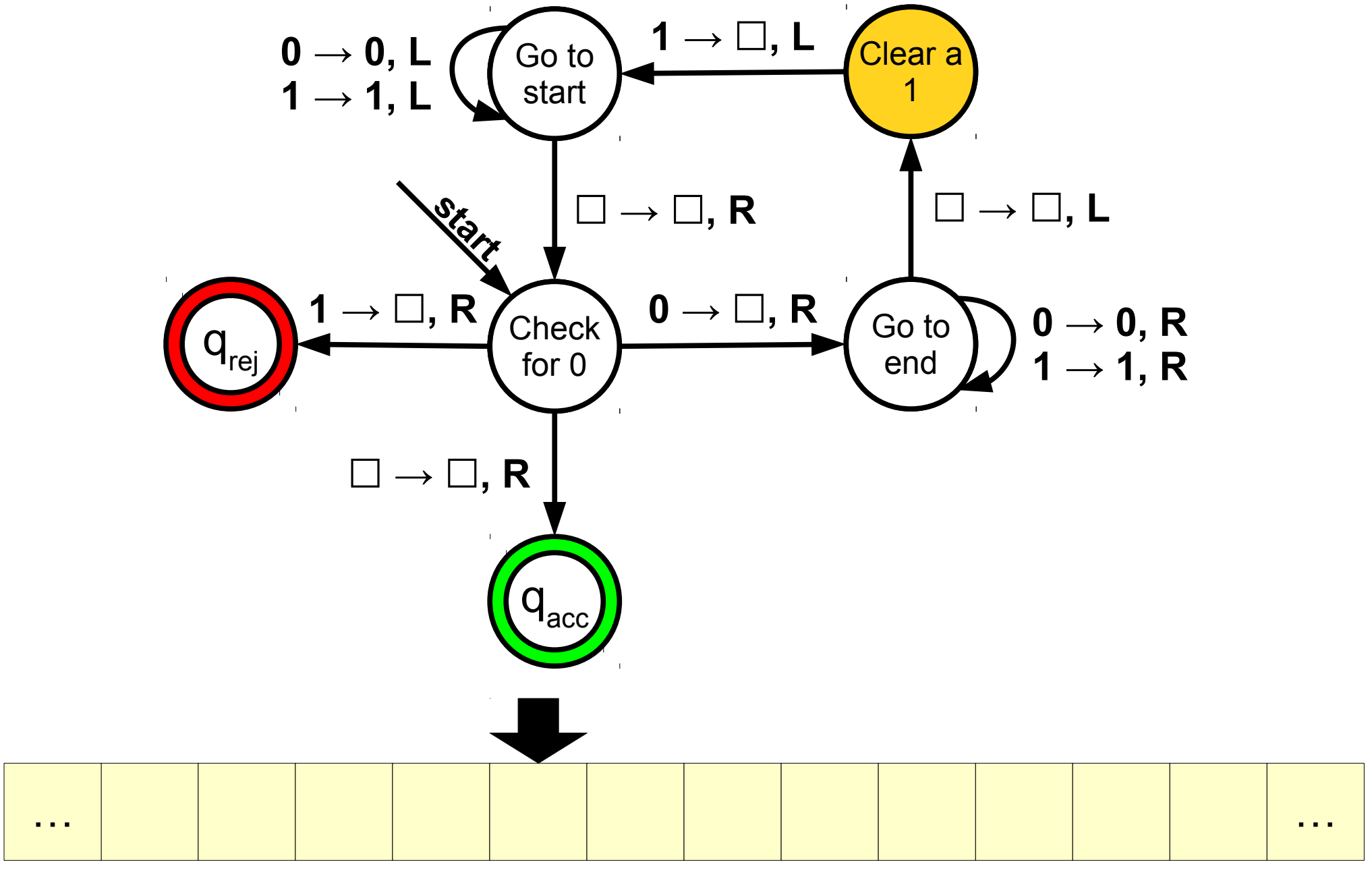

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
1
...

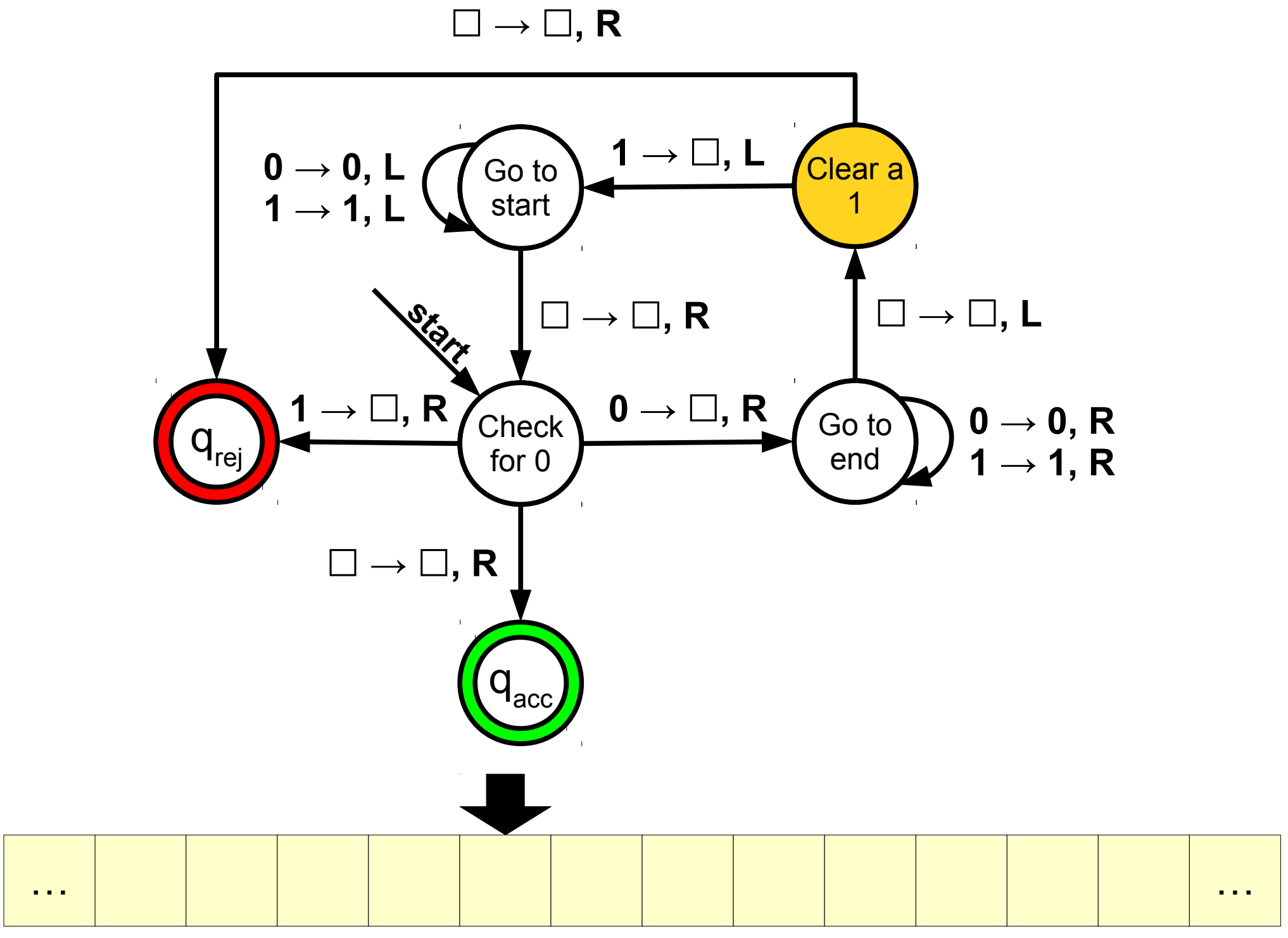

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
...

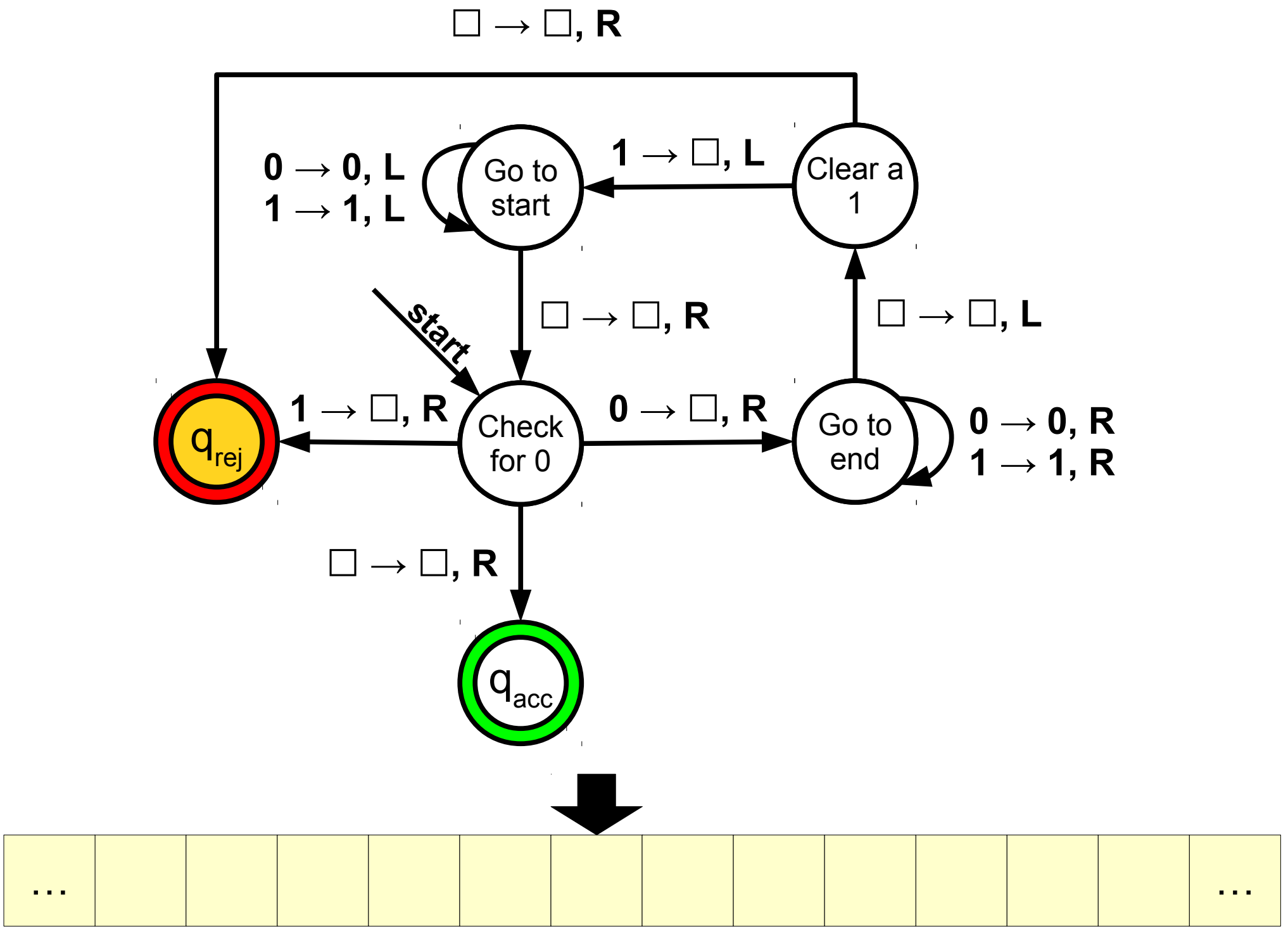

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
…
0
…

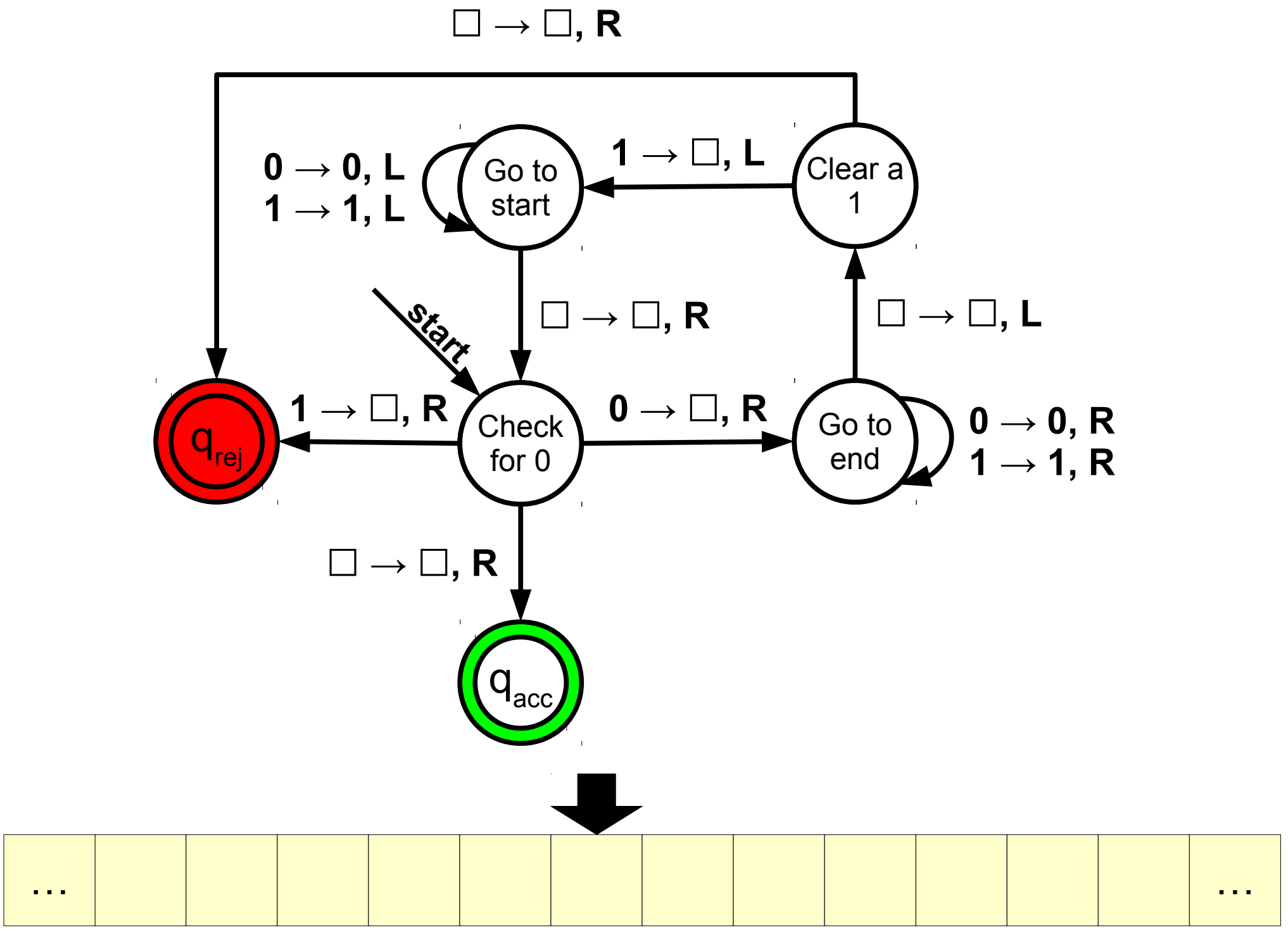

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0
...

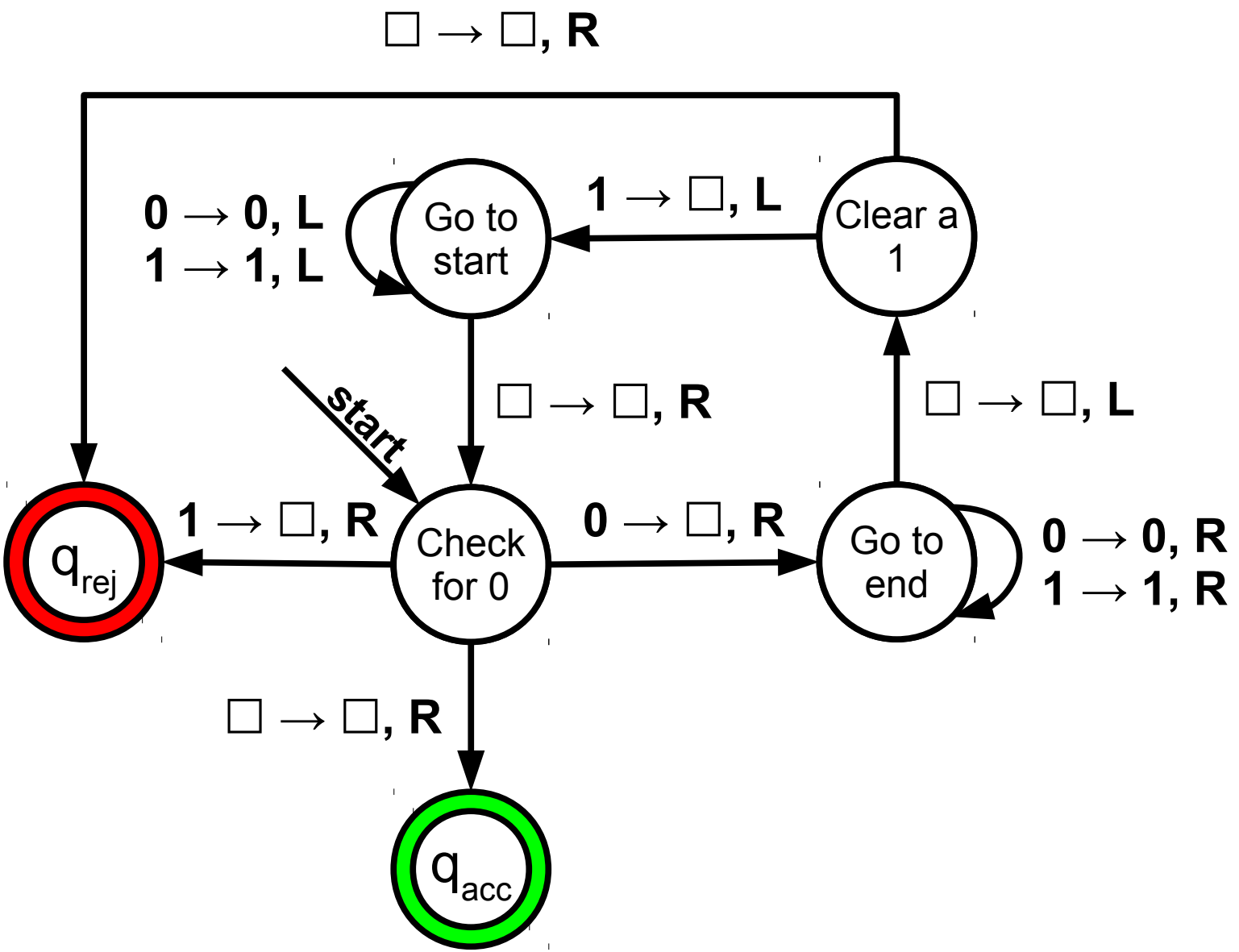

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

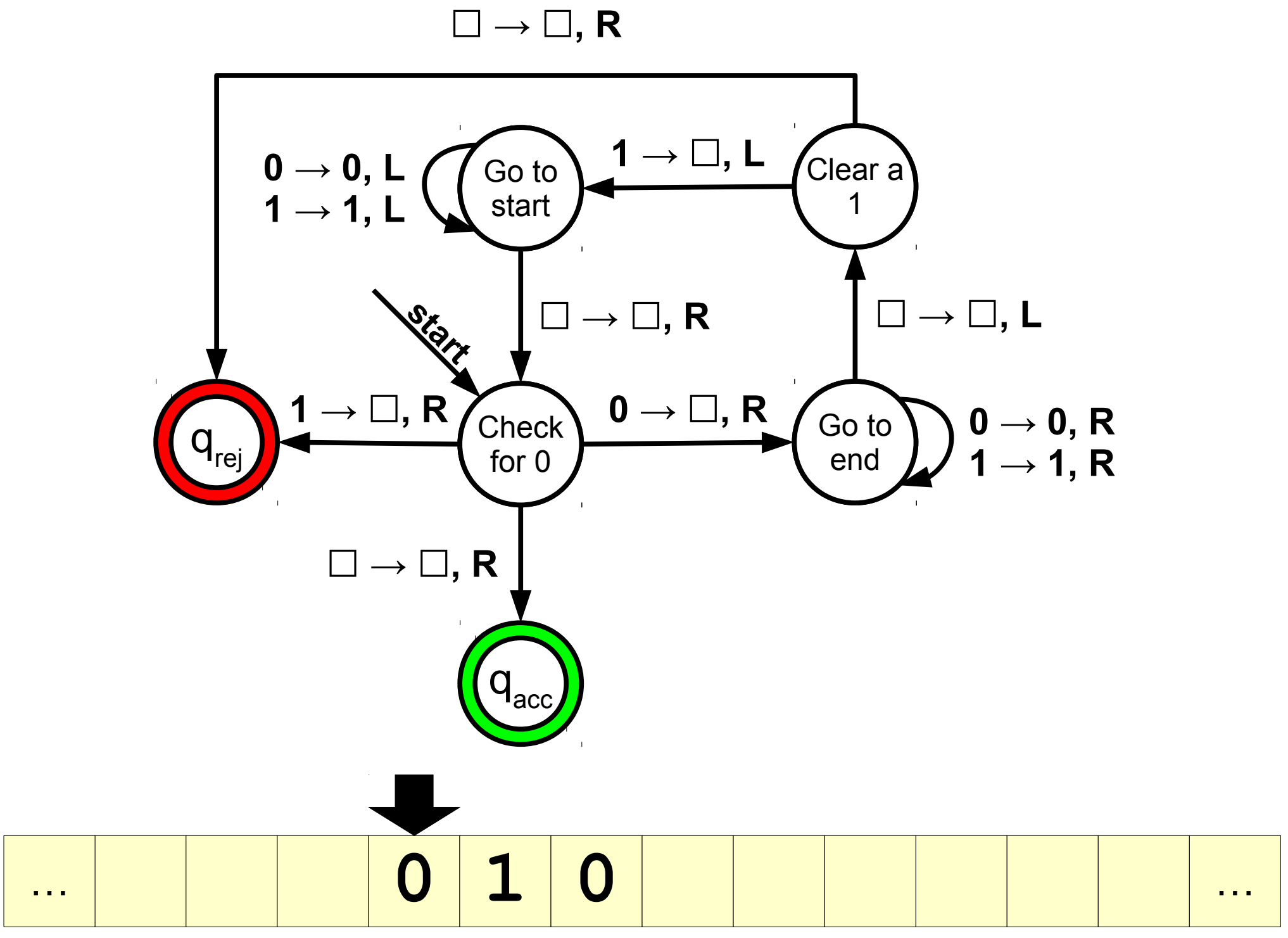

0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

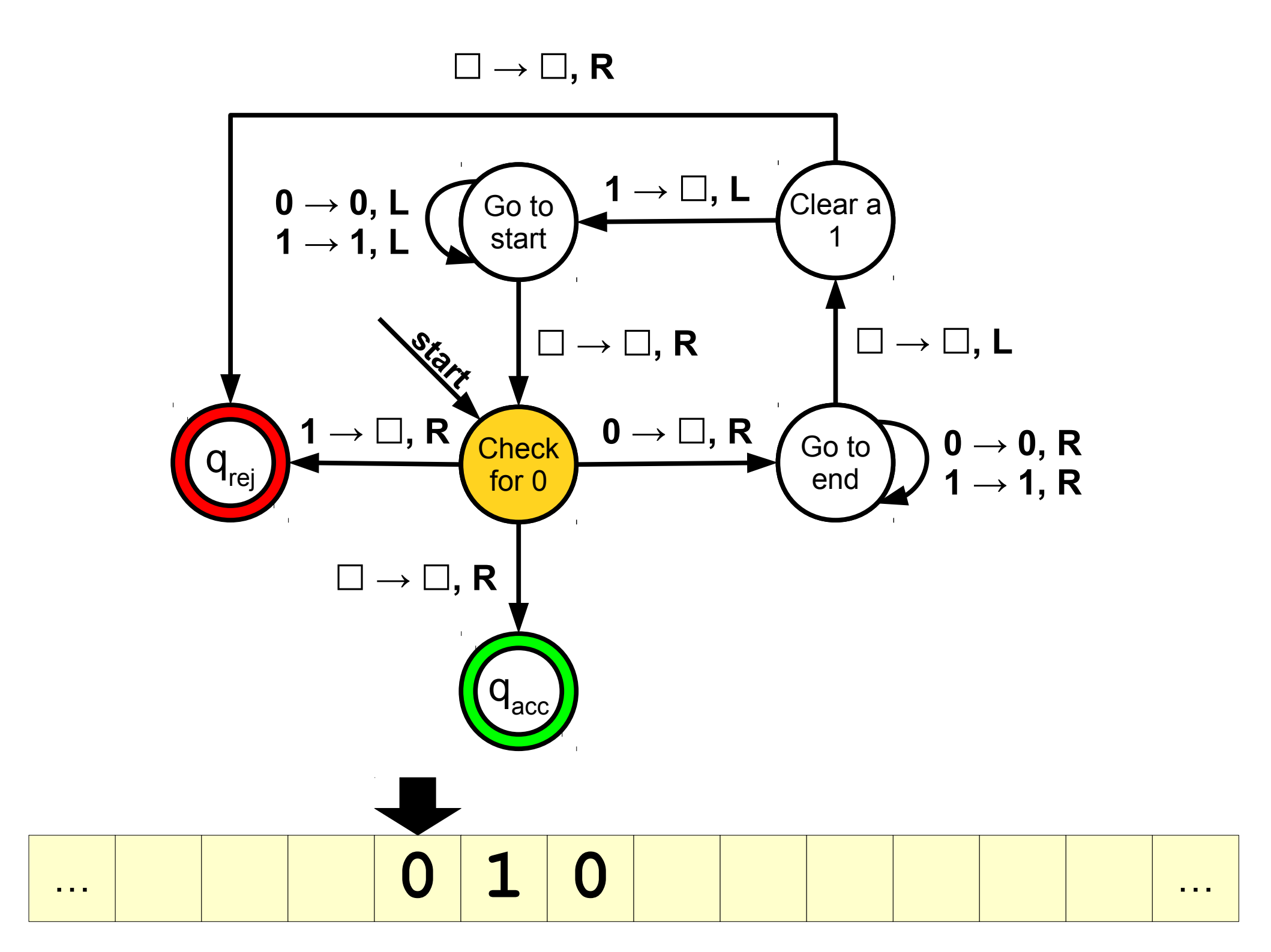

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

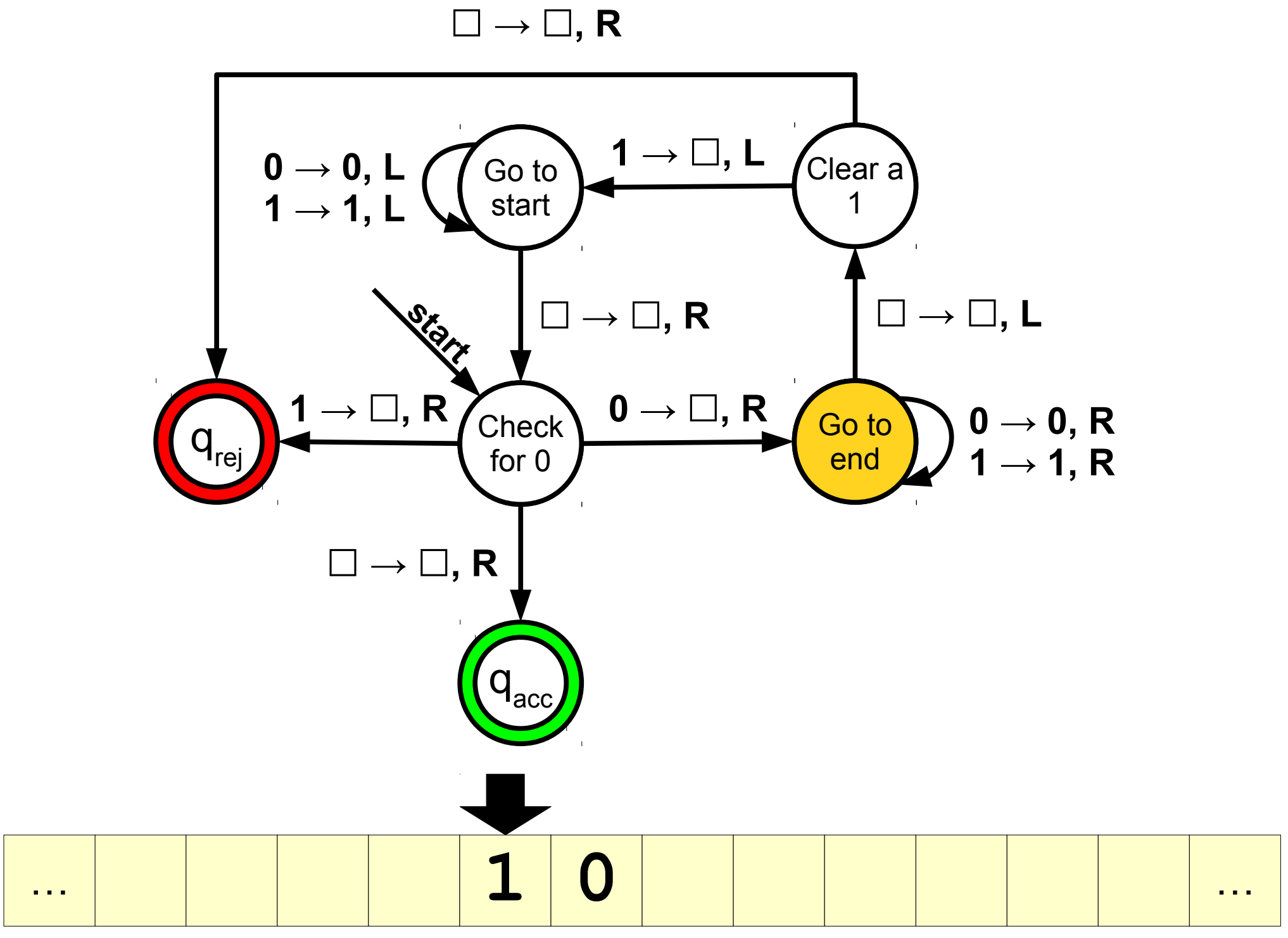

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

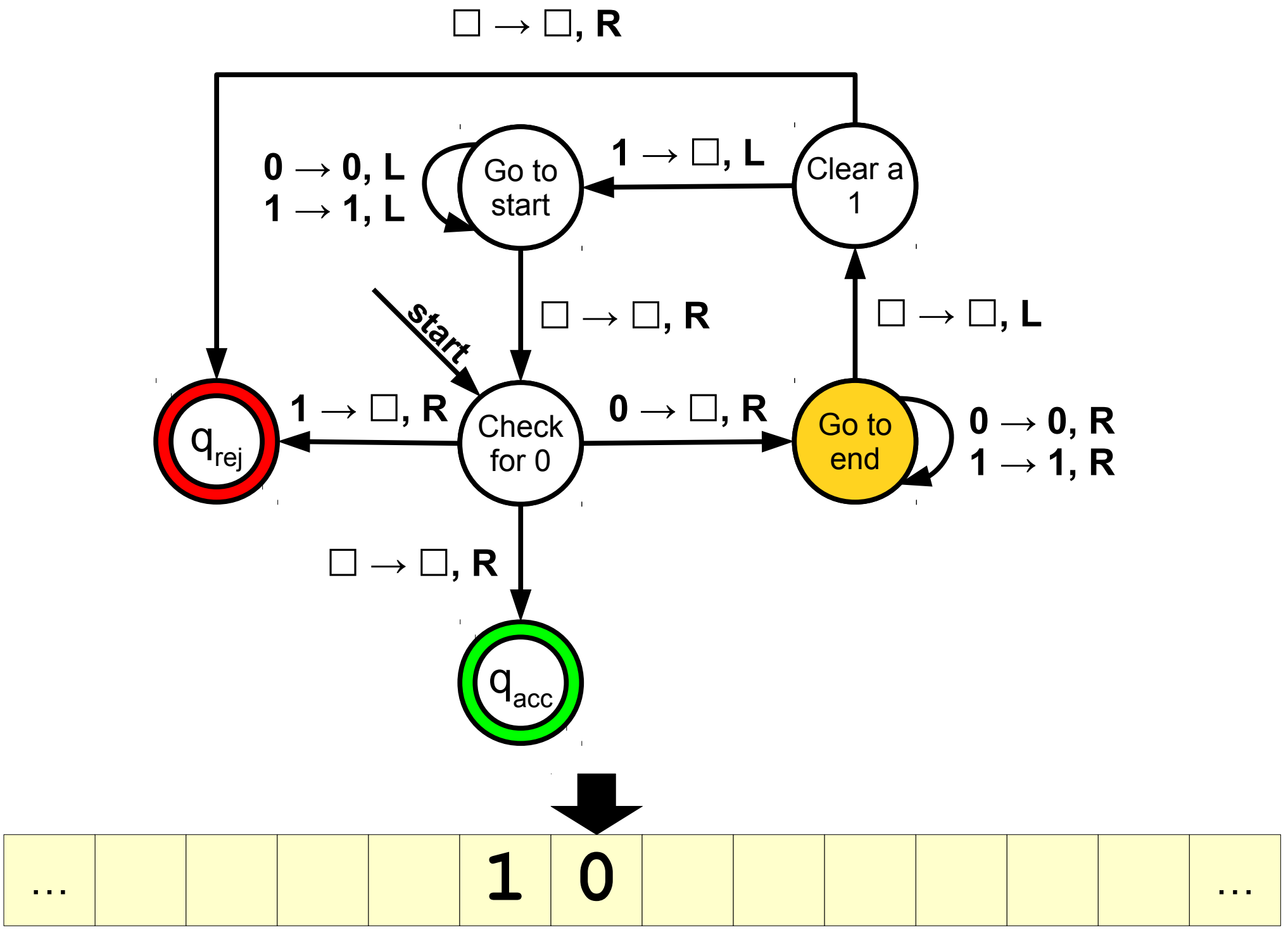

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
...

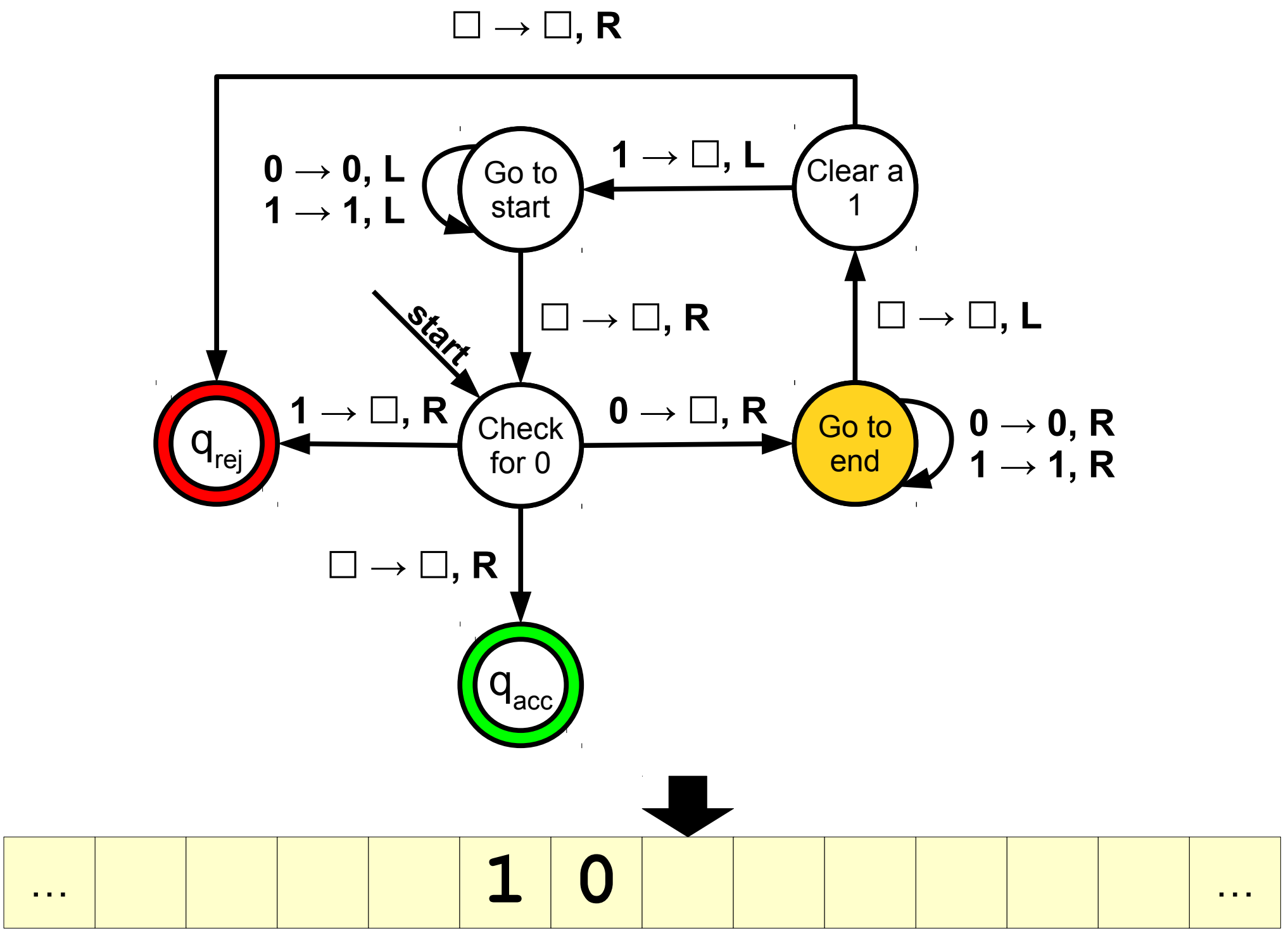

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc

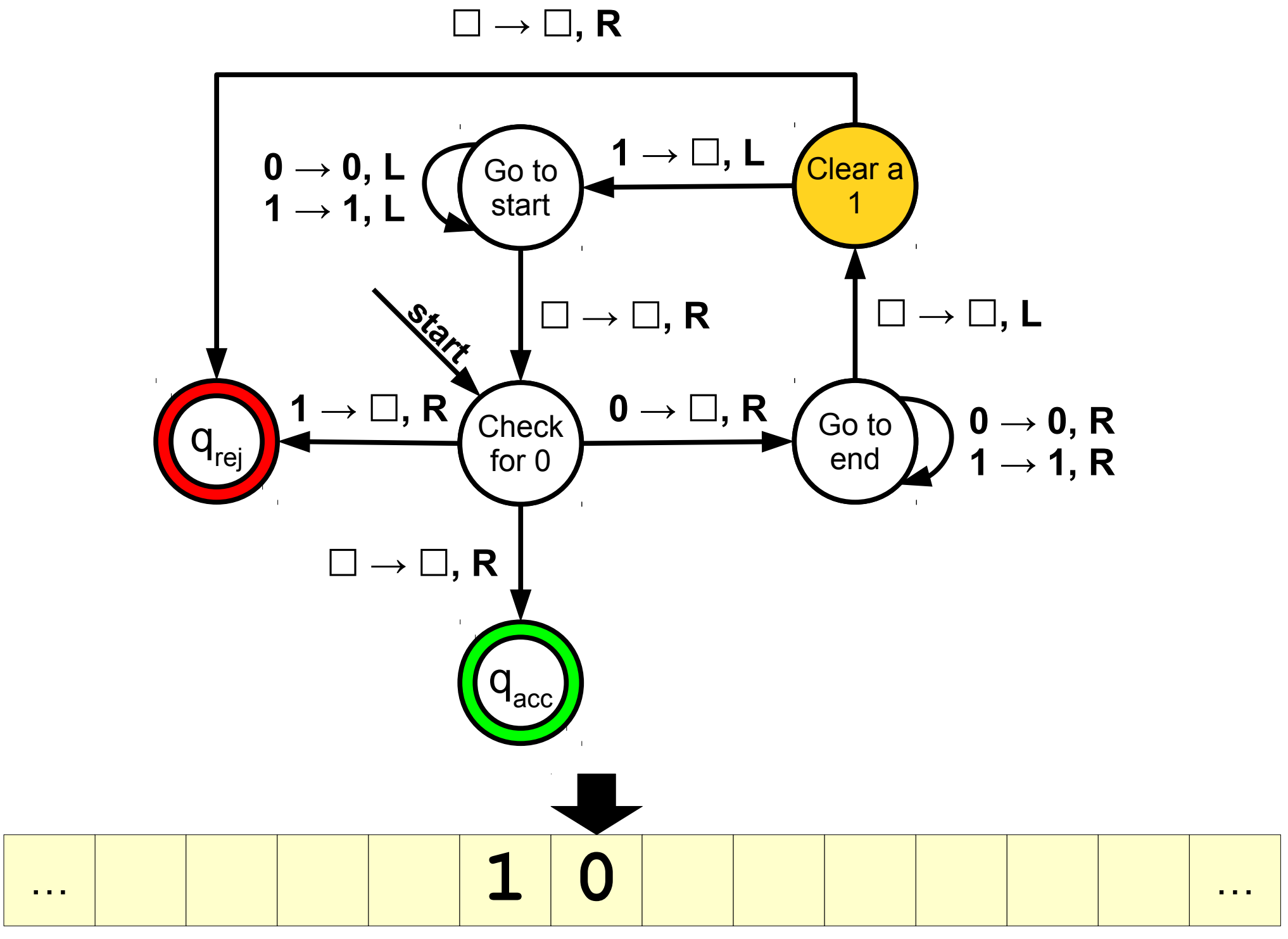

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0 1 0
...

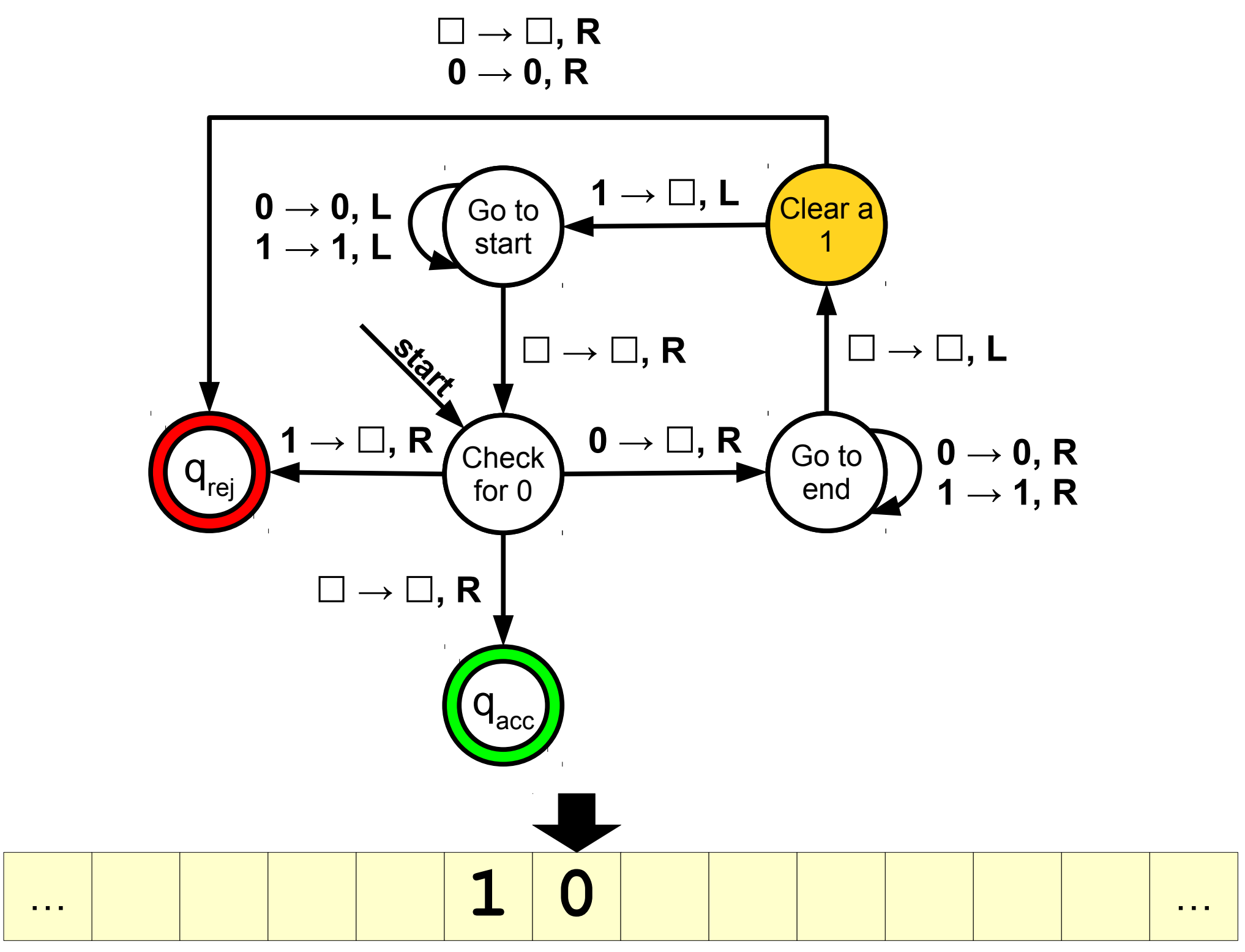

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
0 1 0
...

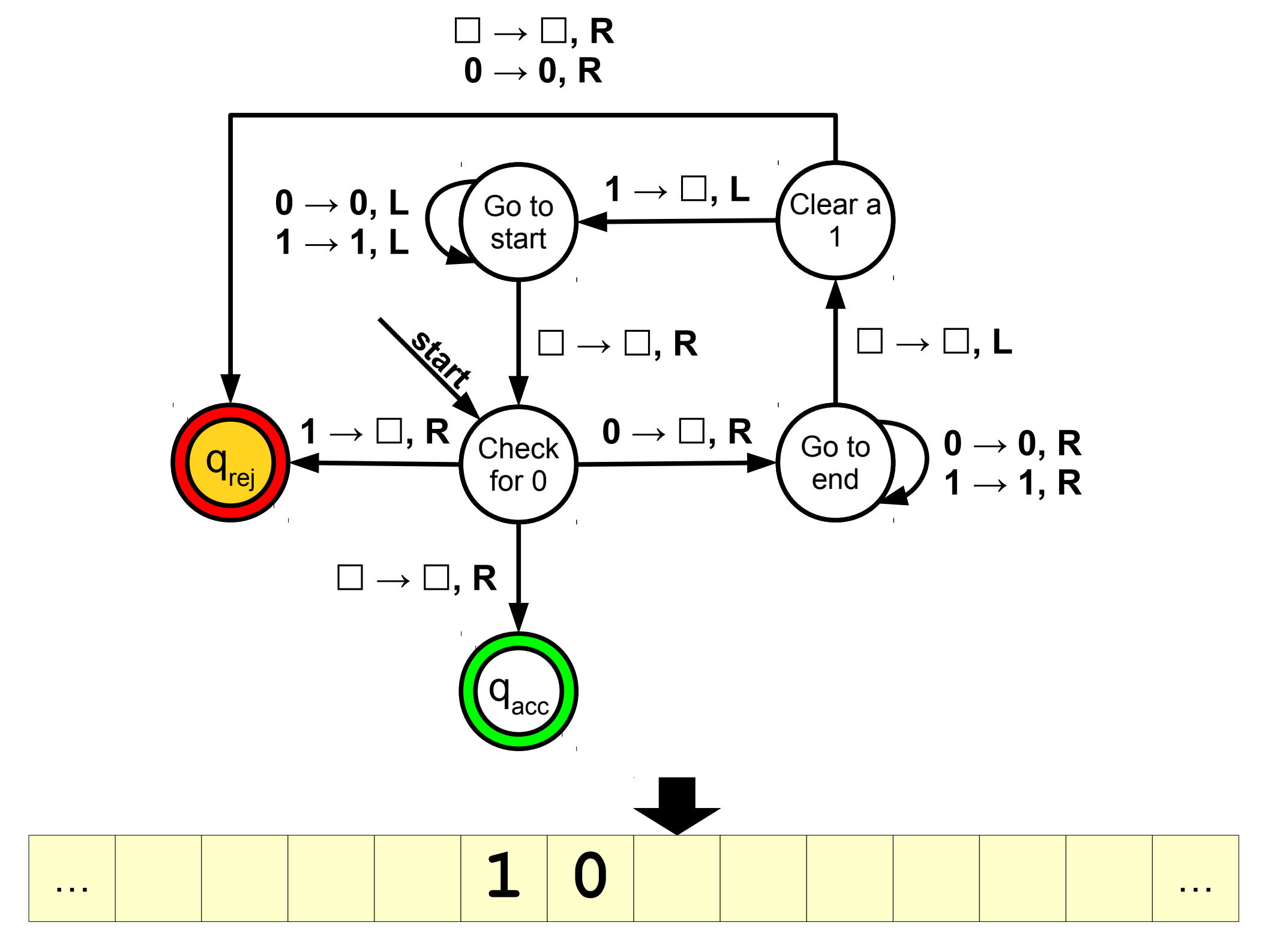

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
0
...

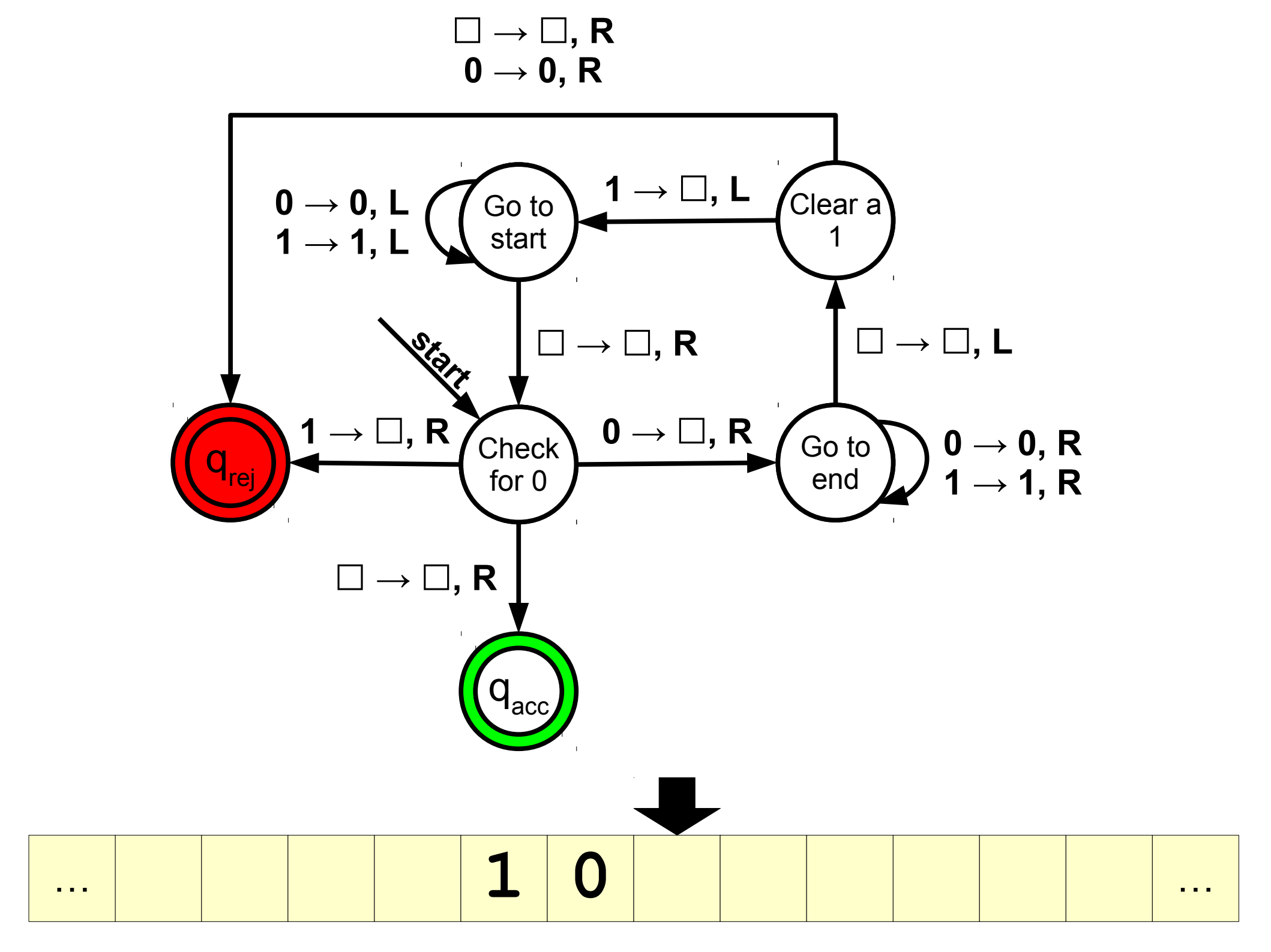

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
0
...

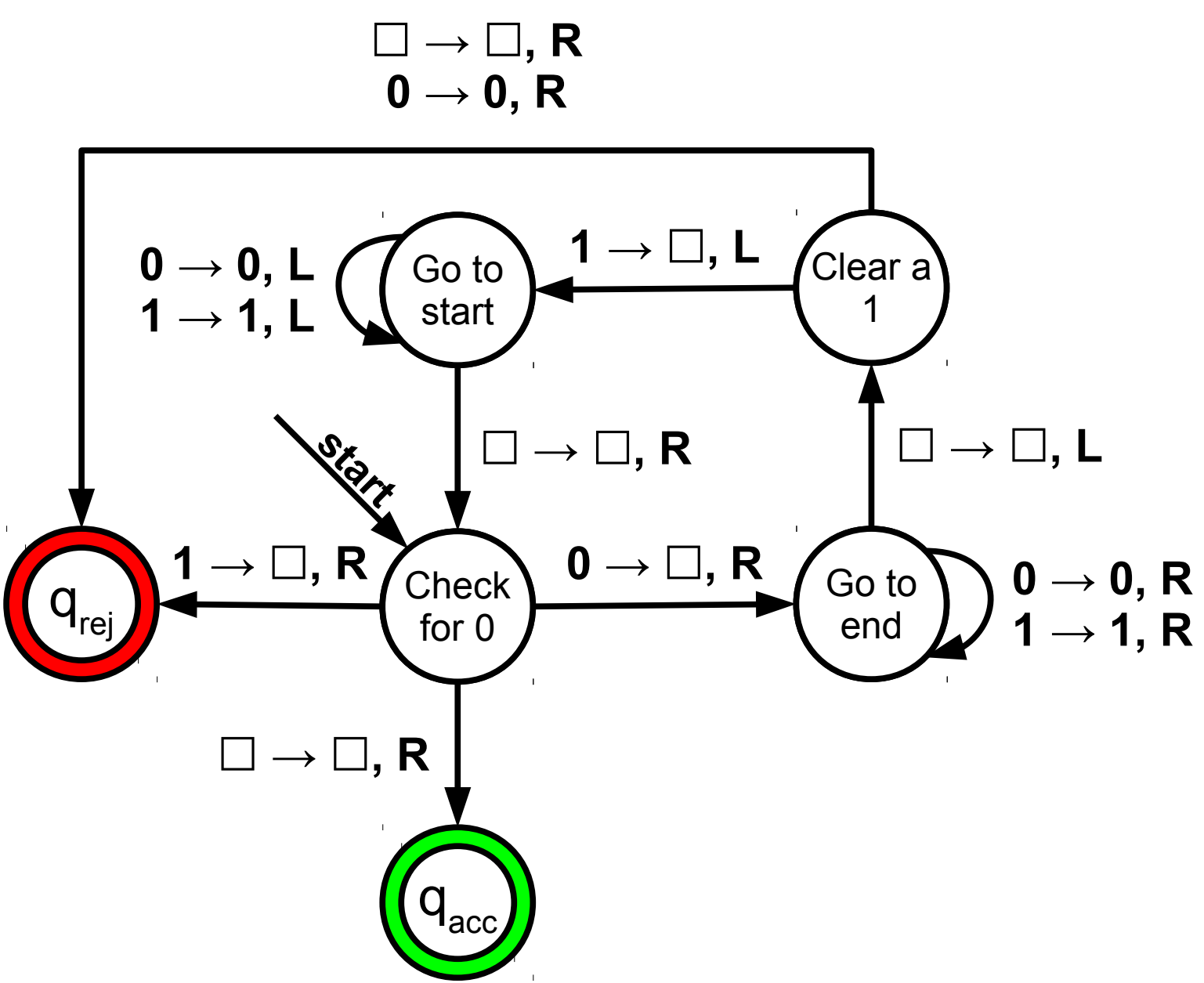

□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
0
...


□ → □, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
0
...


□ → □, R
0 → 0, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
0
...


□ → □, R
0 → 0, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc
...
1
0
...


□ → □, R
0 → 0, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
□ → □, R
q_{rej}
q_{acc}
…
1
0
…


□ → □, R
0 → 0, R
0 → 0, L
1 → 1, L
Go to start
1 → □, L
Clear a 1
start
□ → □, R
□ → □, L
1 → □, R
Check for 0
0 → □, R
Go to end
0 → 0, R
1 → 1, R
q_rej
□ → □, R
q_acc

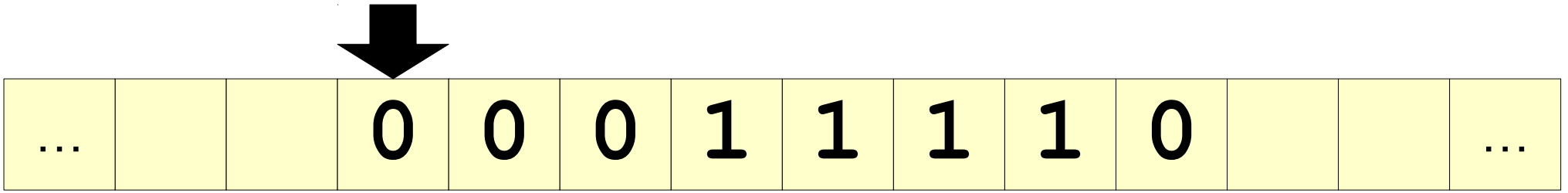
Another TM Design

- We've designed a TM for $\{ \mathtt{0}^n\mathtt{1}^n \mid n \in \mathbb{N} \}$.
- Consider this language over $\Sigma = \{ \mathtt{0}, \mathtt{1} \}$:

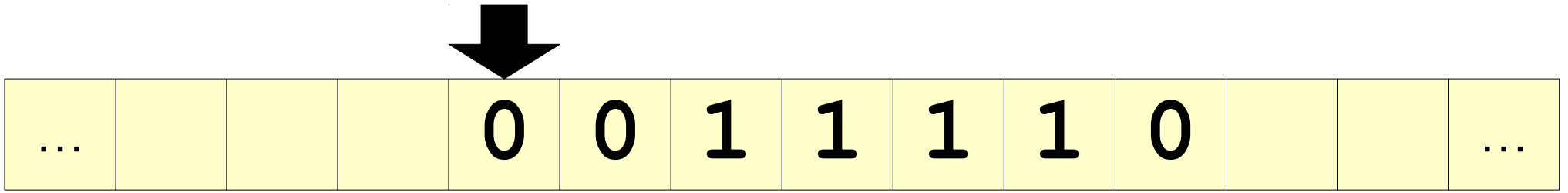
$$L = \{ w \in \Sigma^* \mid w \text{ has the same number of } \mathtt{0}\text{s and } \mathtt{1}\text{s} \}$$

- This language is also not regular, but it is context-free.
- How might we design a TM for it?

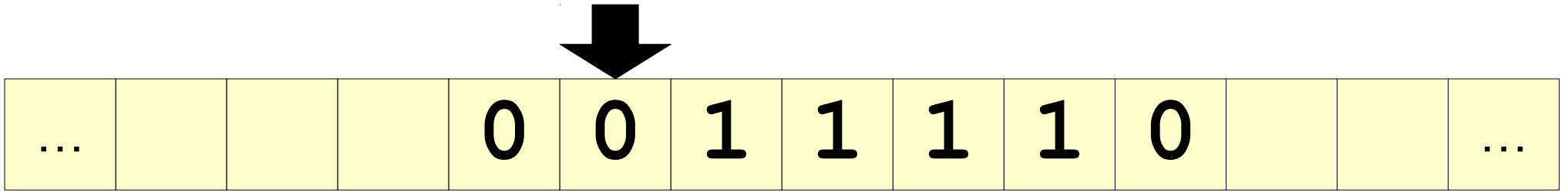
A Caveat



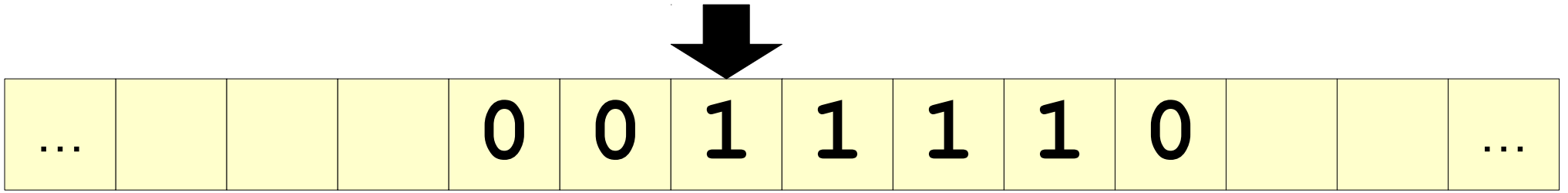
A Caveat



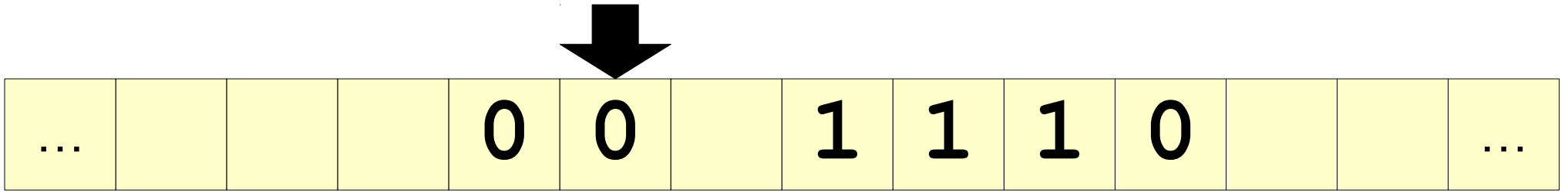
A Caveat



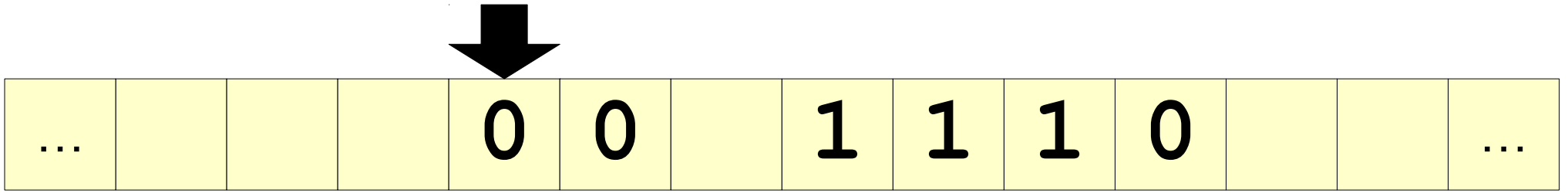
A Caveat



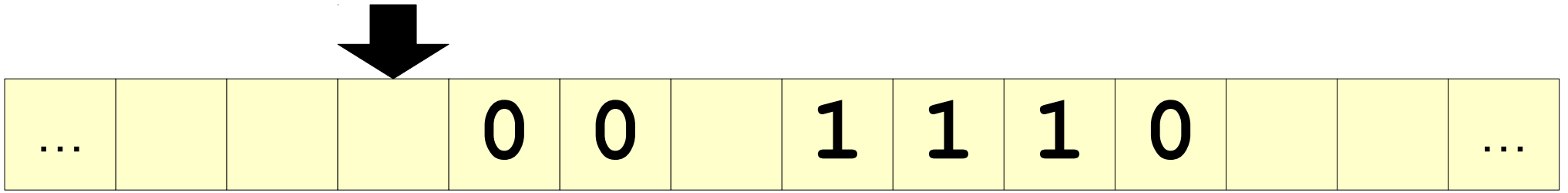
A Caveat



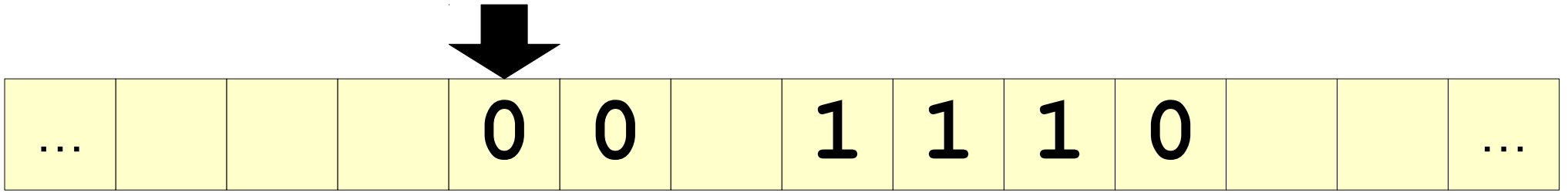
A Caveat



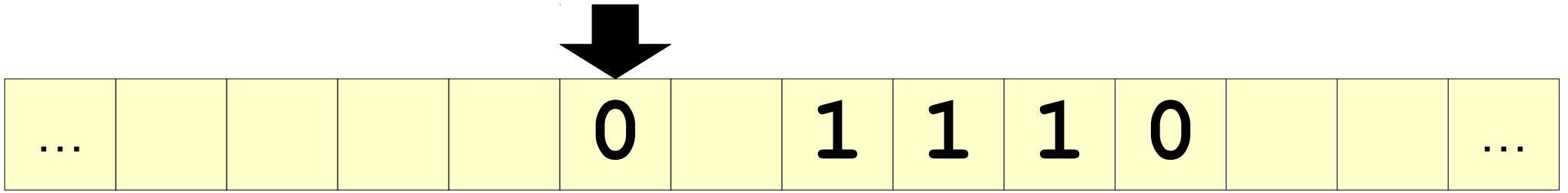
A Caveat



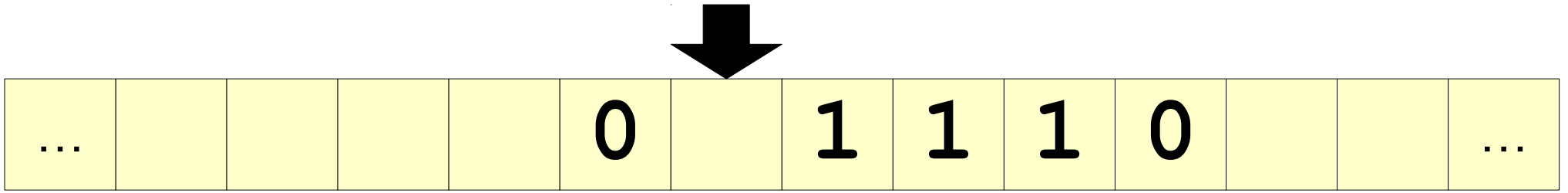
A Caveat



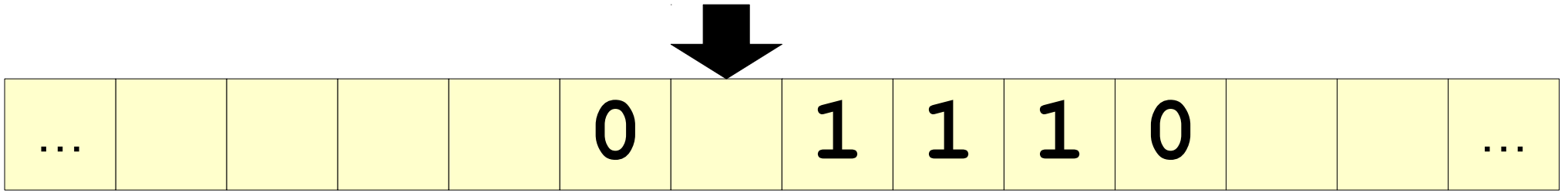
A Caveat



A Caveat

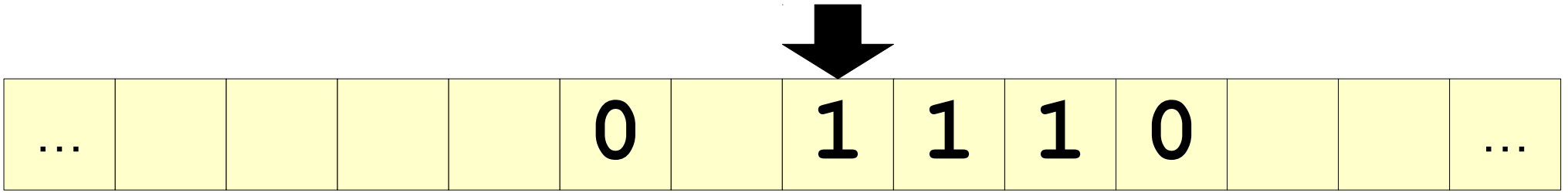


A Caveat

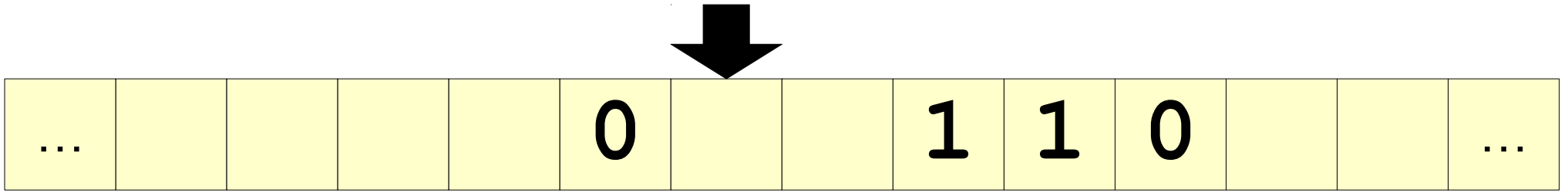


How do we know that this blank isn't one of the infinitely many blanks after our input string?

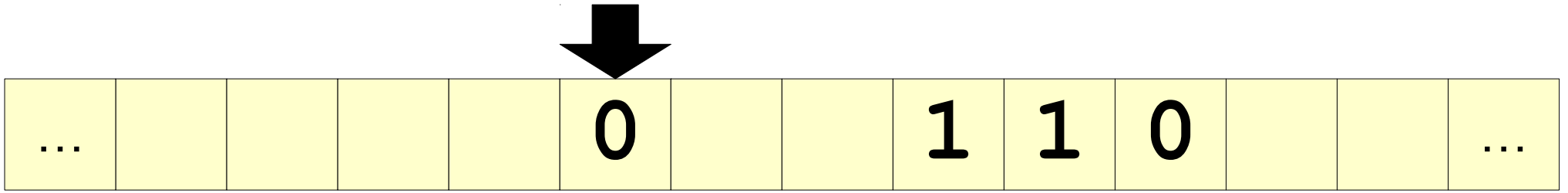
A Caveat



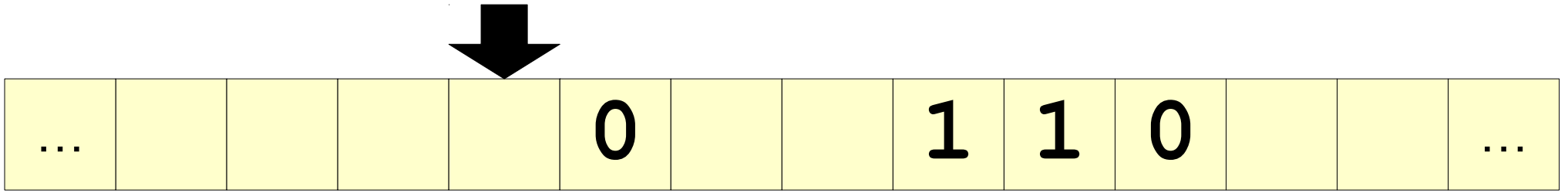
A Caveat



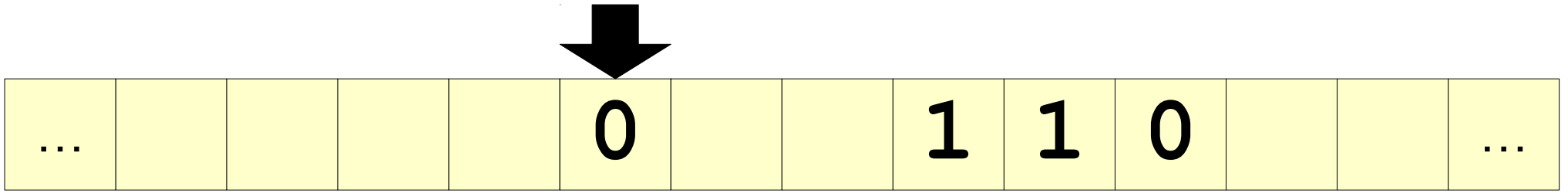
A Caveat



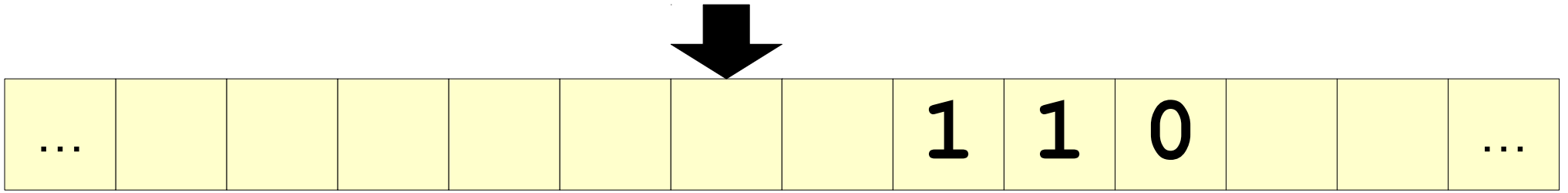
A Caveat



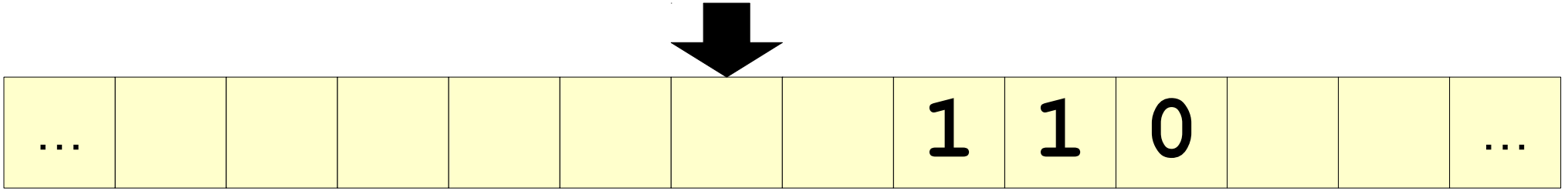
A Caveat



A Caveat

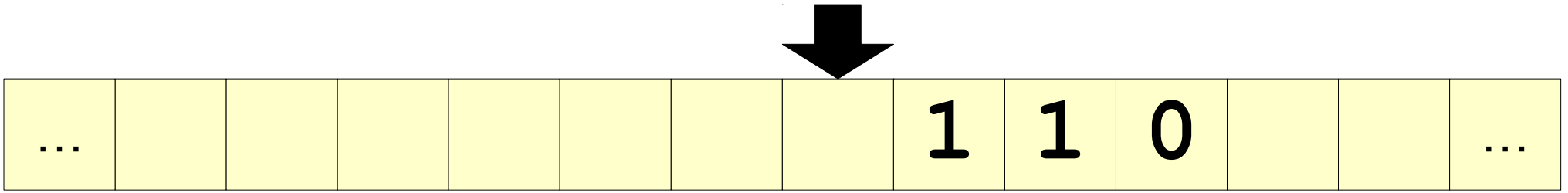


A Caveat

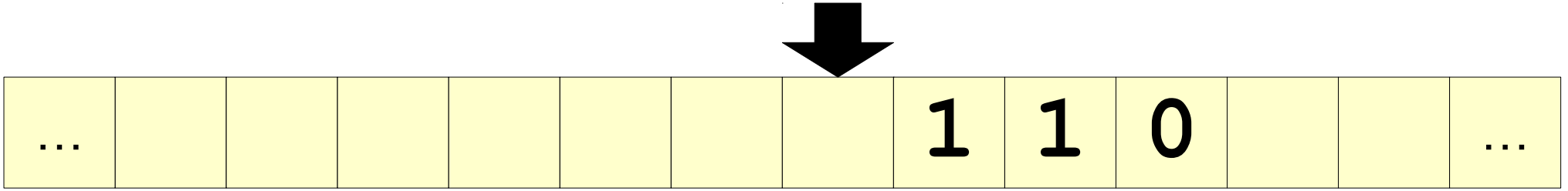


How do we know that this blank isn't one of the infinitely many blanks after our input string?

A Caveat

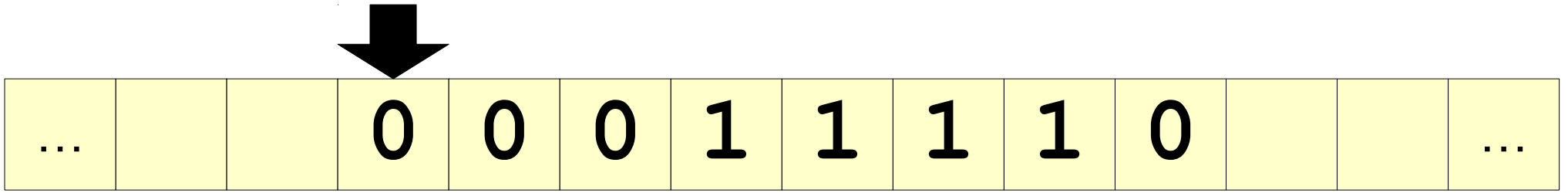


A Caveat

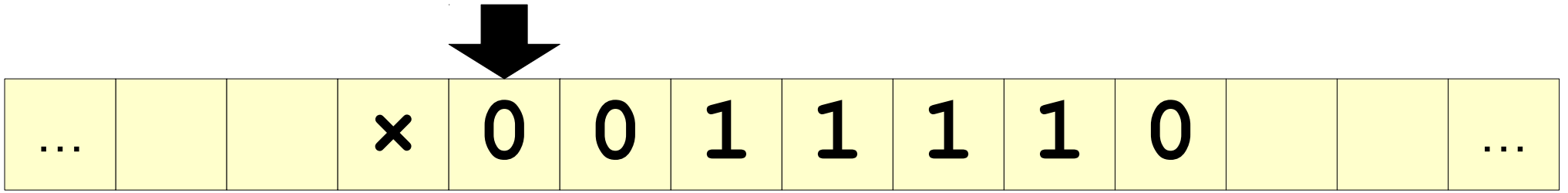


How do we know that this blank isn't one of the infinitely many blanks after our input string?

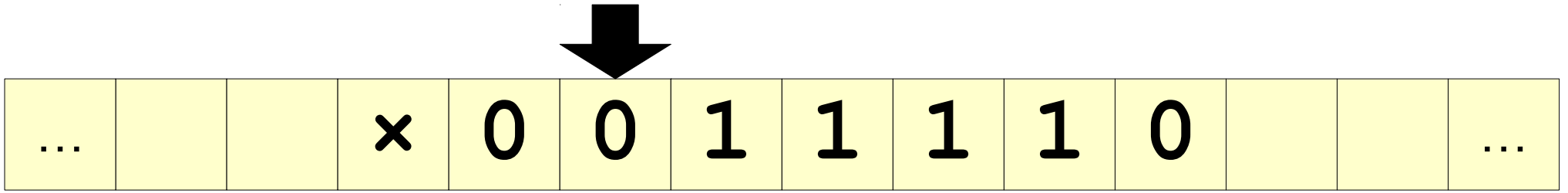
The Solution



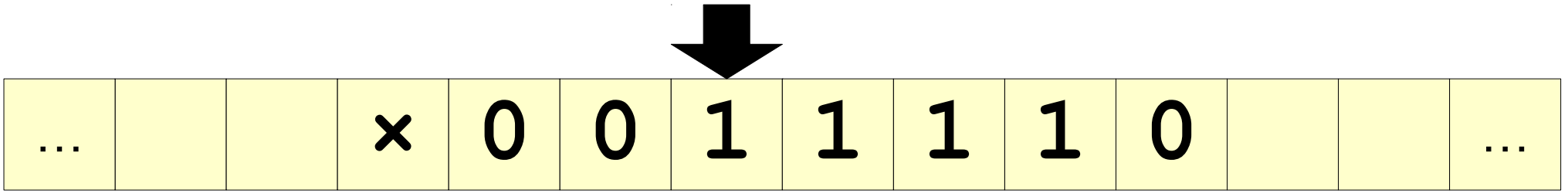
The Solution



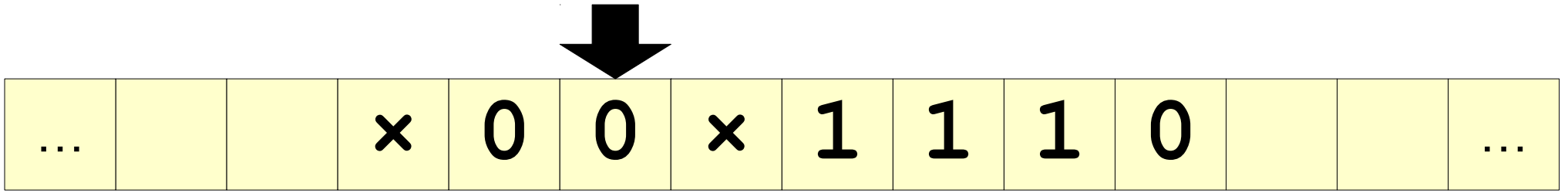
The Solution



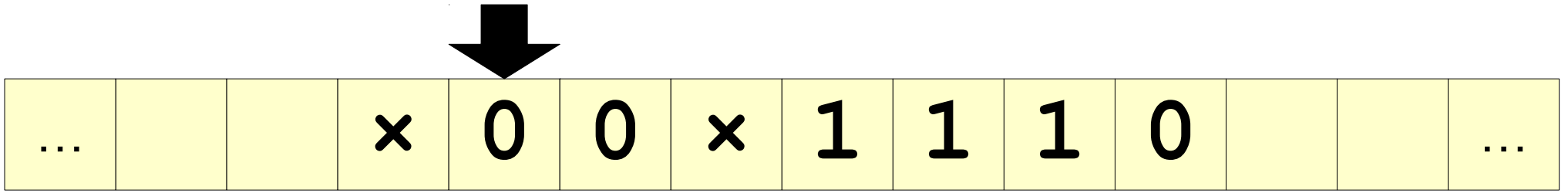
The Solution



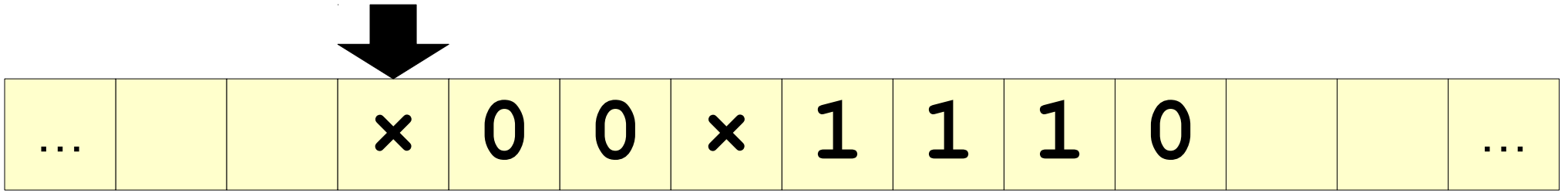
The Solution



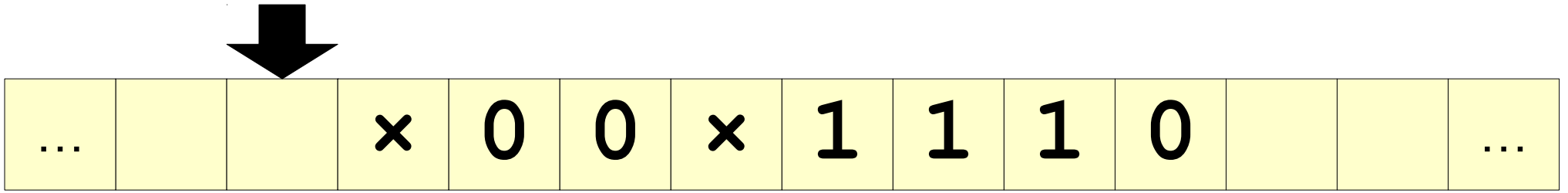
The Solution



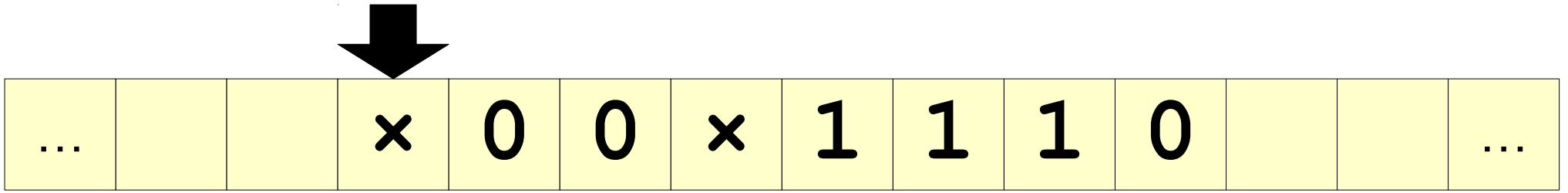
The Solution



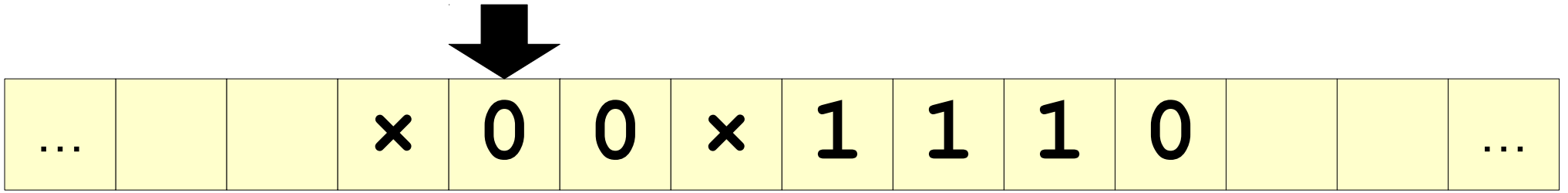
The Solution



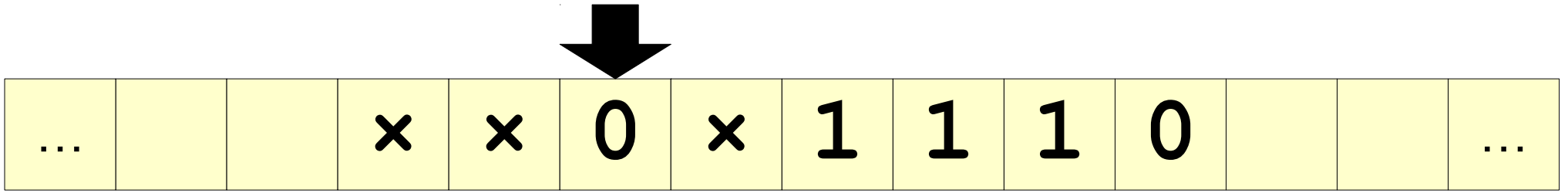
The Solution



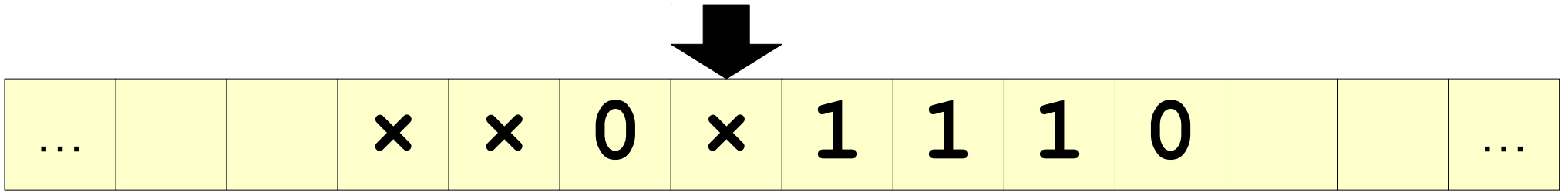
The Solution



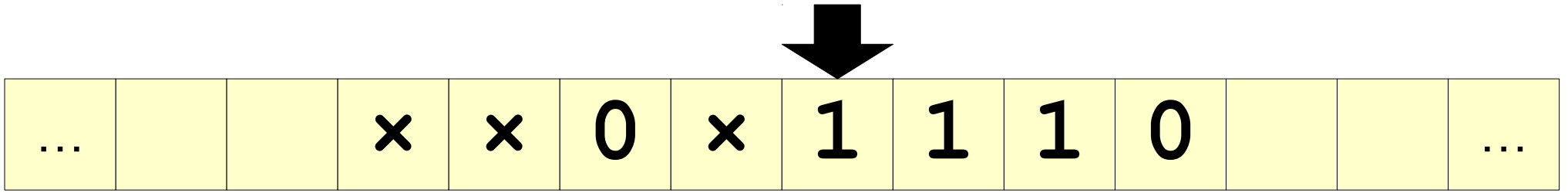
The Solution



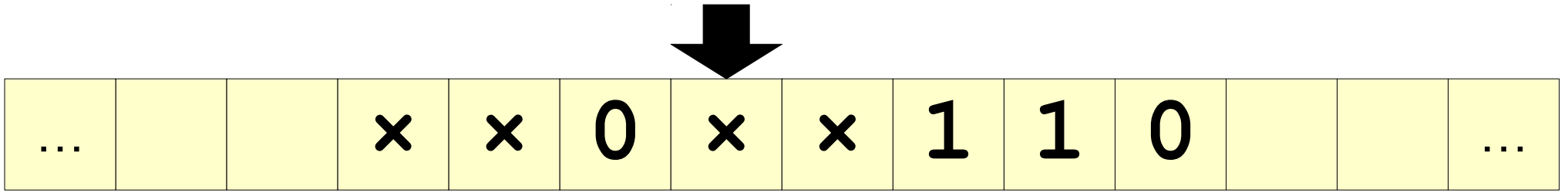
The Solution



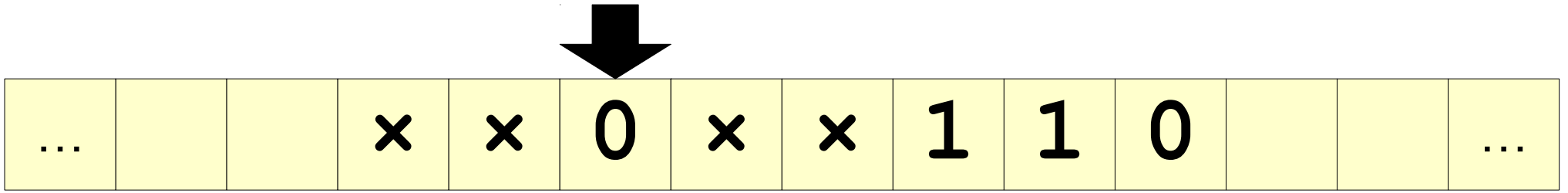
The Solution



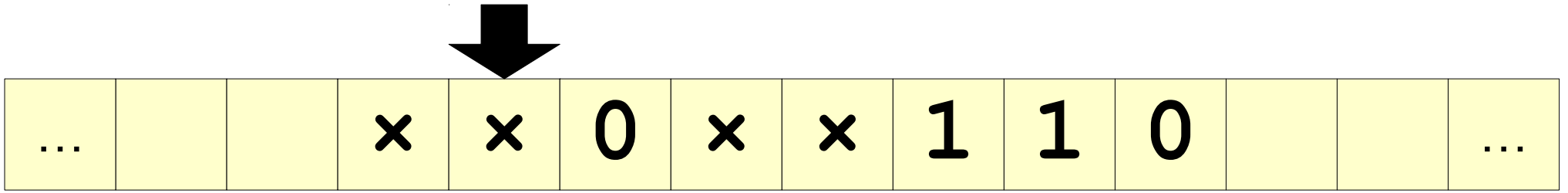
The Solution



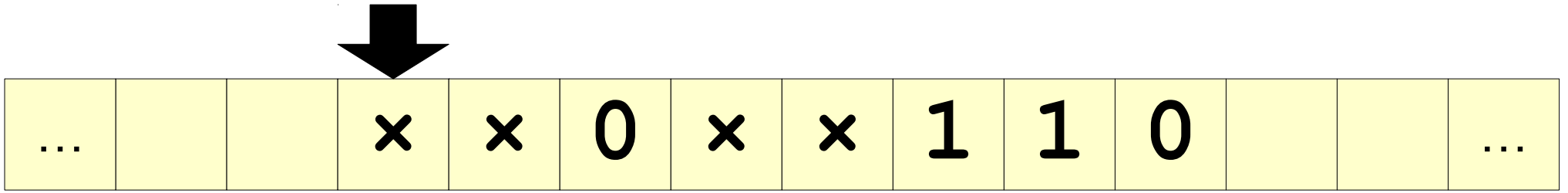
The Solution



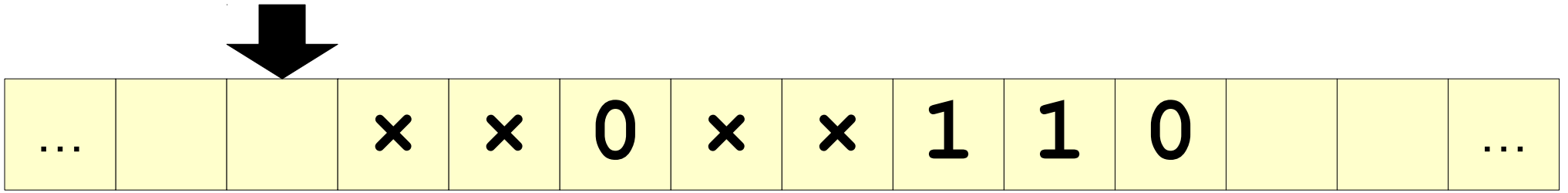
The Solution



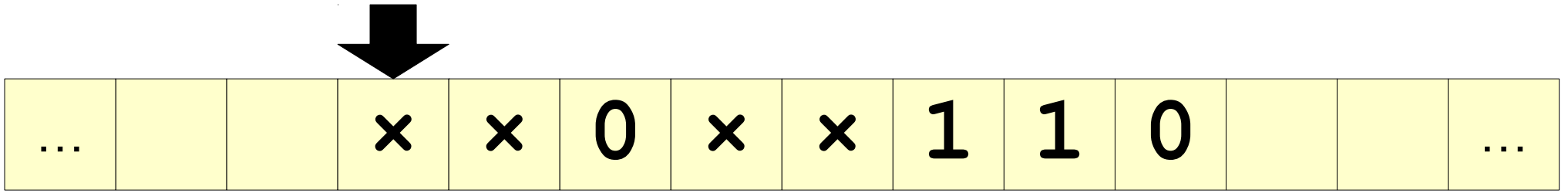
The Solution



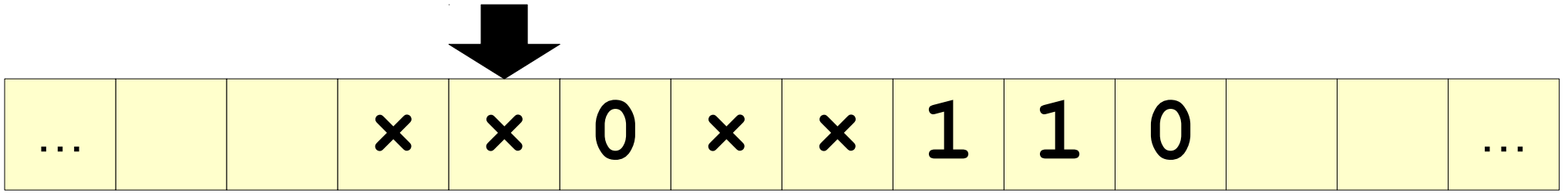
The Solution



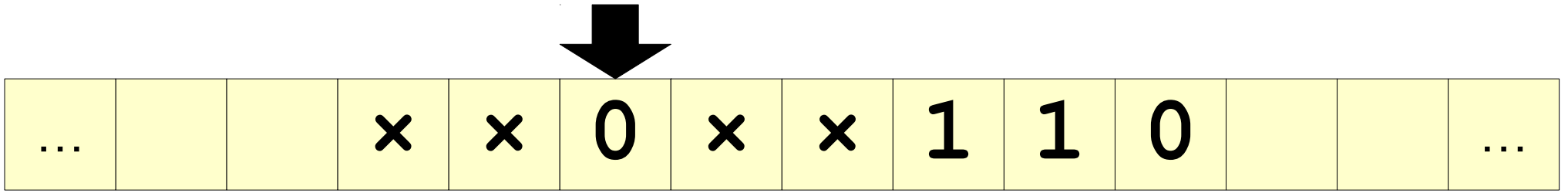
The Solution



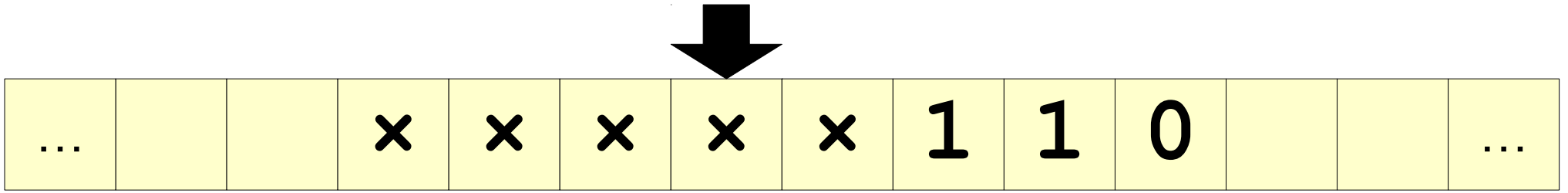
The Solution



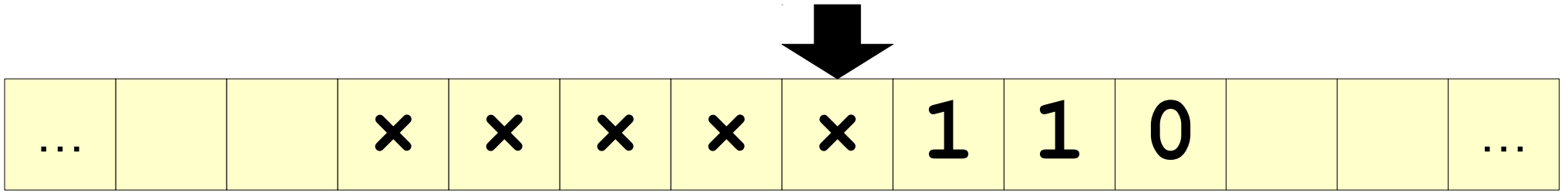
The Solution



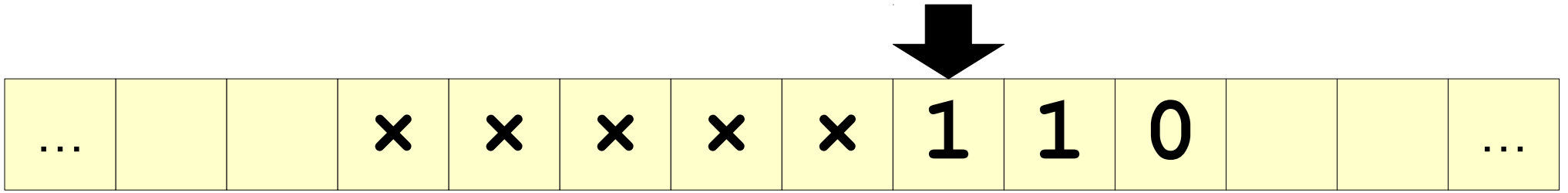
The Solution



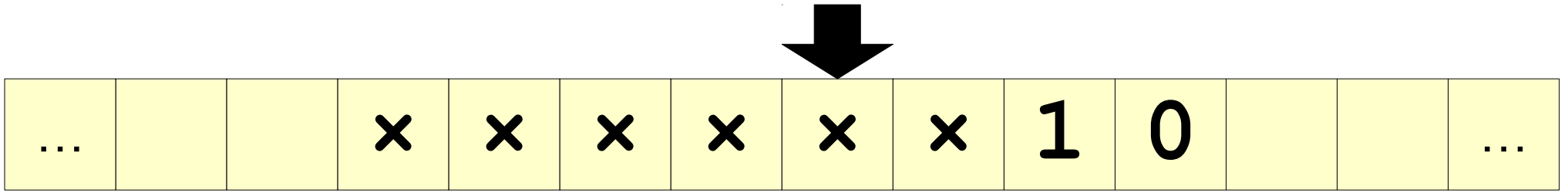
The Solution



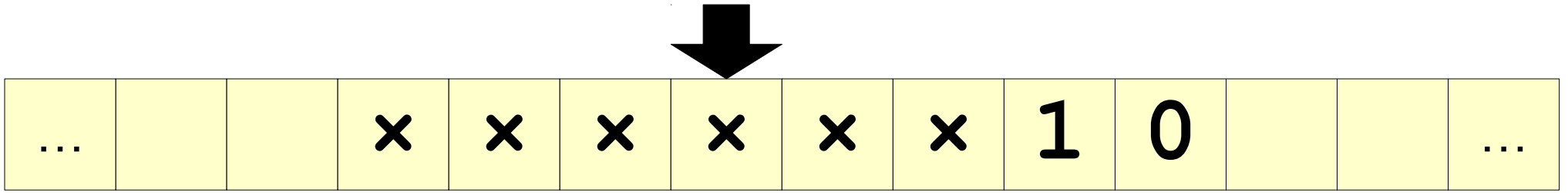
The Solution

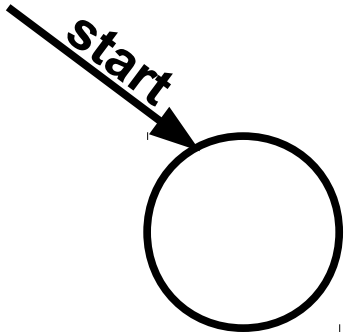


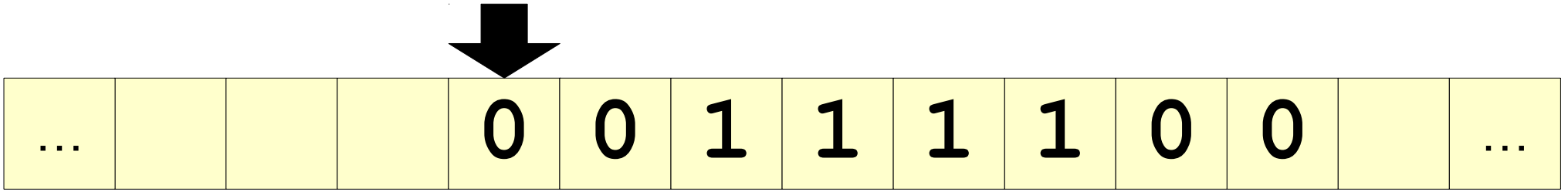
The Solution

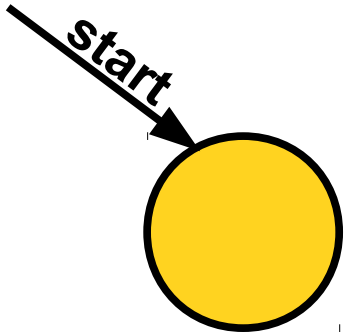


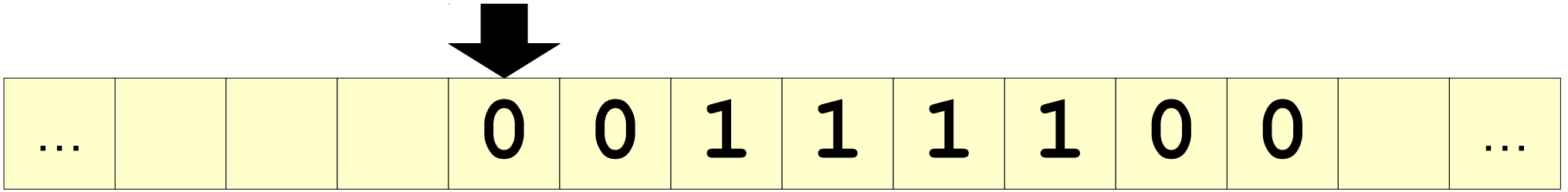
The Solution

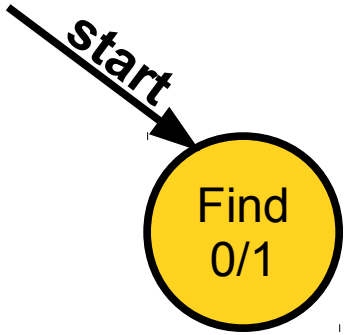


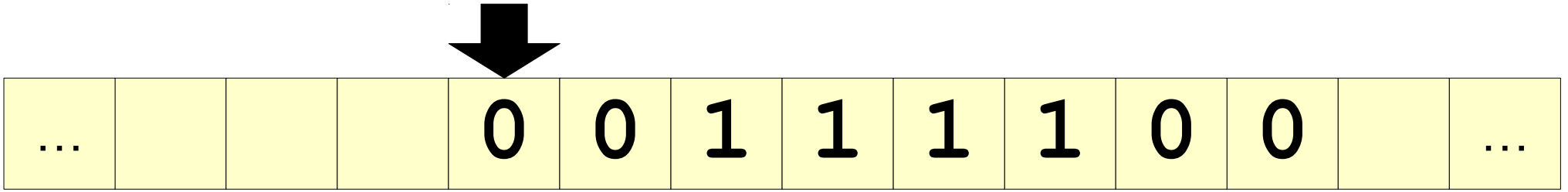

start

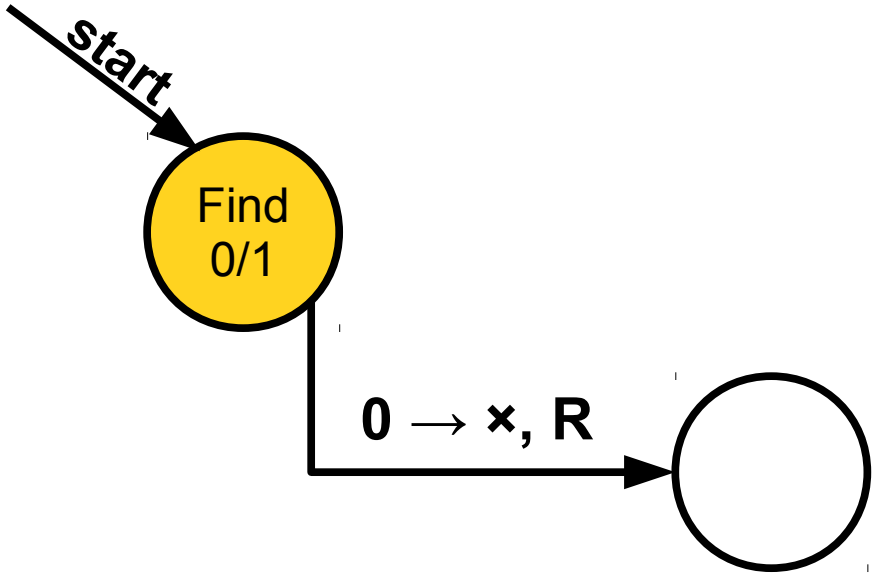

...
0
0
1
1
1
1
0
0
...

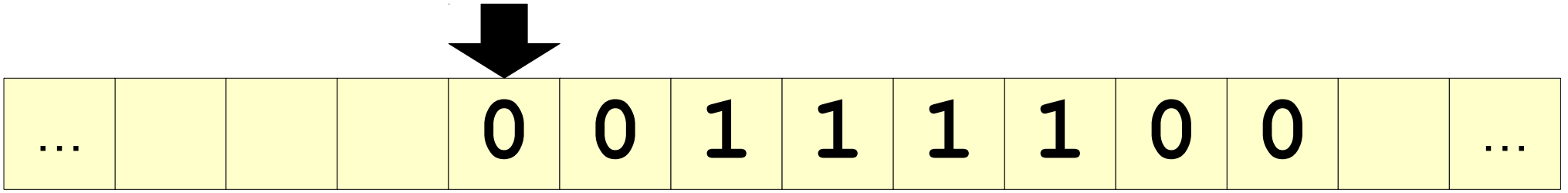

start

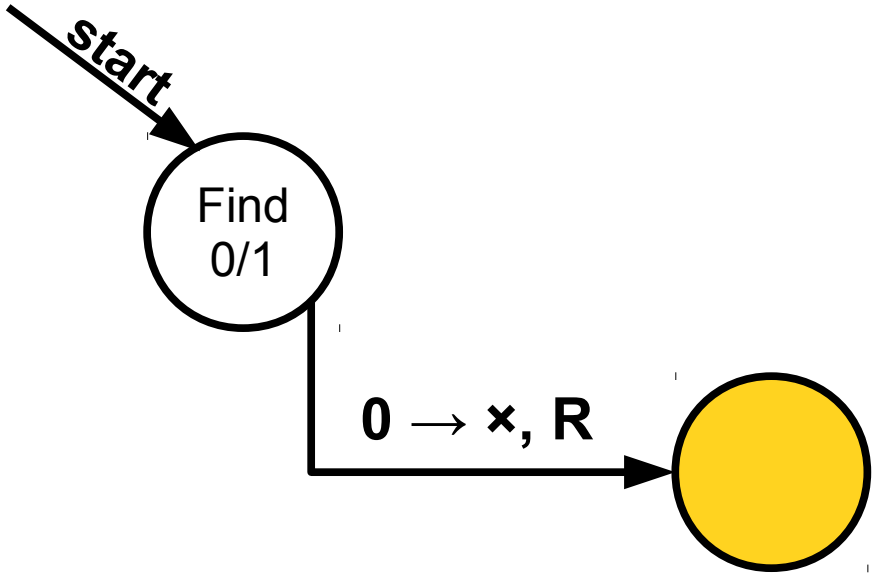

...
0 0 1 1 1 1 0 0
...

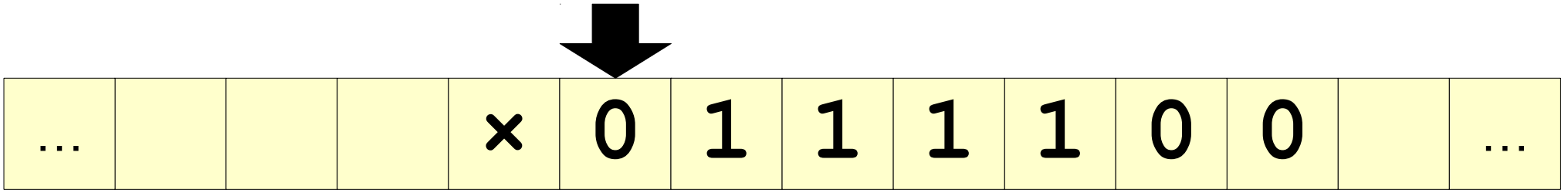

start
Find
0/1

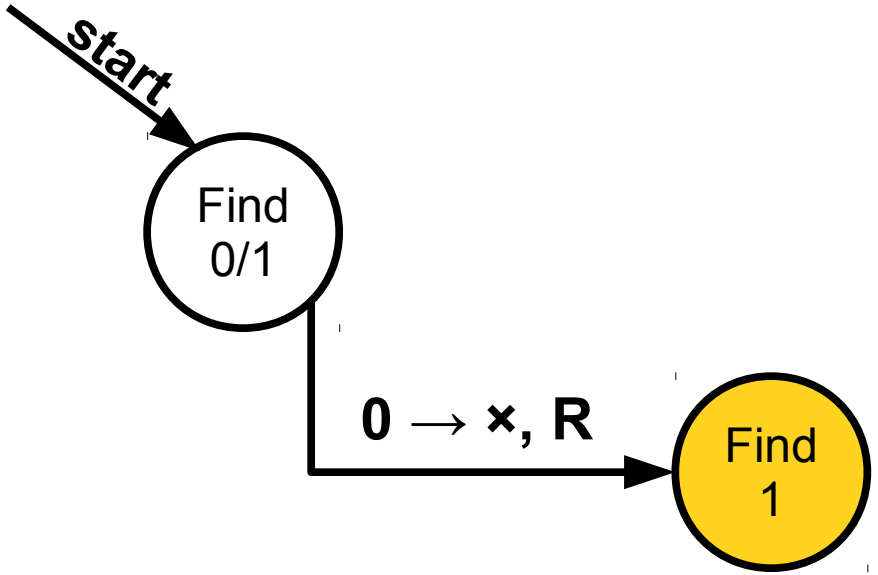

...
0
0
1
1
1
1
0
0
...

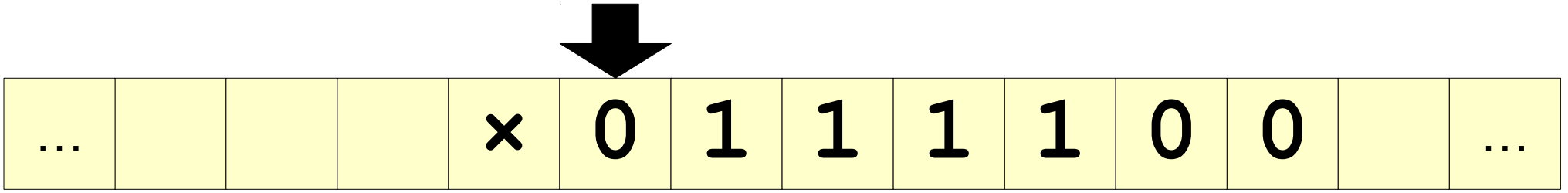

start
Find
0/1
0 → ×, R

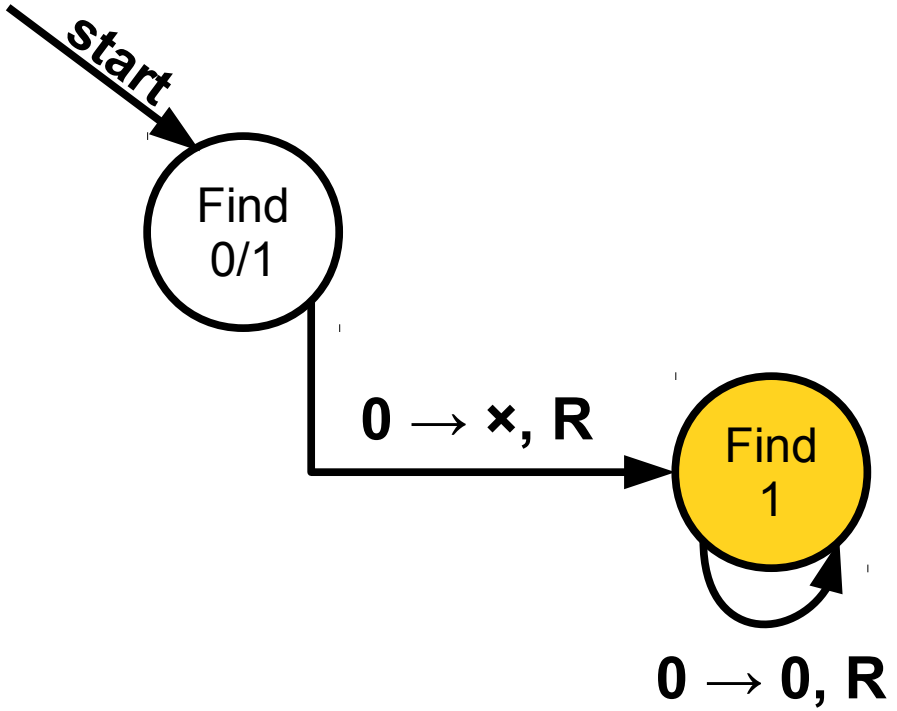

…
0 0 1 1 1 1 0 0
…

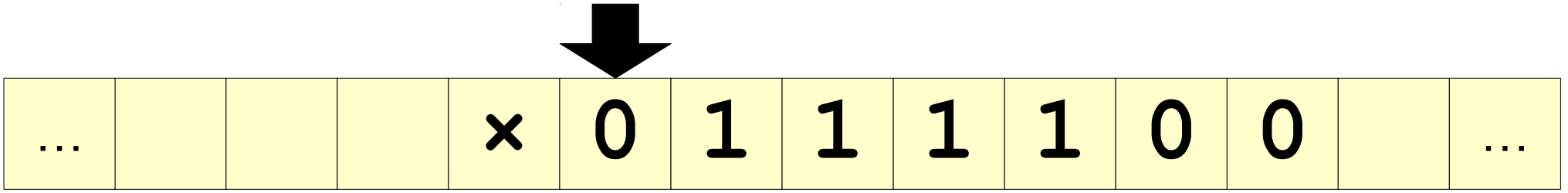

start
Find
0/1
0 → ×, R

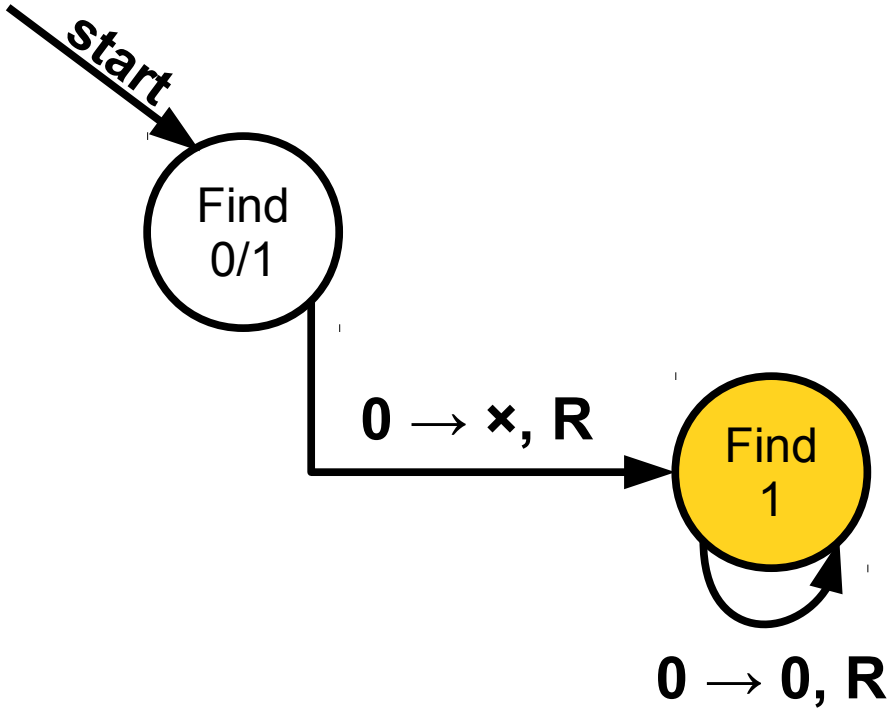

…
×
0
1
1
1
1
0
0
…

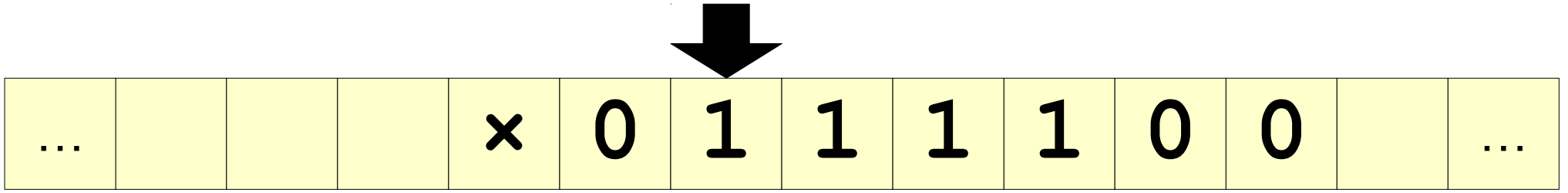

start
Find
0/1
0 → ×, R
Find
1

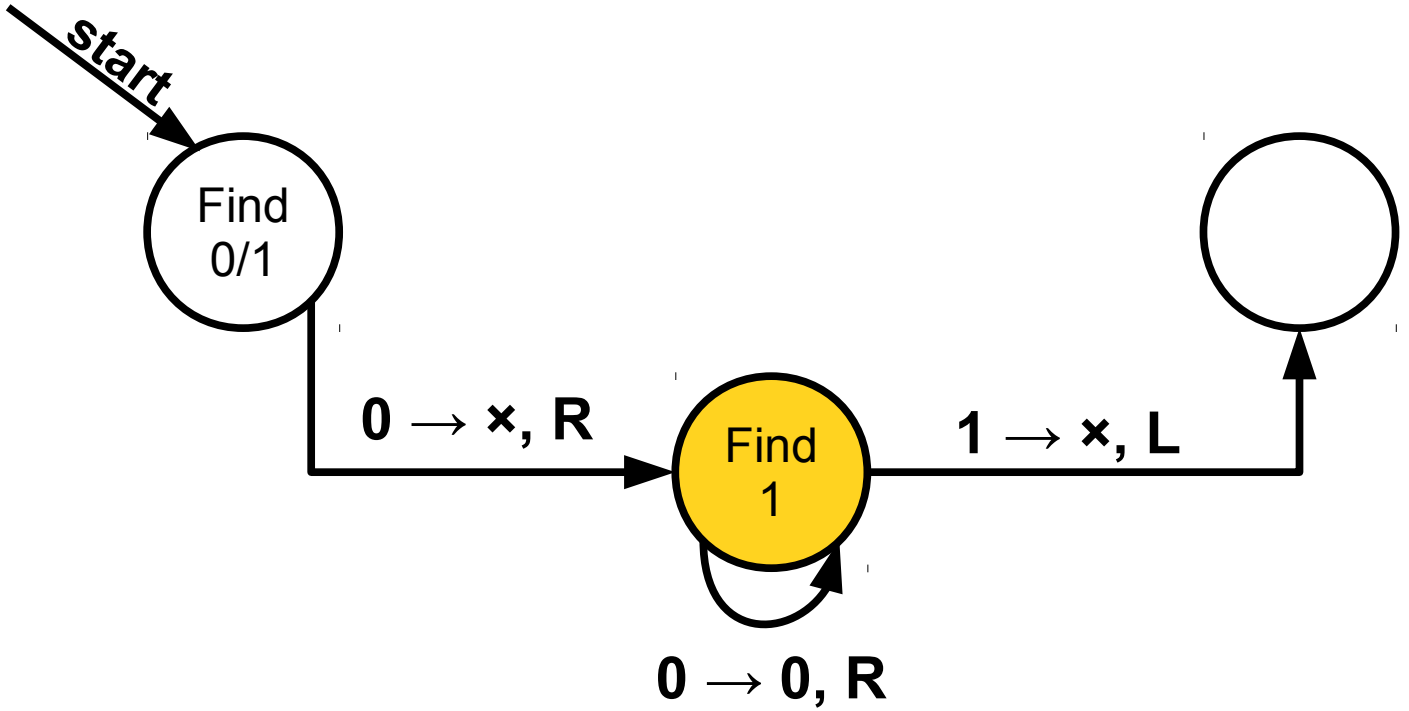

…
×
0
1
1
1
1
0
0
…

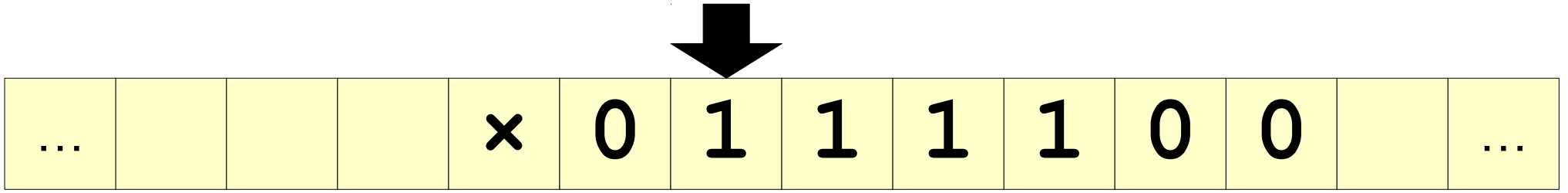

start
Find
0/1
0 → ×, R
Find
1
0 → 0, R

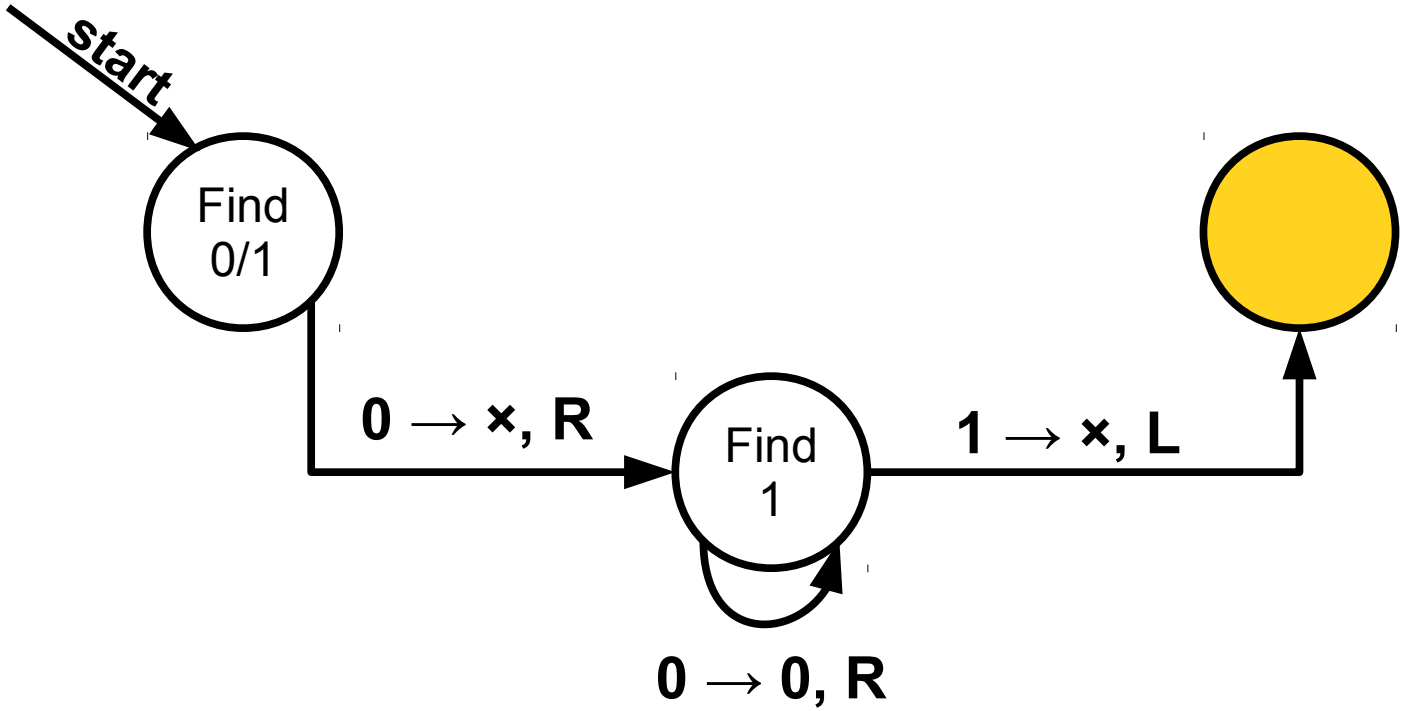

…
×
0
1
1
1
1
0
0
…

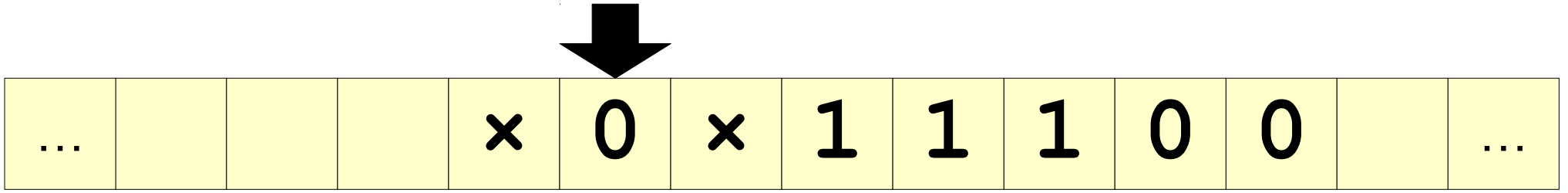

start
Find
0/1
0 → ×, R
Find
1
0 → 0, R

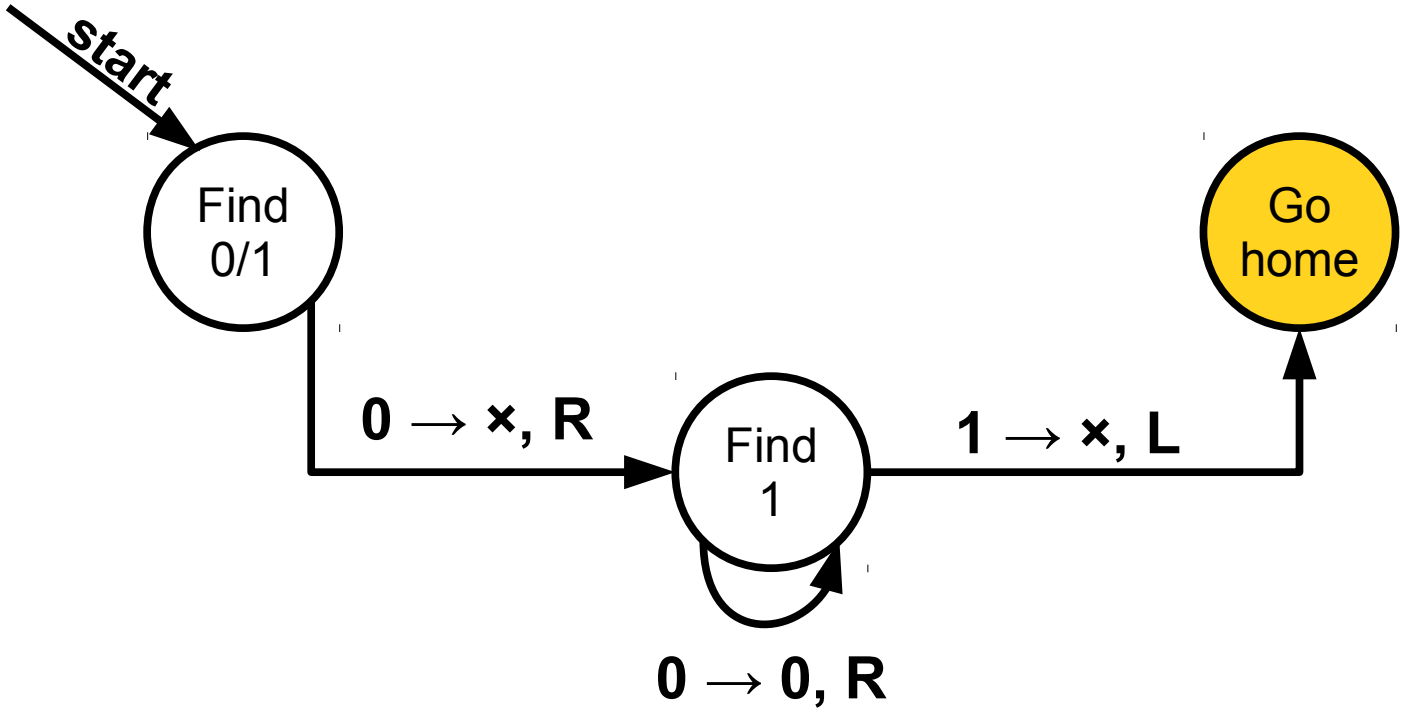

…
×
0
1
1
1
1
0
0
…

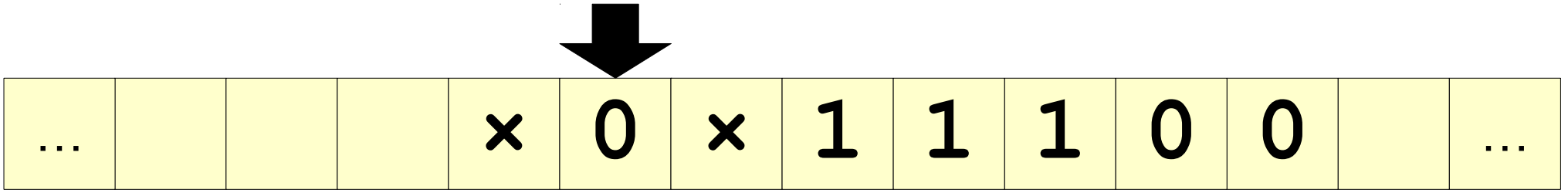

start
Find
0/1
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

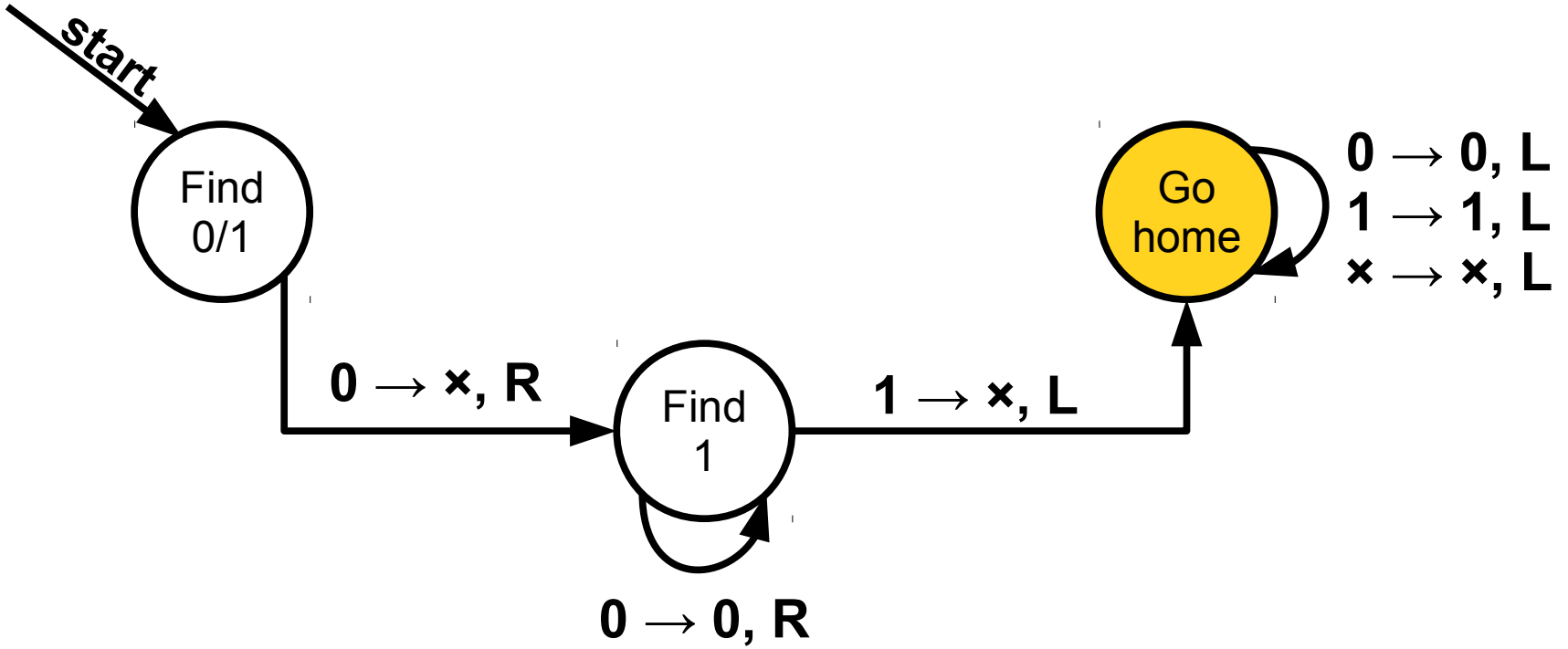

…
×
0
1
1
1
1
0
0
…

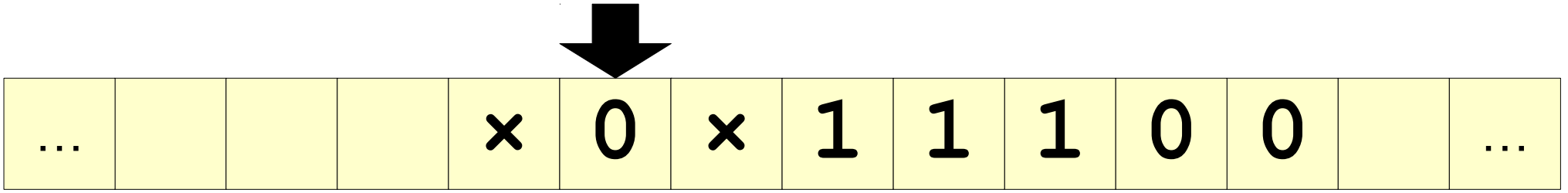

start
Find
0/1
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

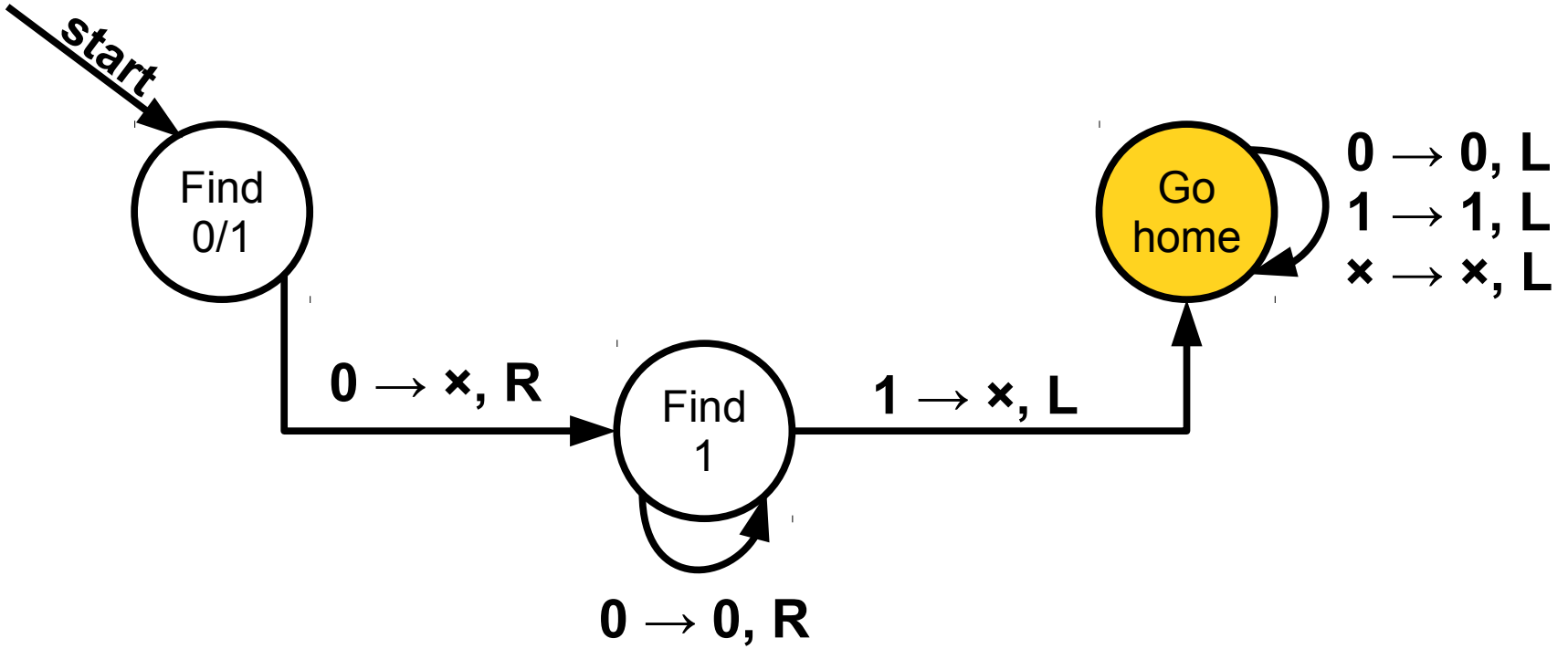

…
×
0
×
1
1
1
0
0
…

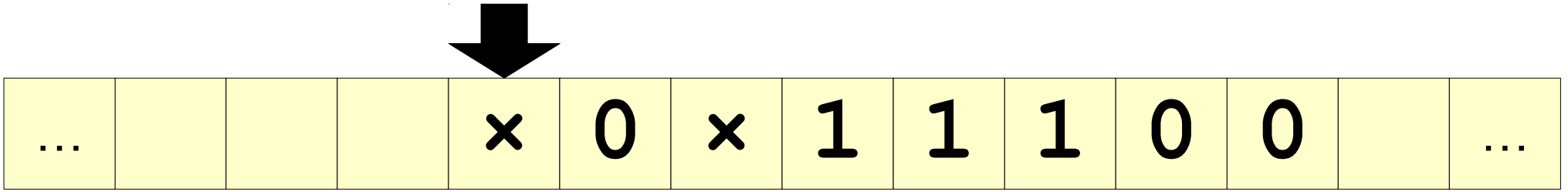

start
Find
0/1
0 → ×, R
Find
1
1 → ×, L
Go
home
0 → 0, R

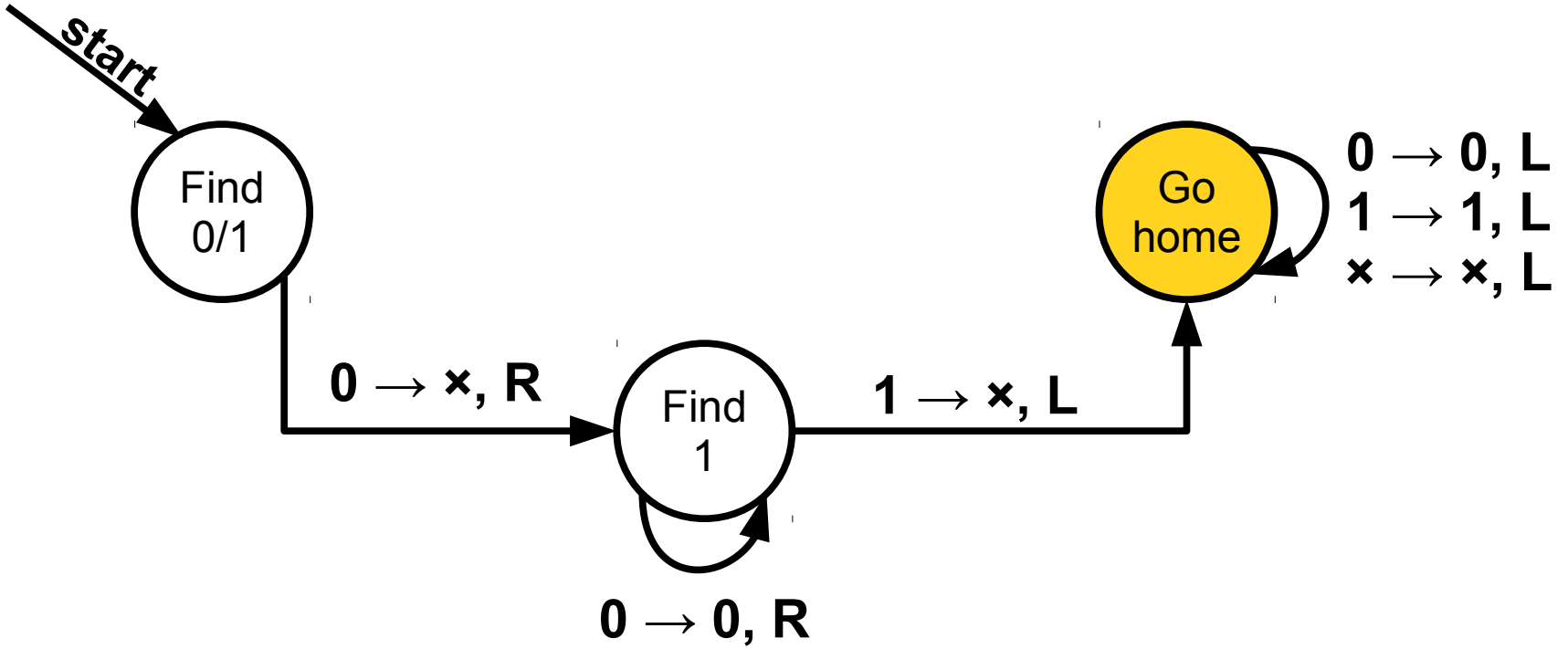

…
×
0
×
1
1
1
0
0
…

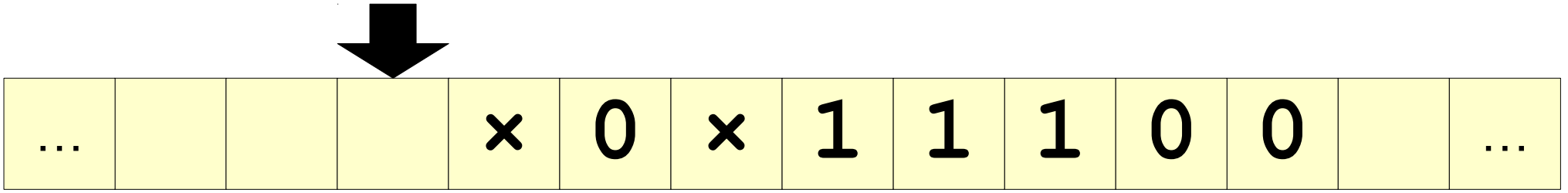

start
Find
0/1
0 → ×, R
Find
1
0 → 0, R
1 → ×, L
Go
home
0 → 0, L
1 → 1, L
× → ×, L

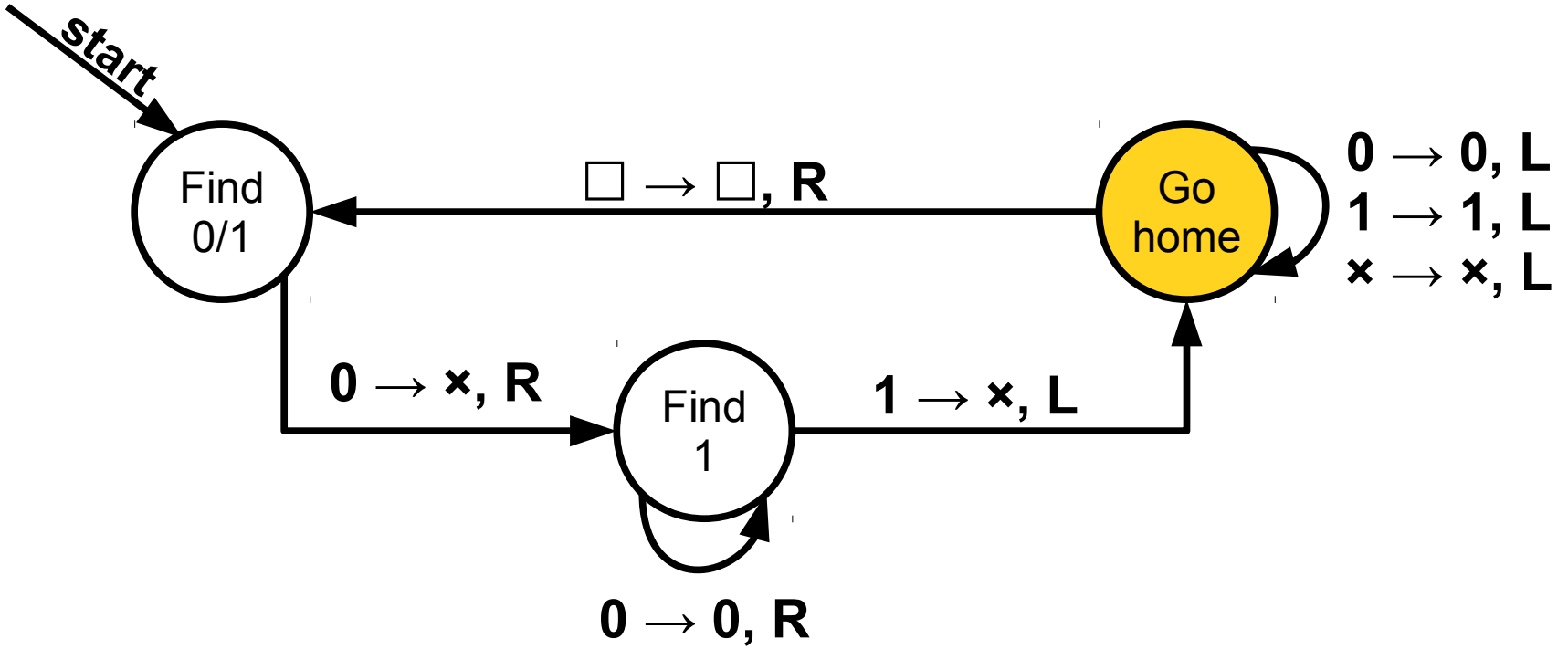

…
×
0
×
1
1
1
0
0
…

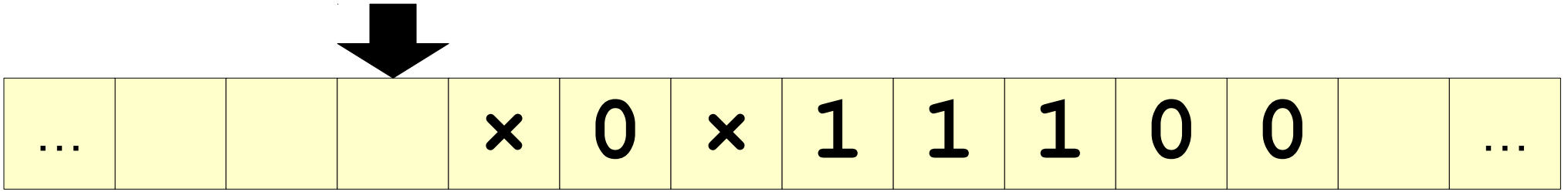

start
Find
0/1
0 → ×, R
Find
1
0 → 0, R
1 → ×, L
Go
home
0 → 0, L
1 → 1, L
× → ×, L

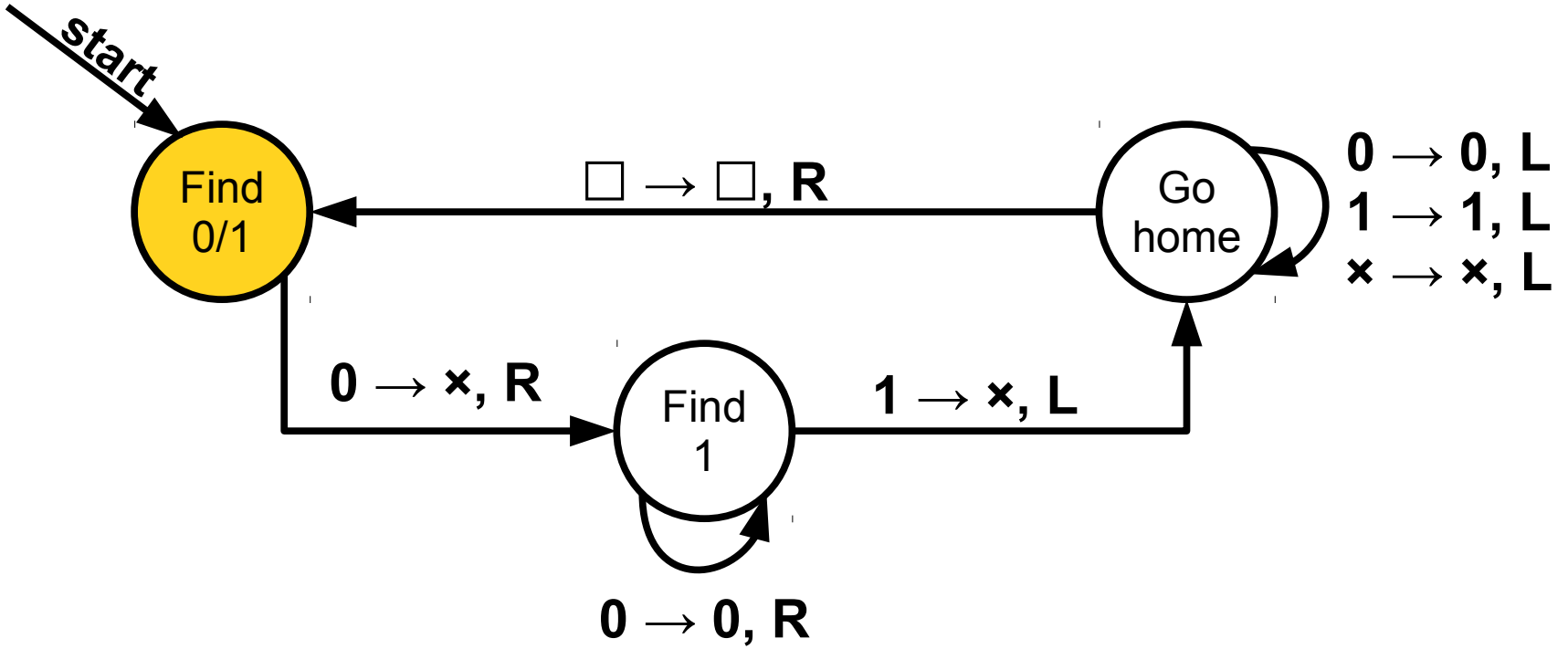

…
×
0
×
1
1
1
0
0
…

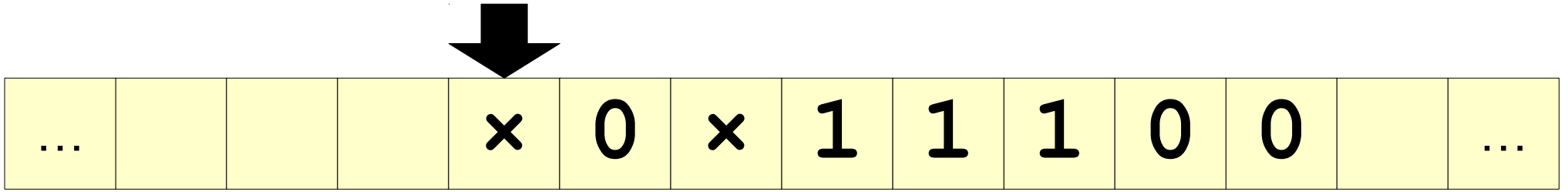

start
Find
0/1
0 → ×, R
Find
1
0 → 0, R
1 → ×, L
Go
home
0 → 0, L
1 → 1, L
× → ×, L

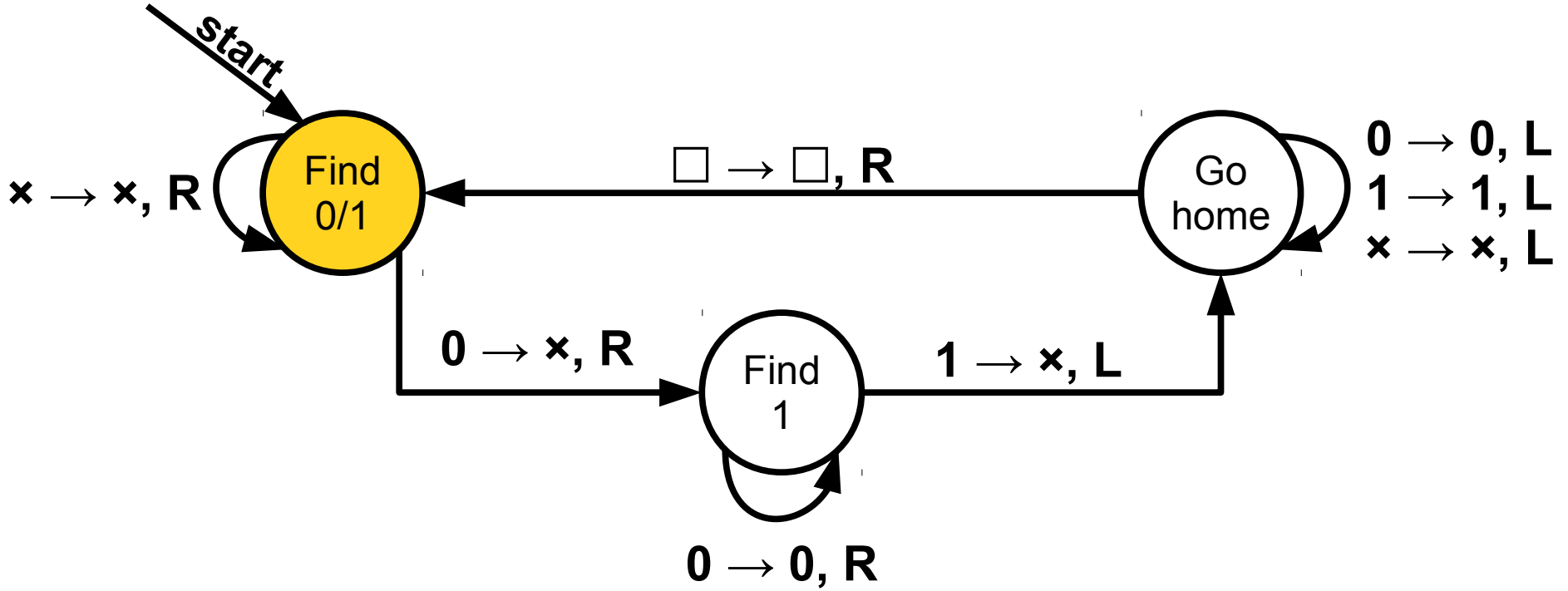

…
×
0
×
1
1
1
0
0
…

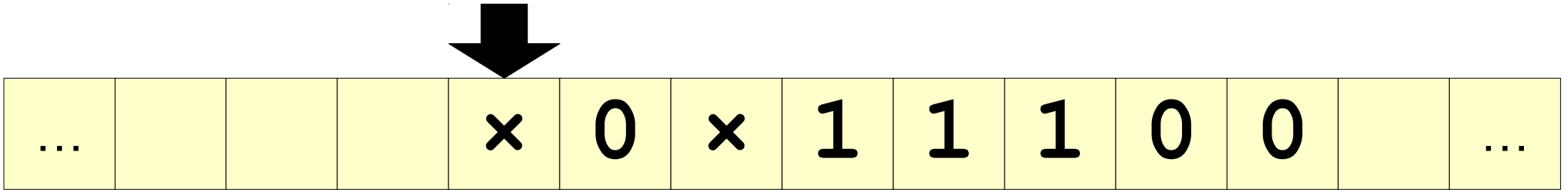

start
Find
0/1
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

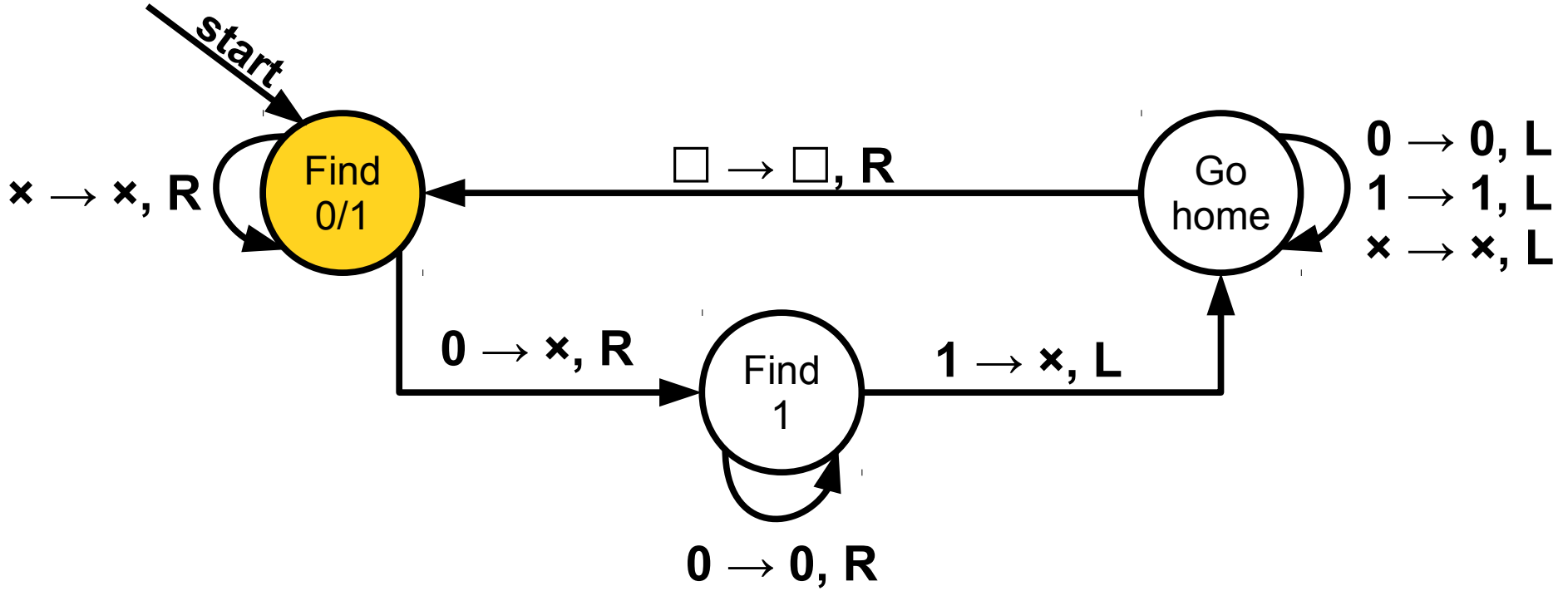

…
×
0
×
1
1
1
0
0
…

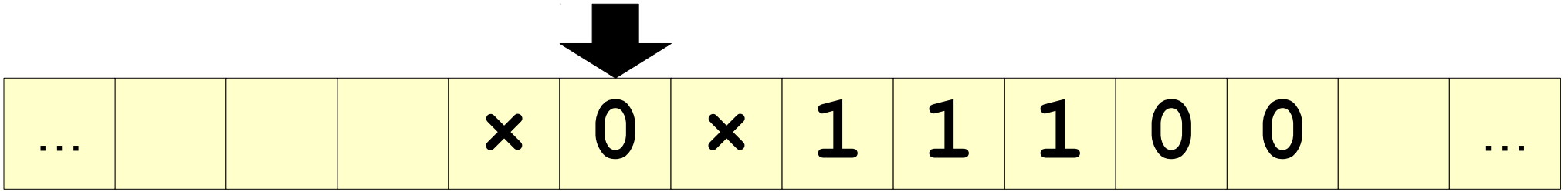

start
Find
0/1
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

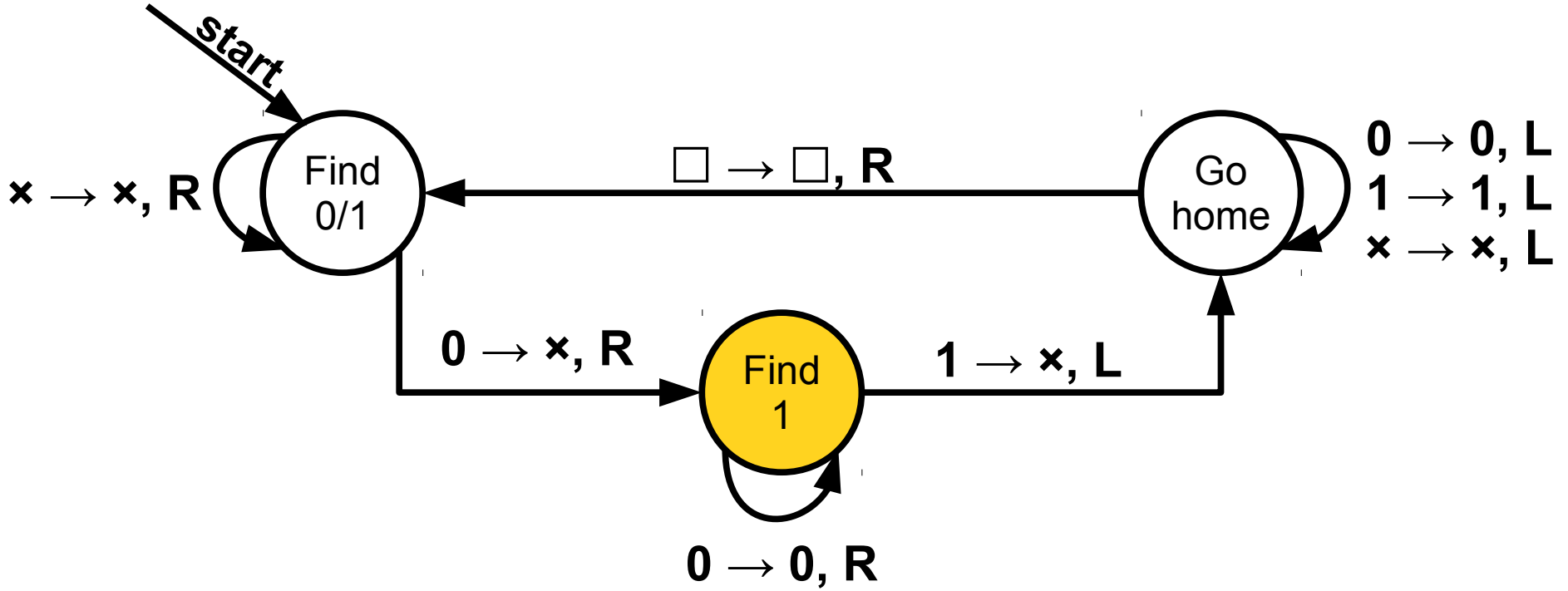

…
×
0
×
1
1
1
0
0
…

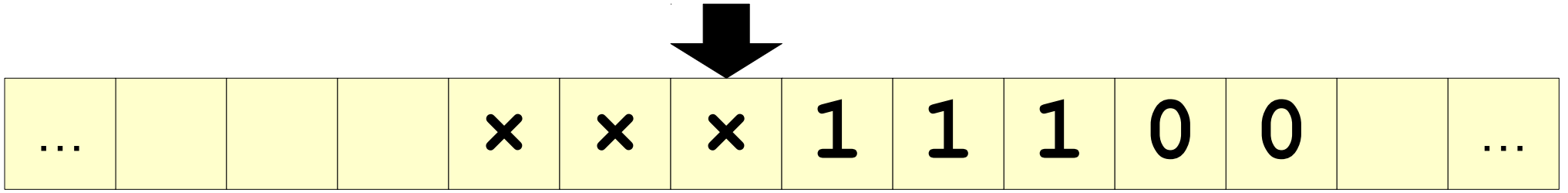

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

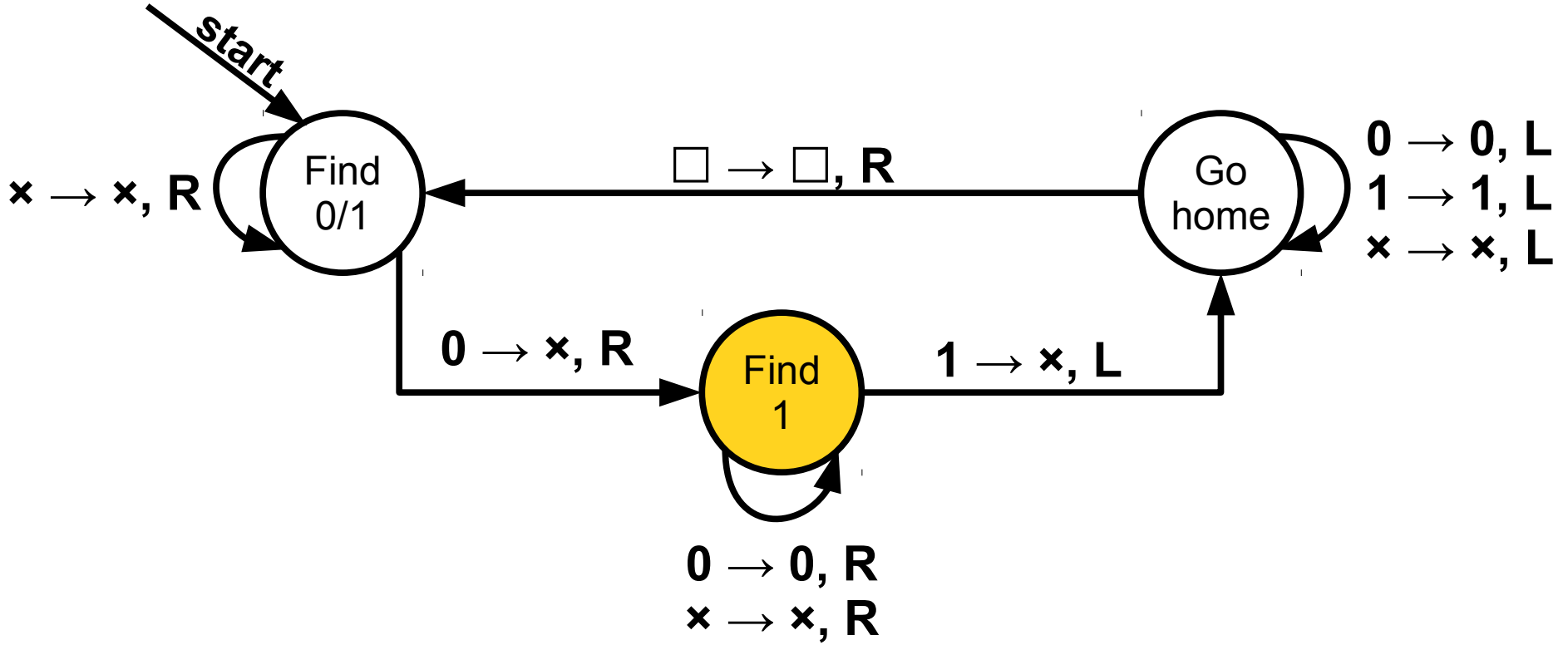

…
×
0
×
1
1
1
0
0
…

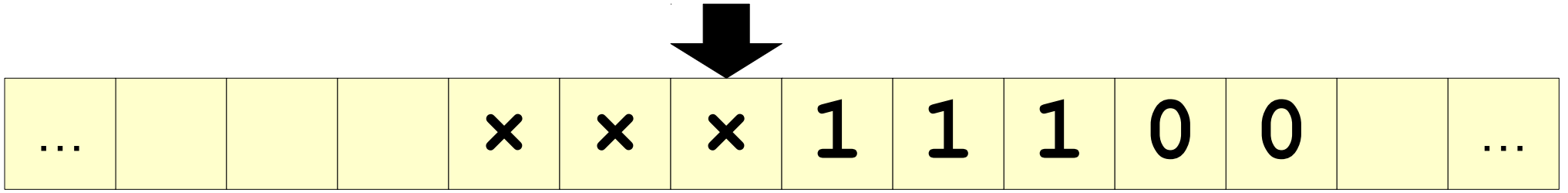

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

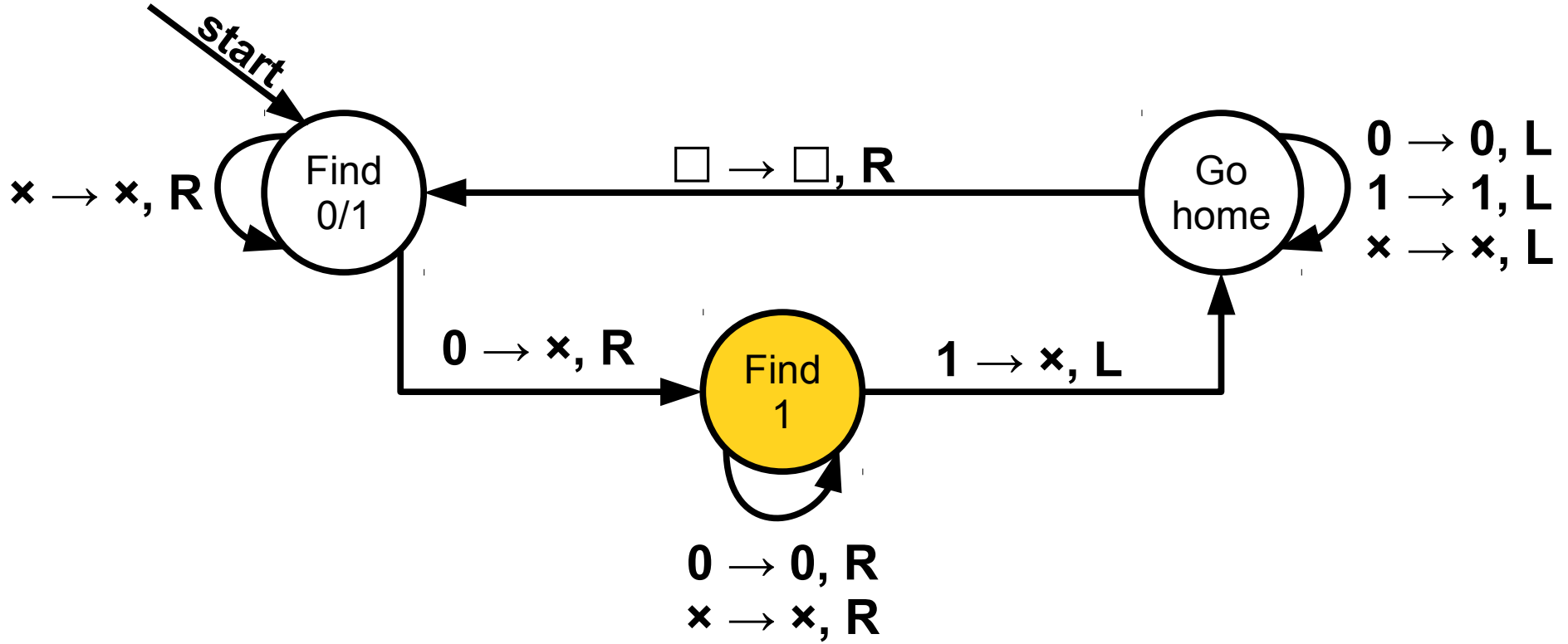

…
×
0
×
1
1
1
0
0
…

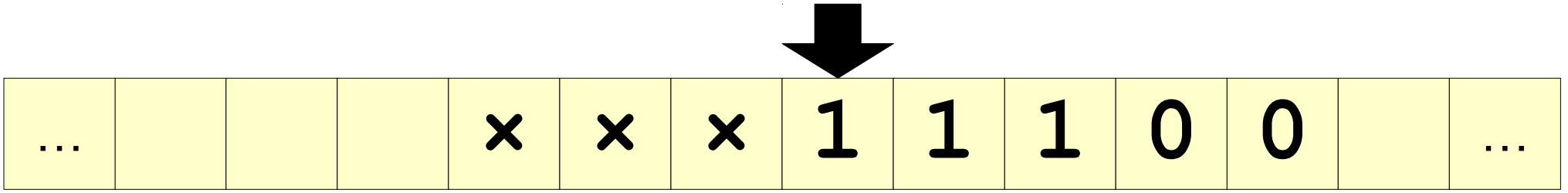

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R

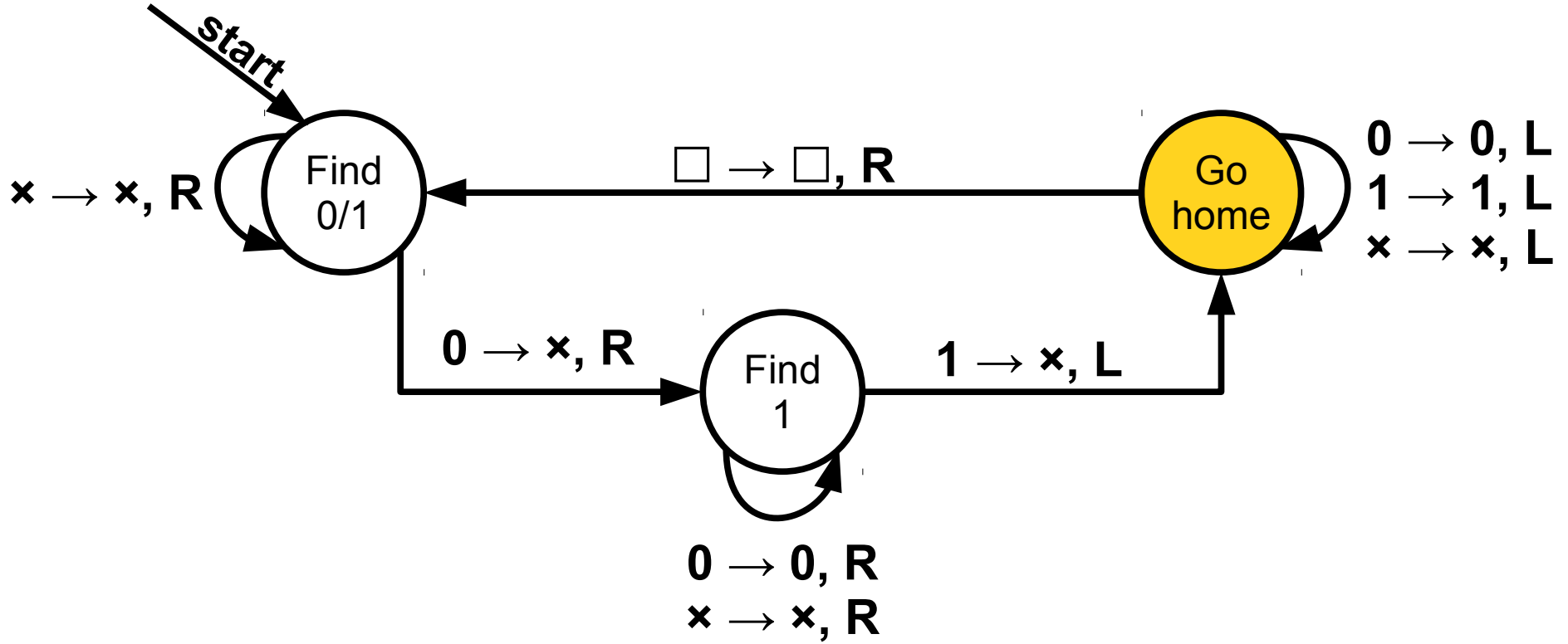

…
×
×
×
1
1
1
0
0
…

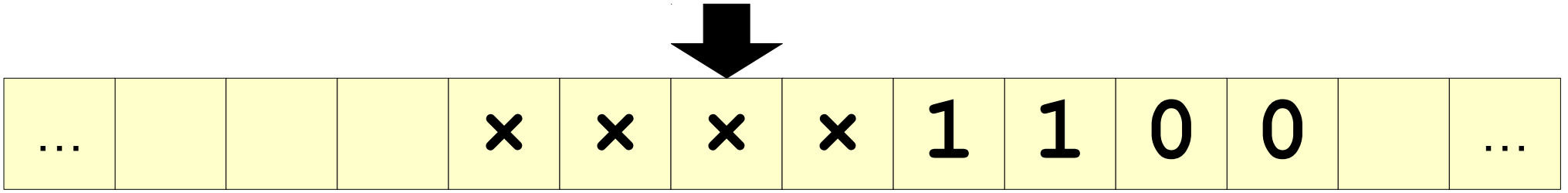

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

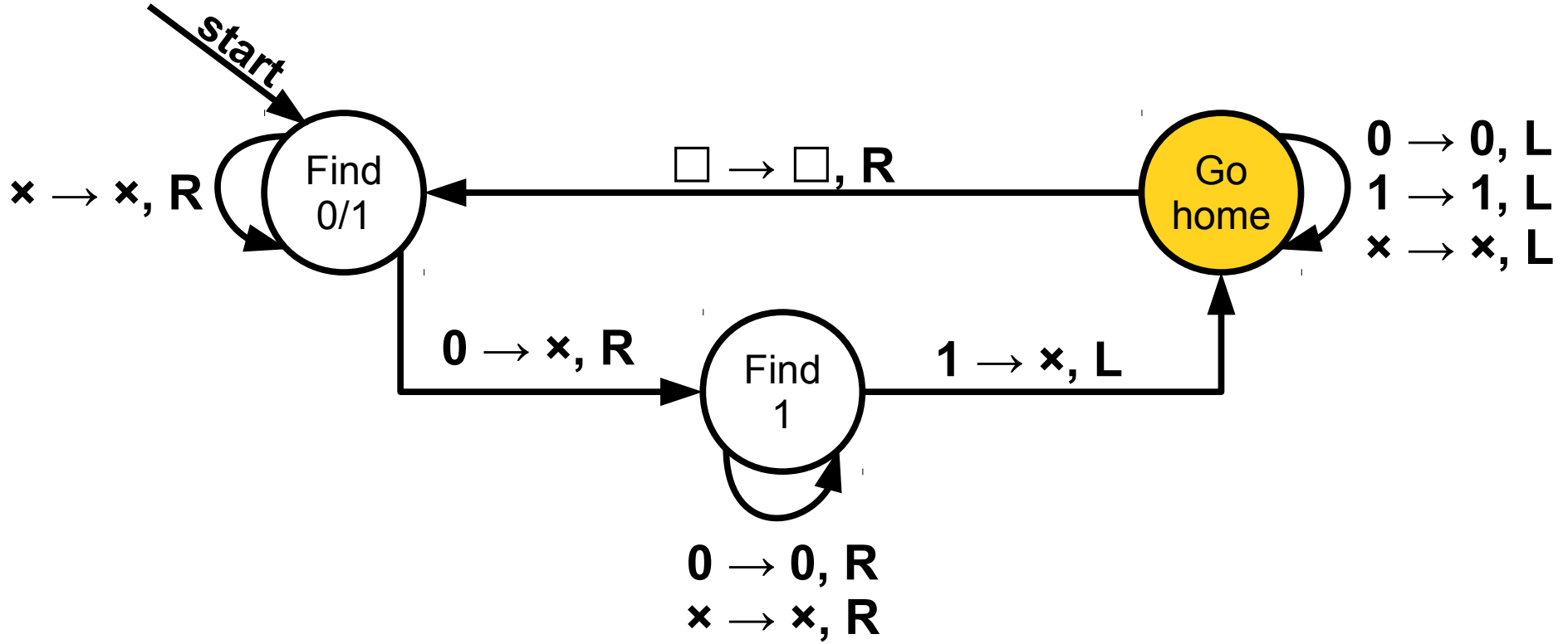

…
×
×
×
1
1
1
0
0
…

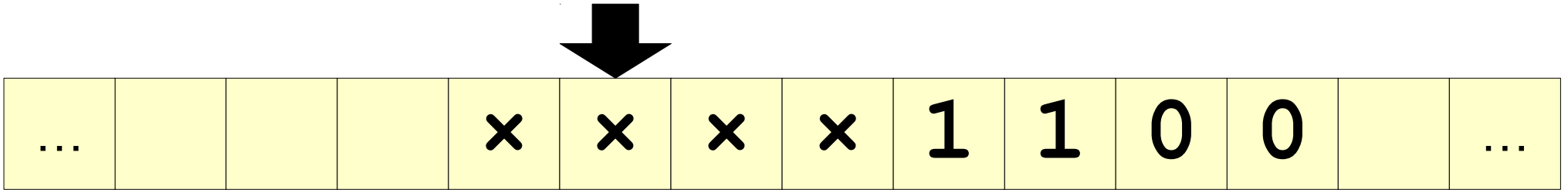

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

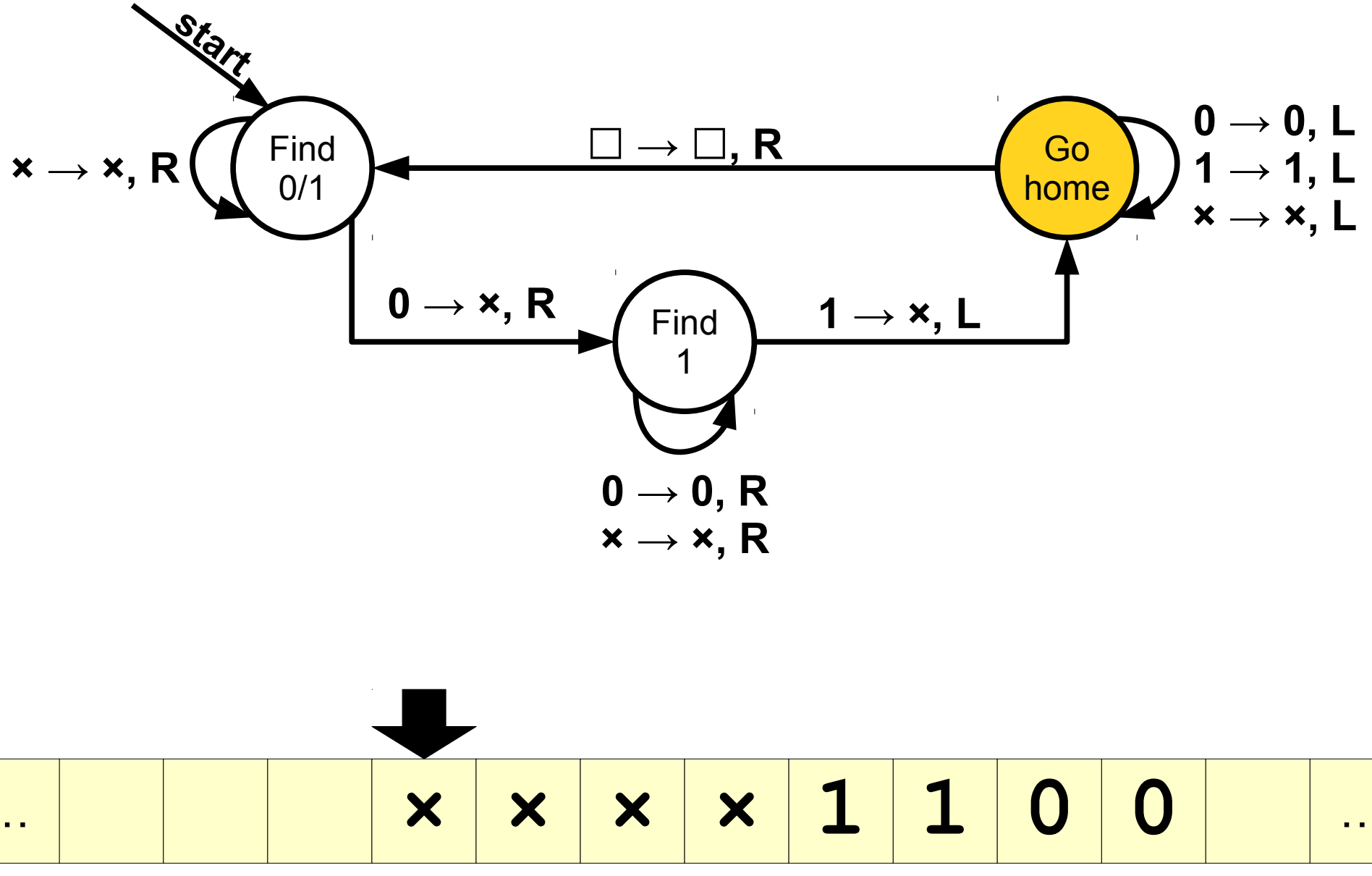

…
×
×
×
1
1
1
0
0
…

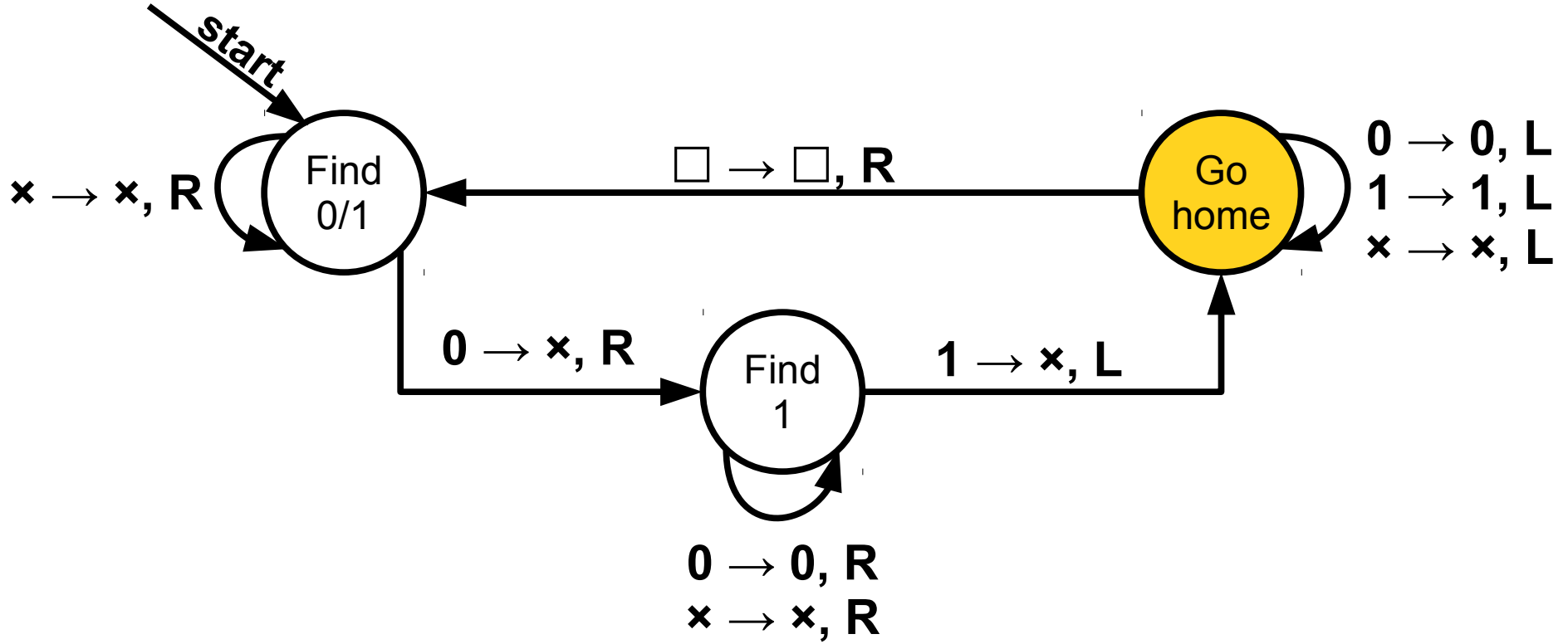

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

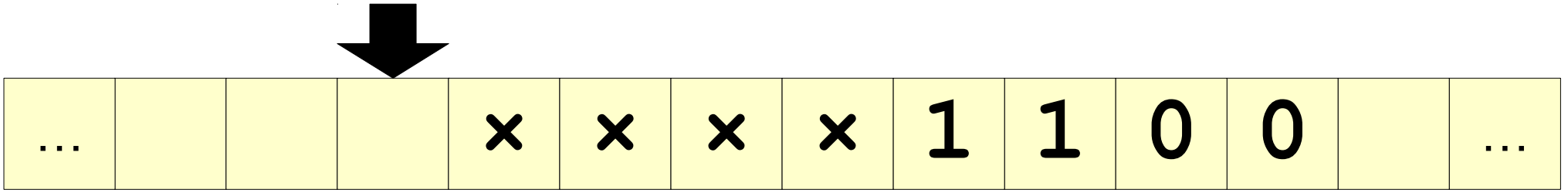

…
×
×
×
×
1
1
0
0
…

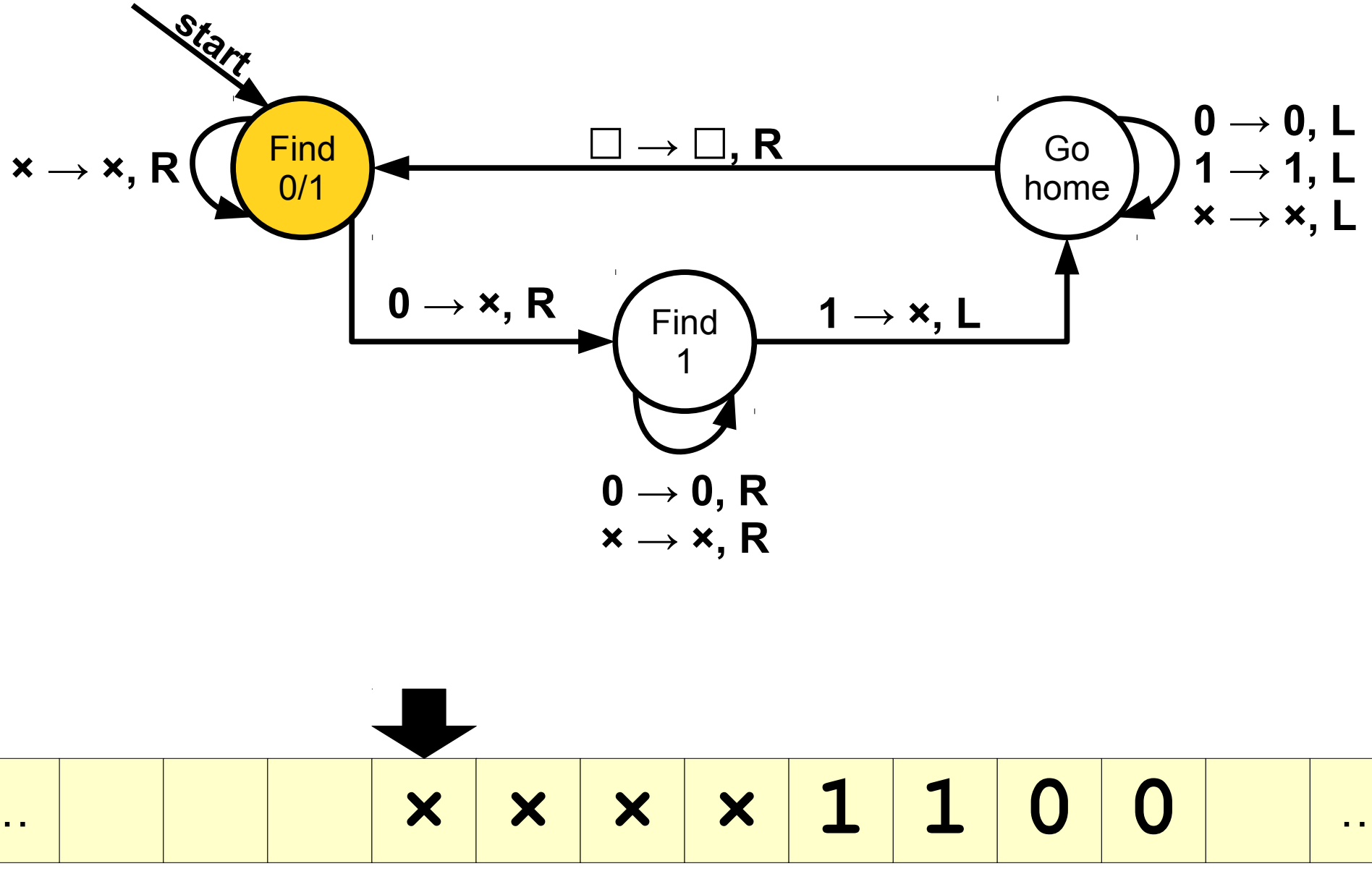

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

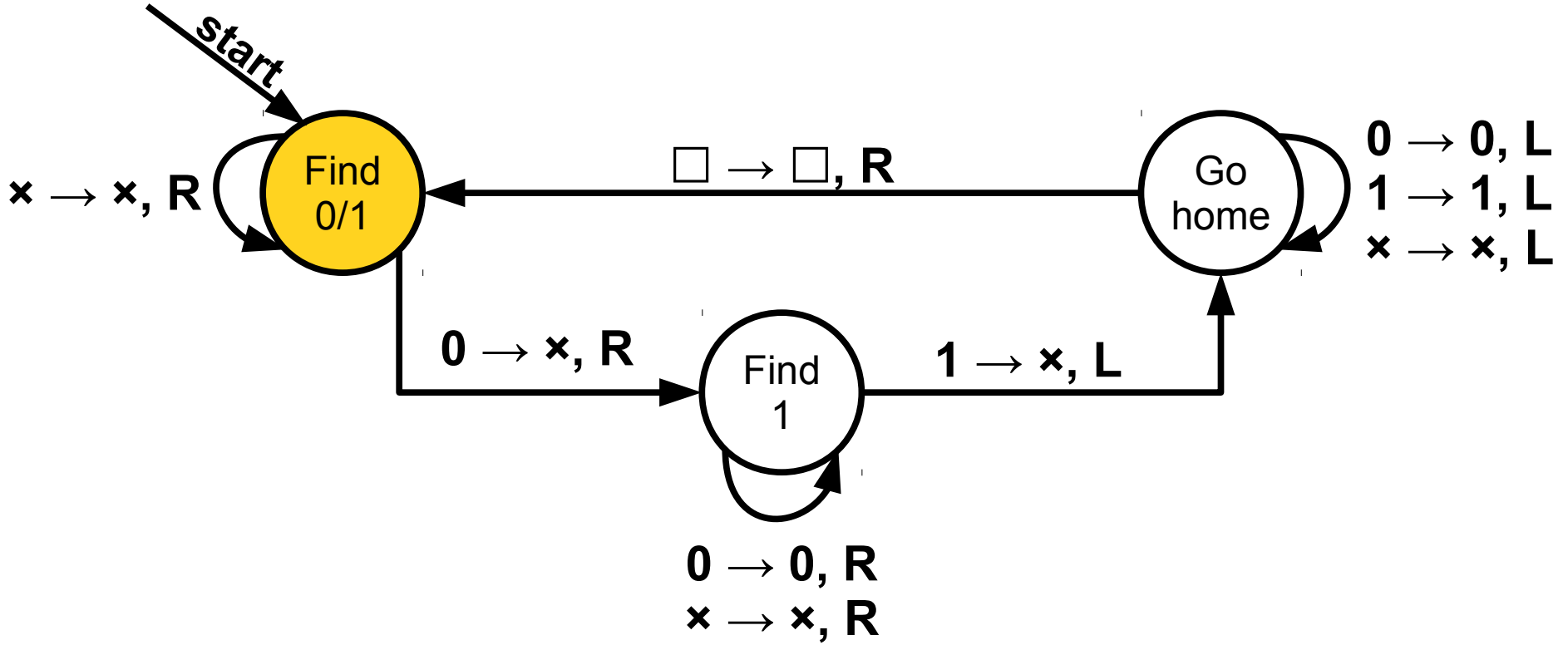

…
×
×
×
×
1
1
0
0
…

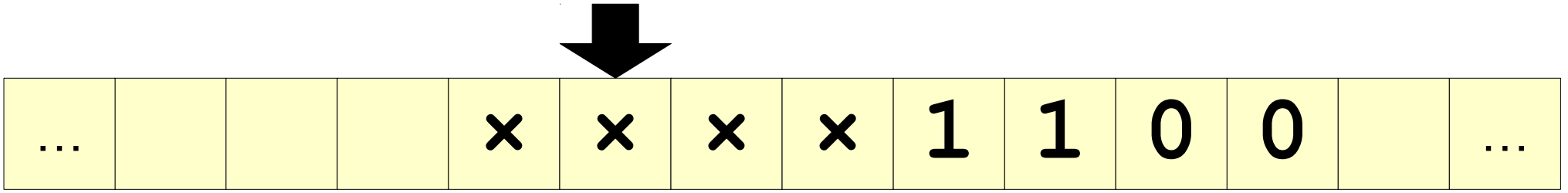

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R
… × × × × 1 1 0 0 …

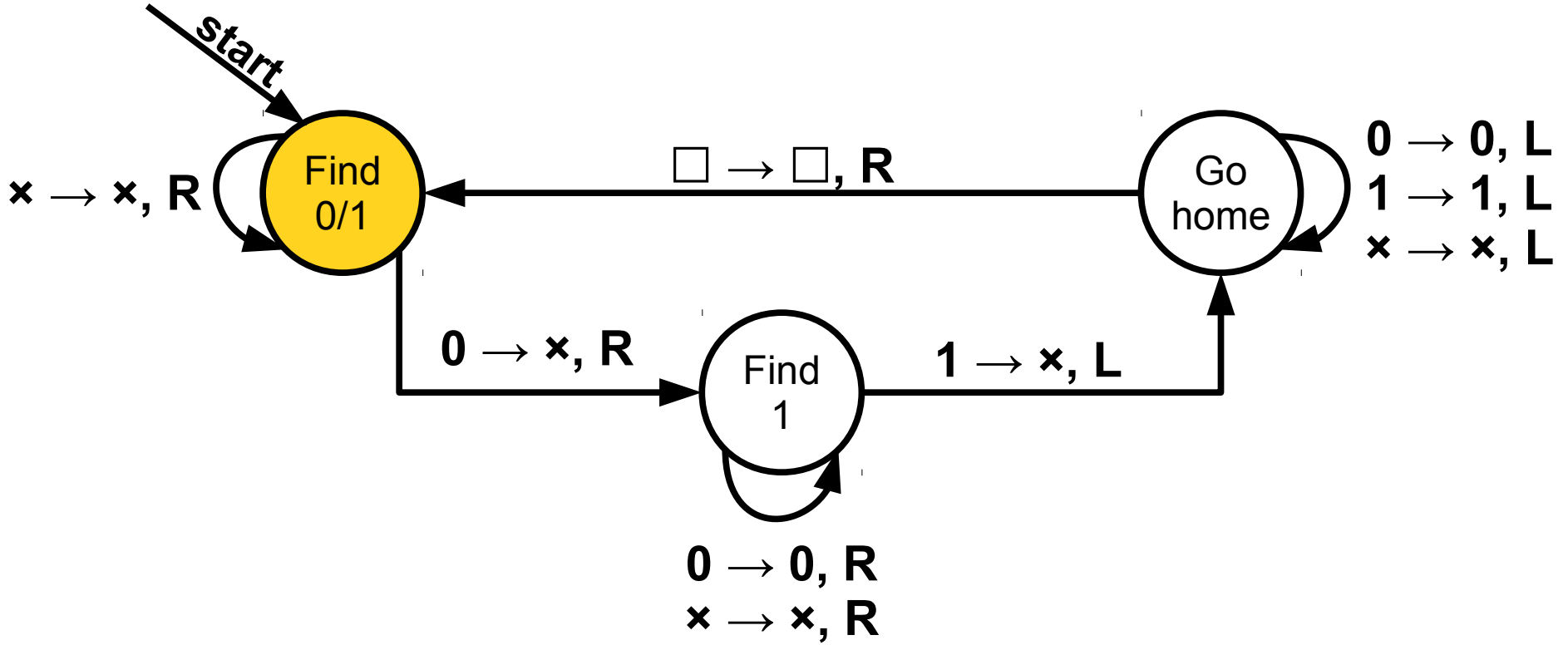

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

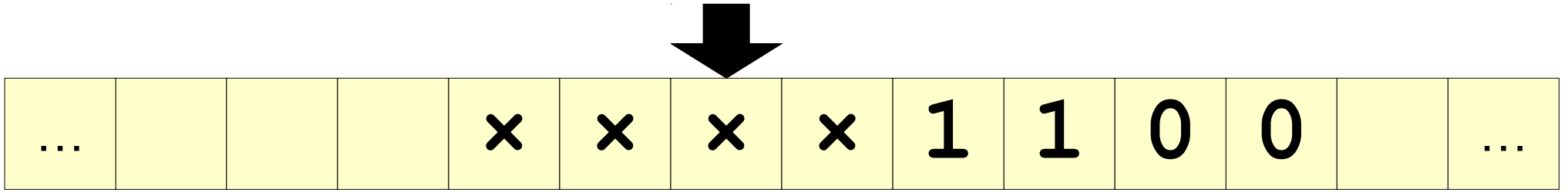

…
×
×
×
×
1
1
0
0
…

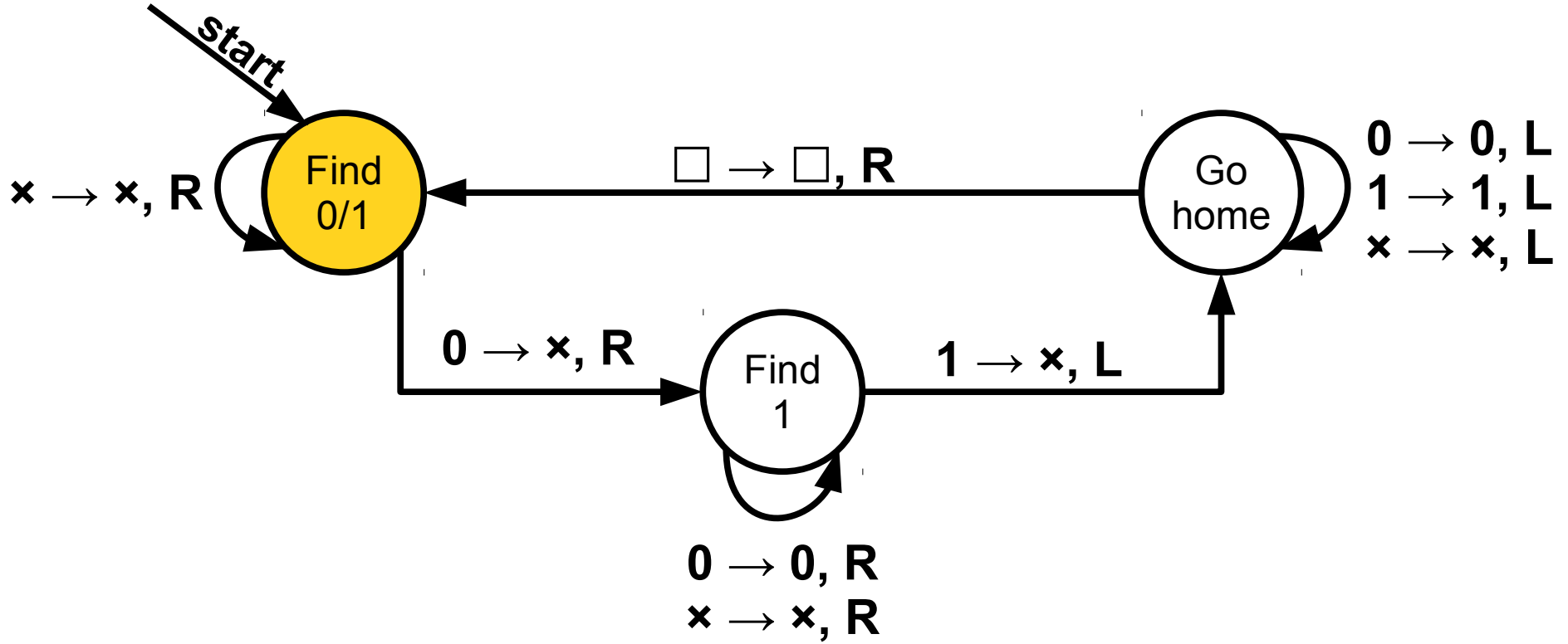

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R
… × × × × 1 1 0 0 …

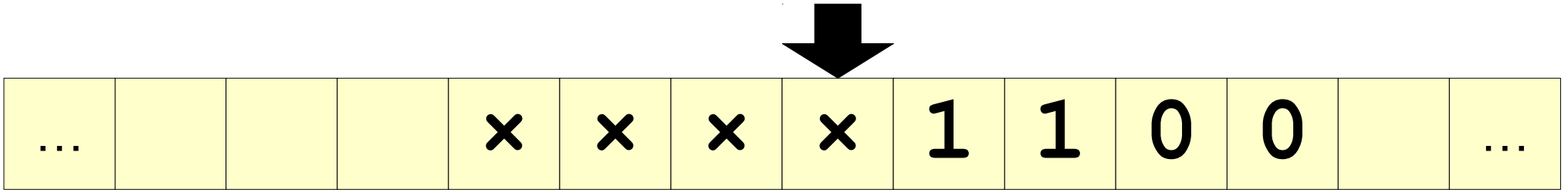

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

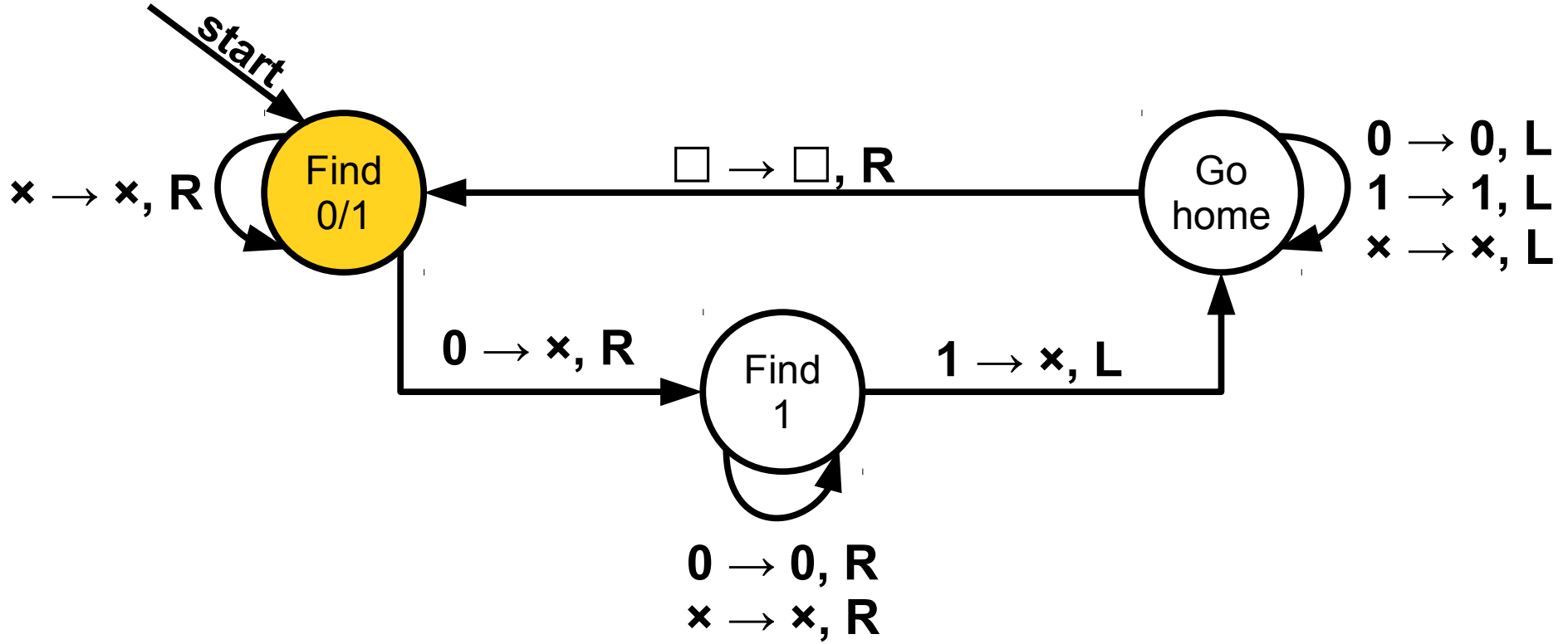

…
×
×
×
×
1
1
0
0
…

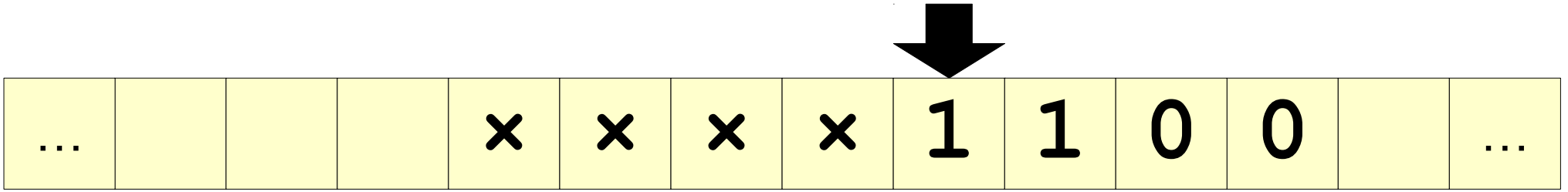

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

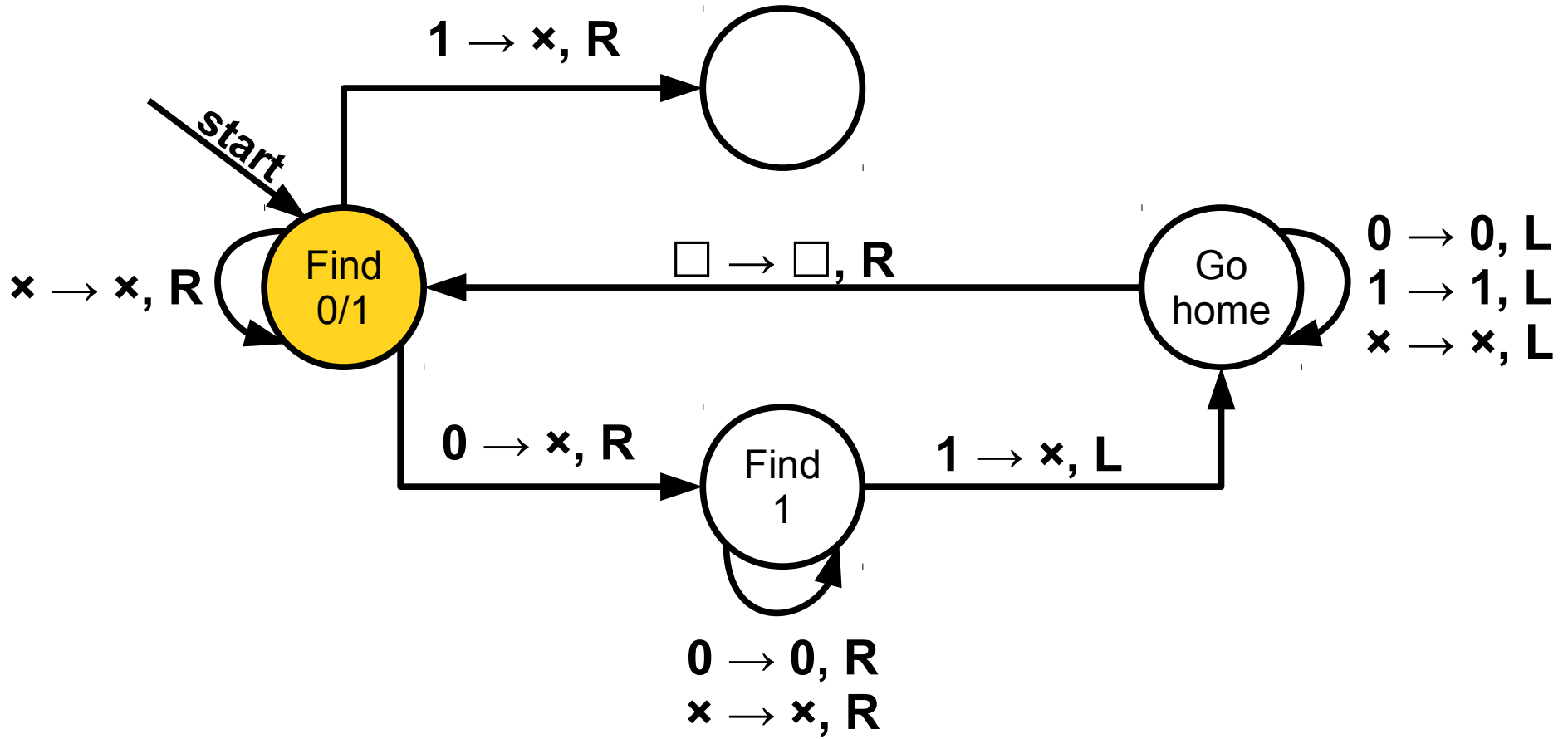

…
×
×
×
×
1
1
0
0
…

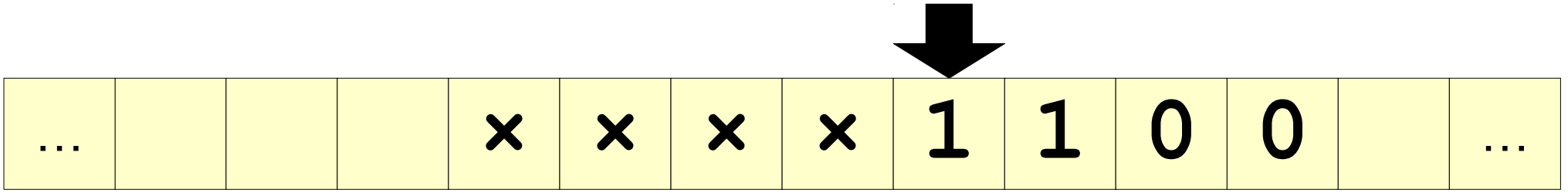

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

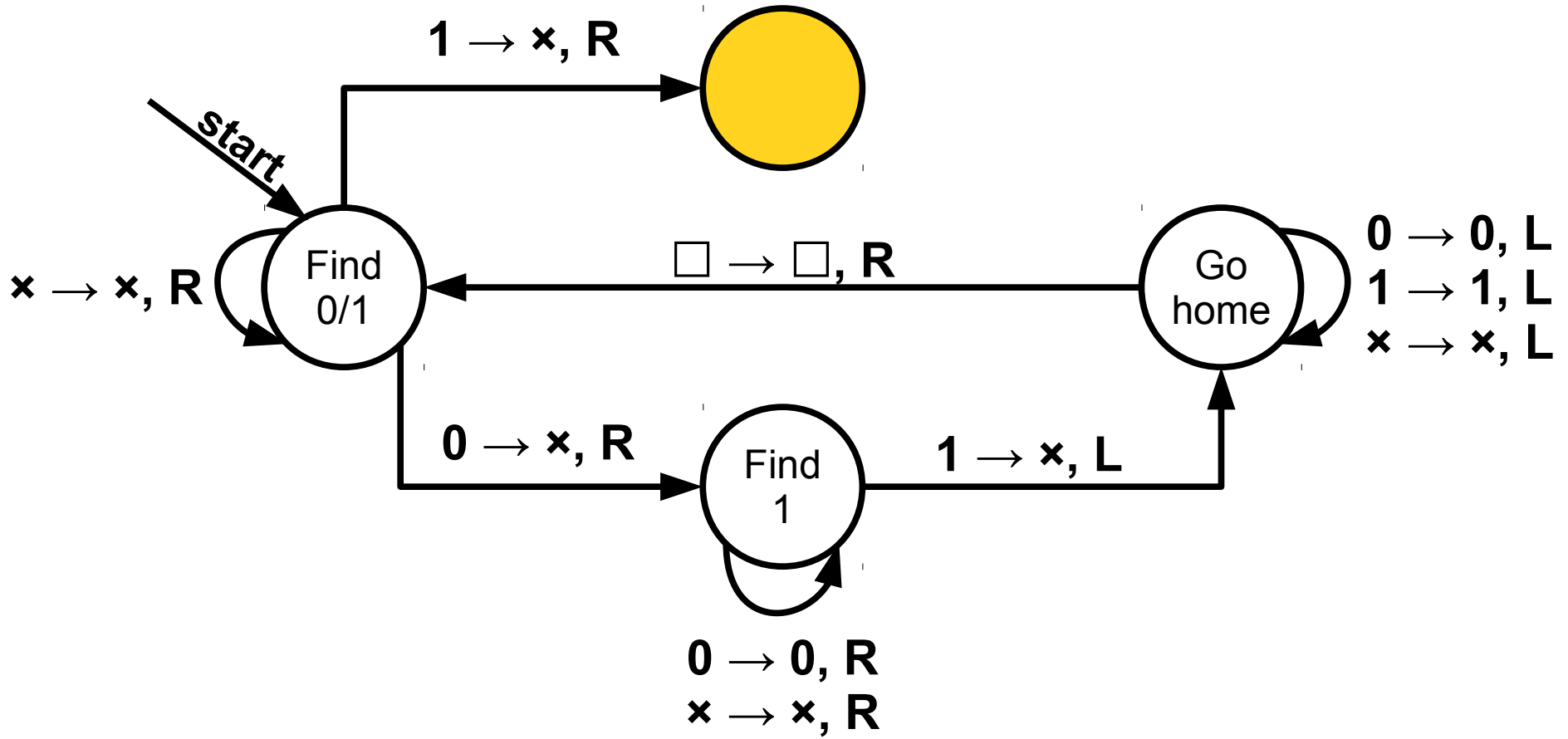

…
×
×
×
×
1
1
0
0
…

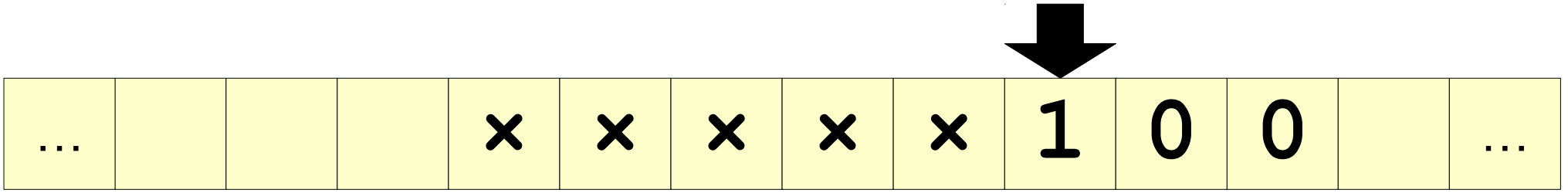

start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

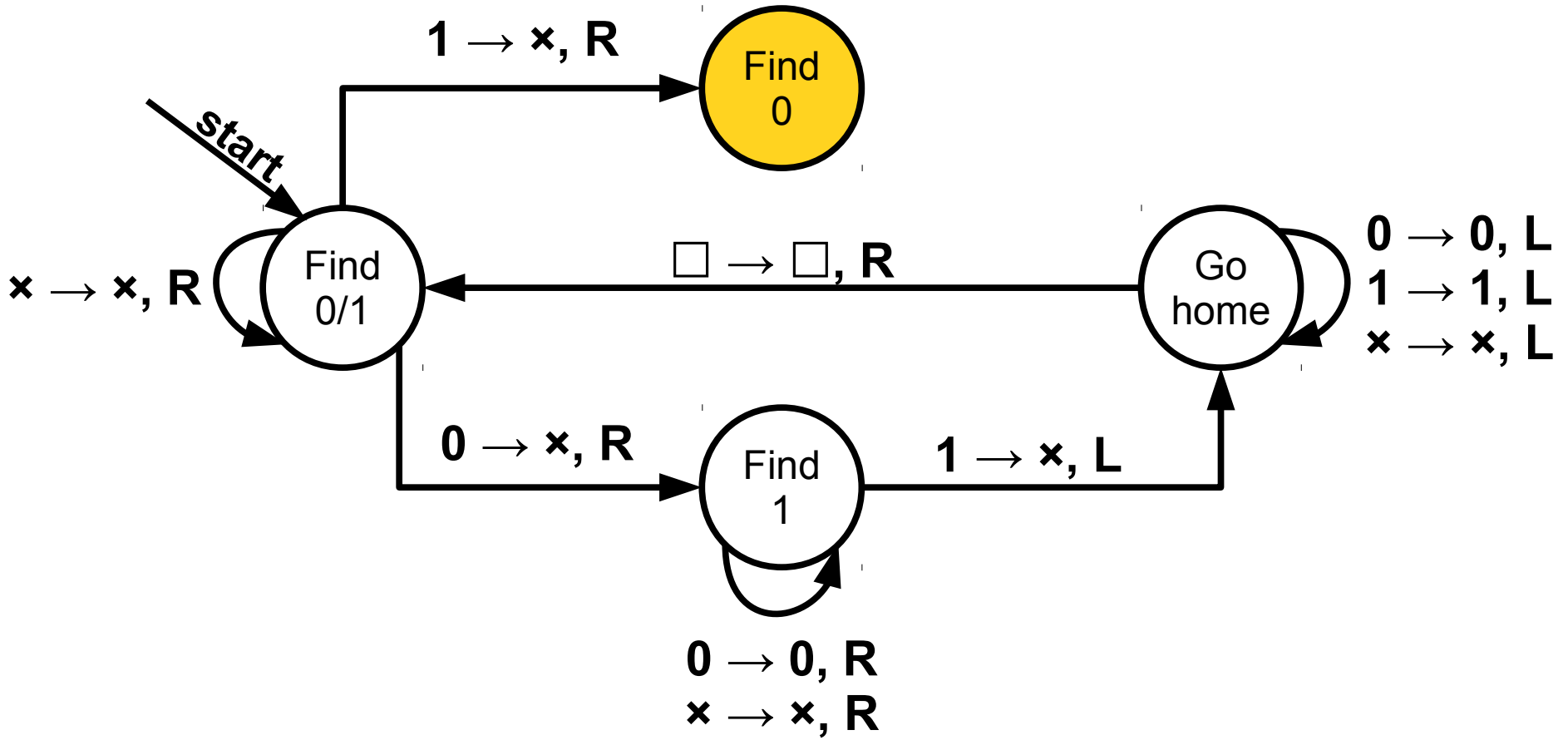

…
×
×
×
×
1
1
0
0
…

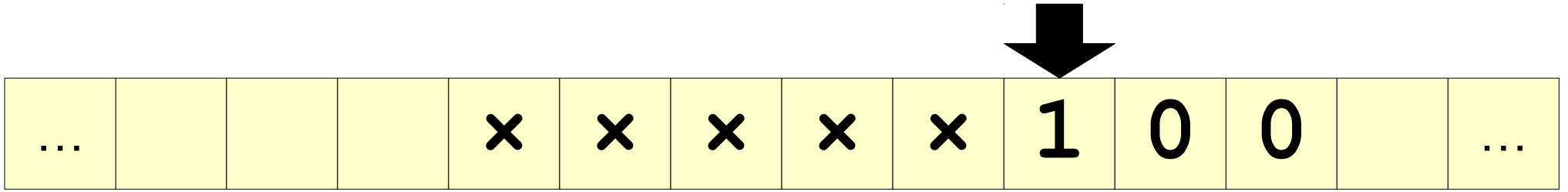

1 → ×, R
start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

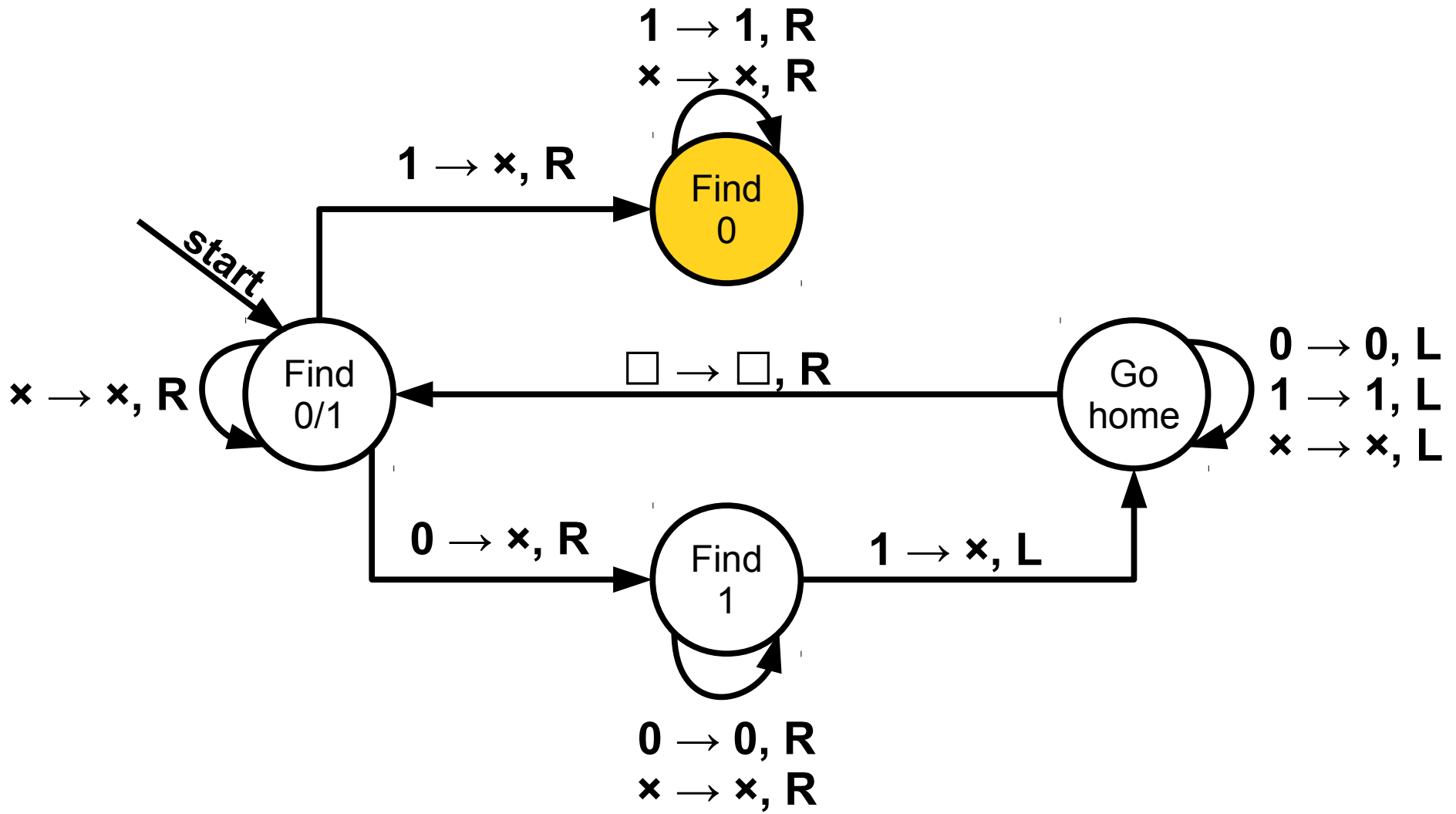

…
×
×
×
×
1
1
0
0
…

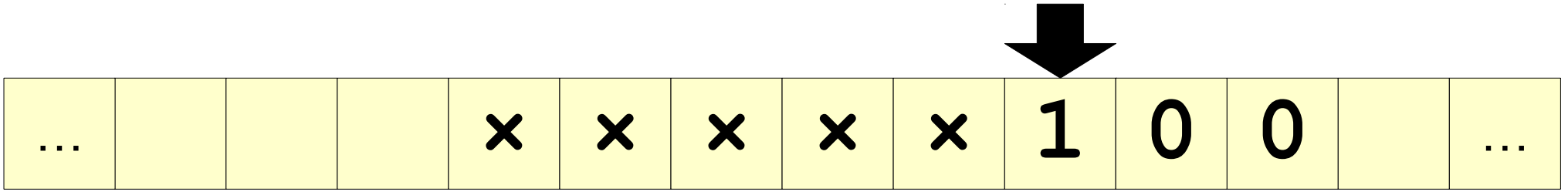

1 → ×, R
start
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

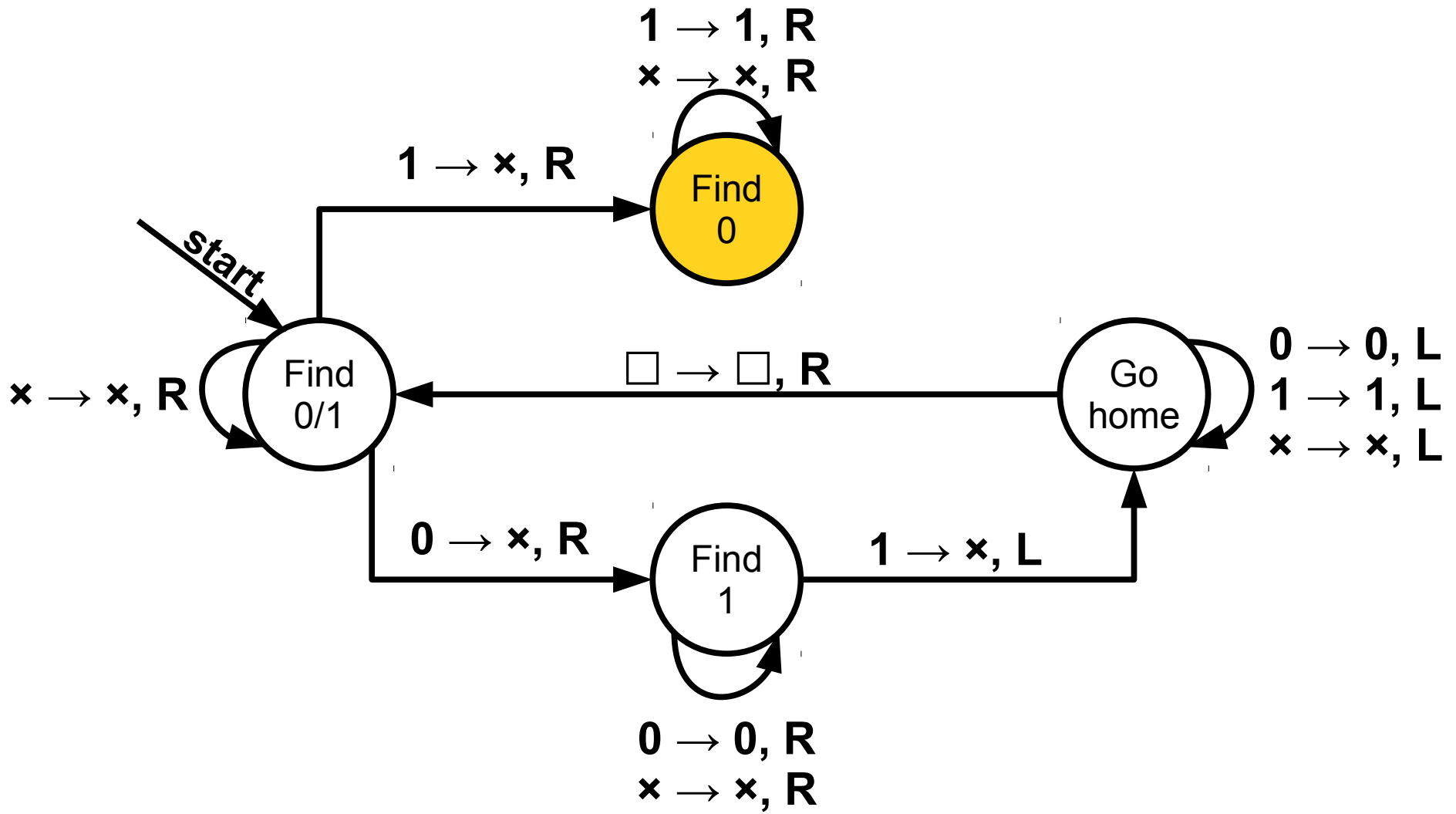

…
×
×
×
×
×
1
0
0
…

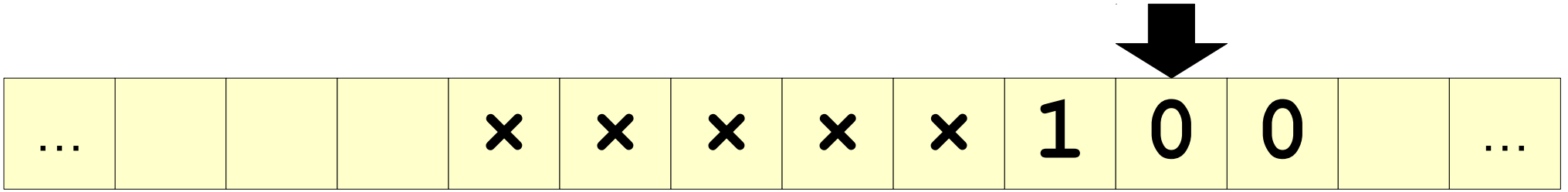

1 → ×, R
Find 0
start
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find 1
1 → ×, L
0 → 0, R
× → ×, R

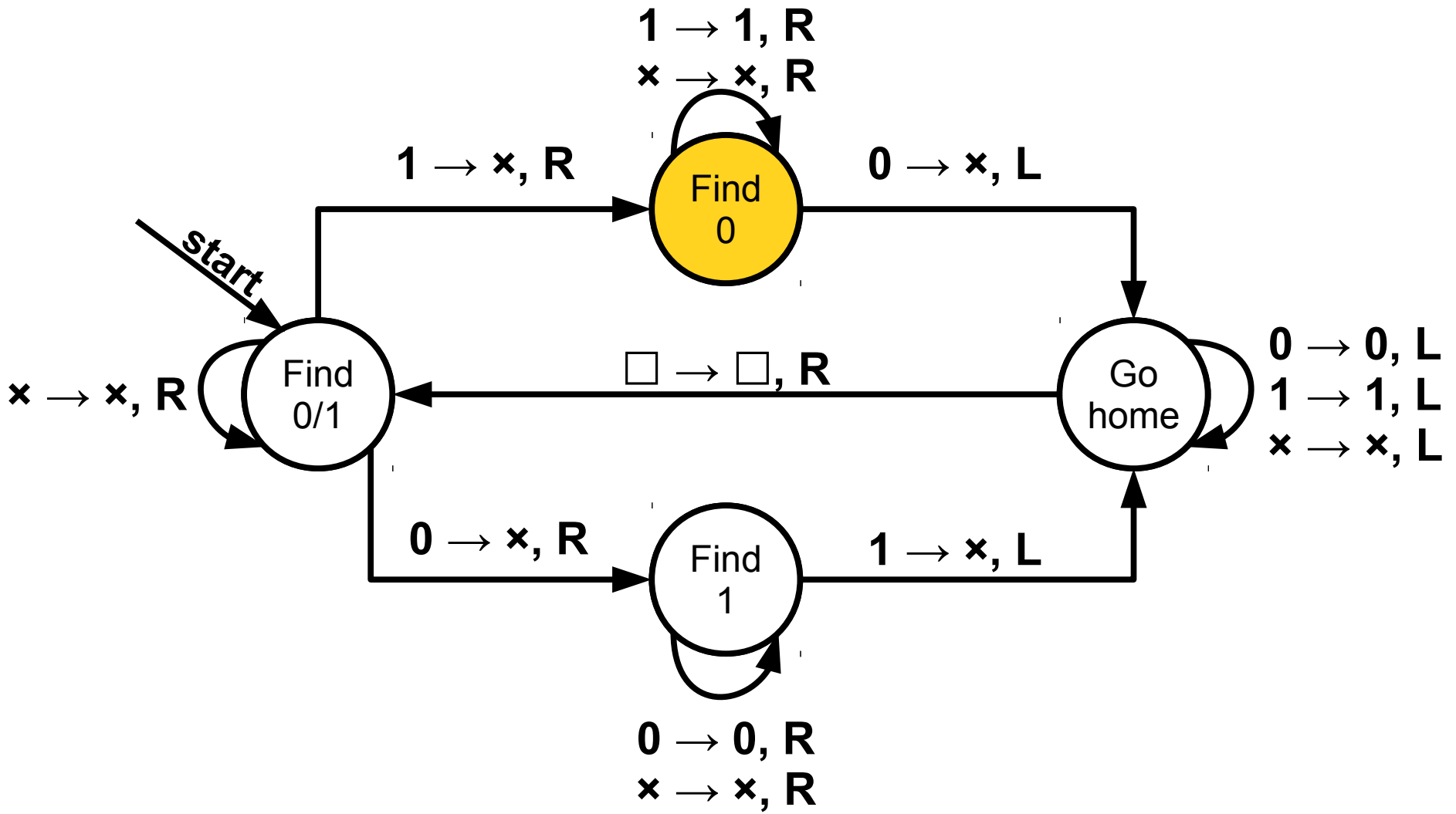

…
×
×
×
×
×
1
0
0
…

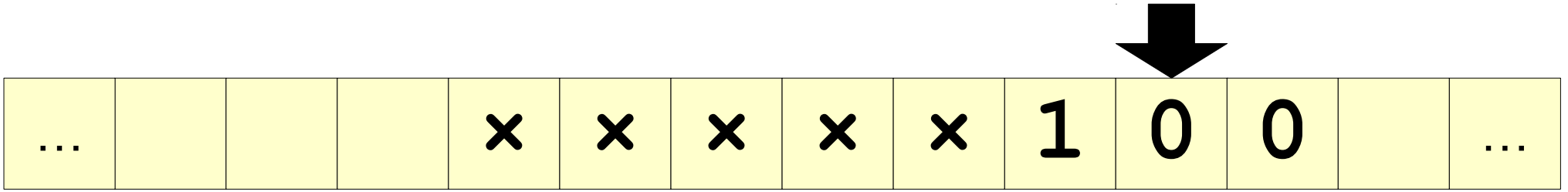

1 → 1, R
× → ×, R
1 → ×, R
Find 0
start
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find 1
1 → ×, L
0 → 0, R
× → ×, R

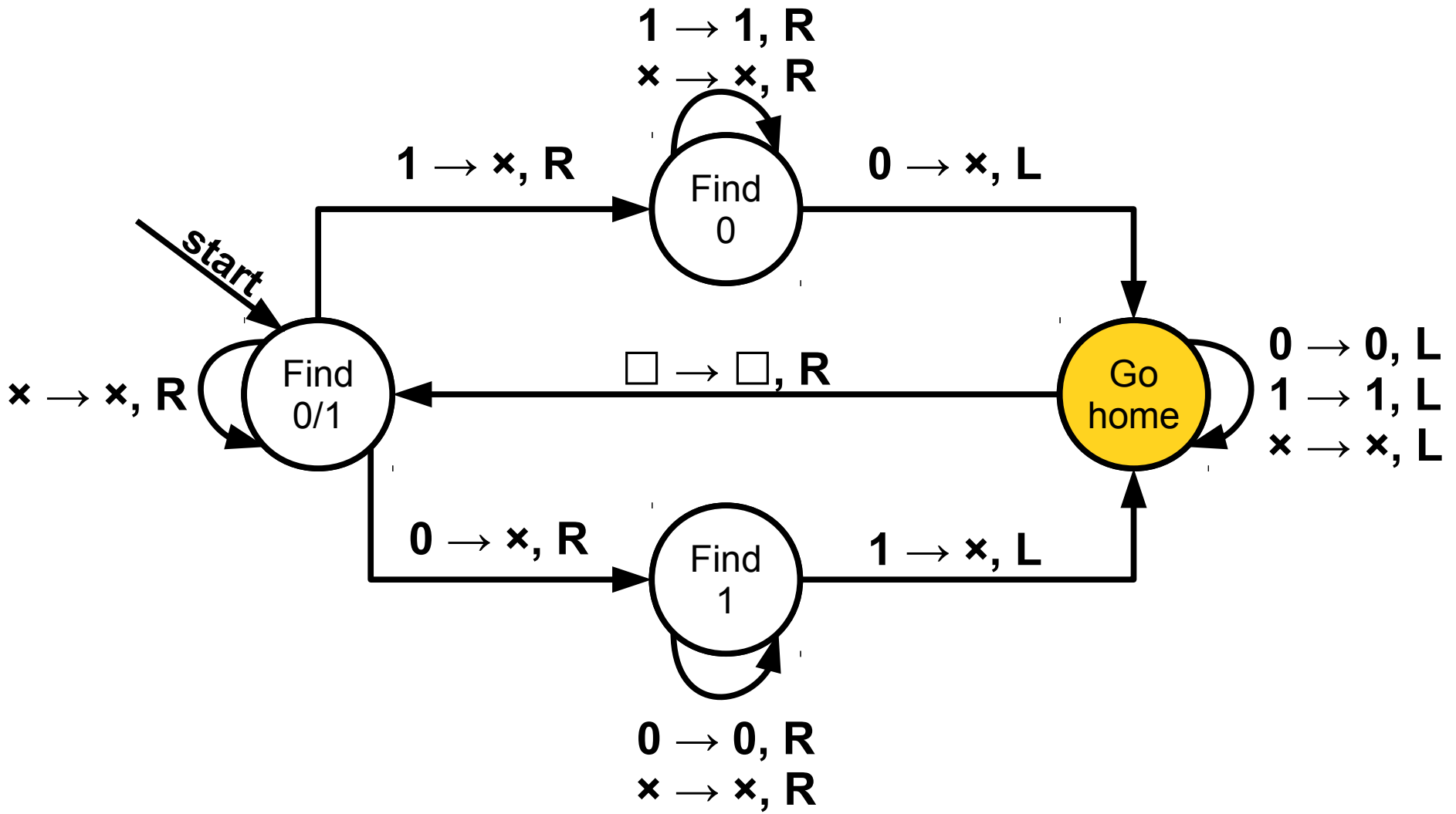

…
×
×
×
×
×
1
0
0
…

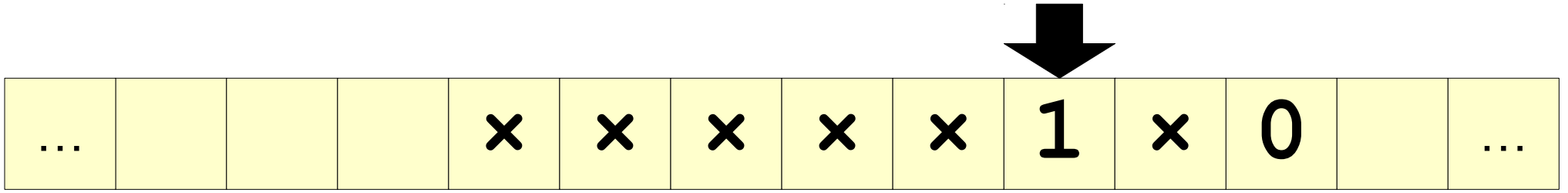

1 → 1, R
× → ×, R
1 → ×, R
Find 0
start
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find 1
1 → ×, L
0 → 0, R
× → ×, R

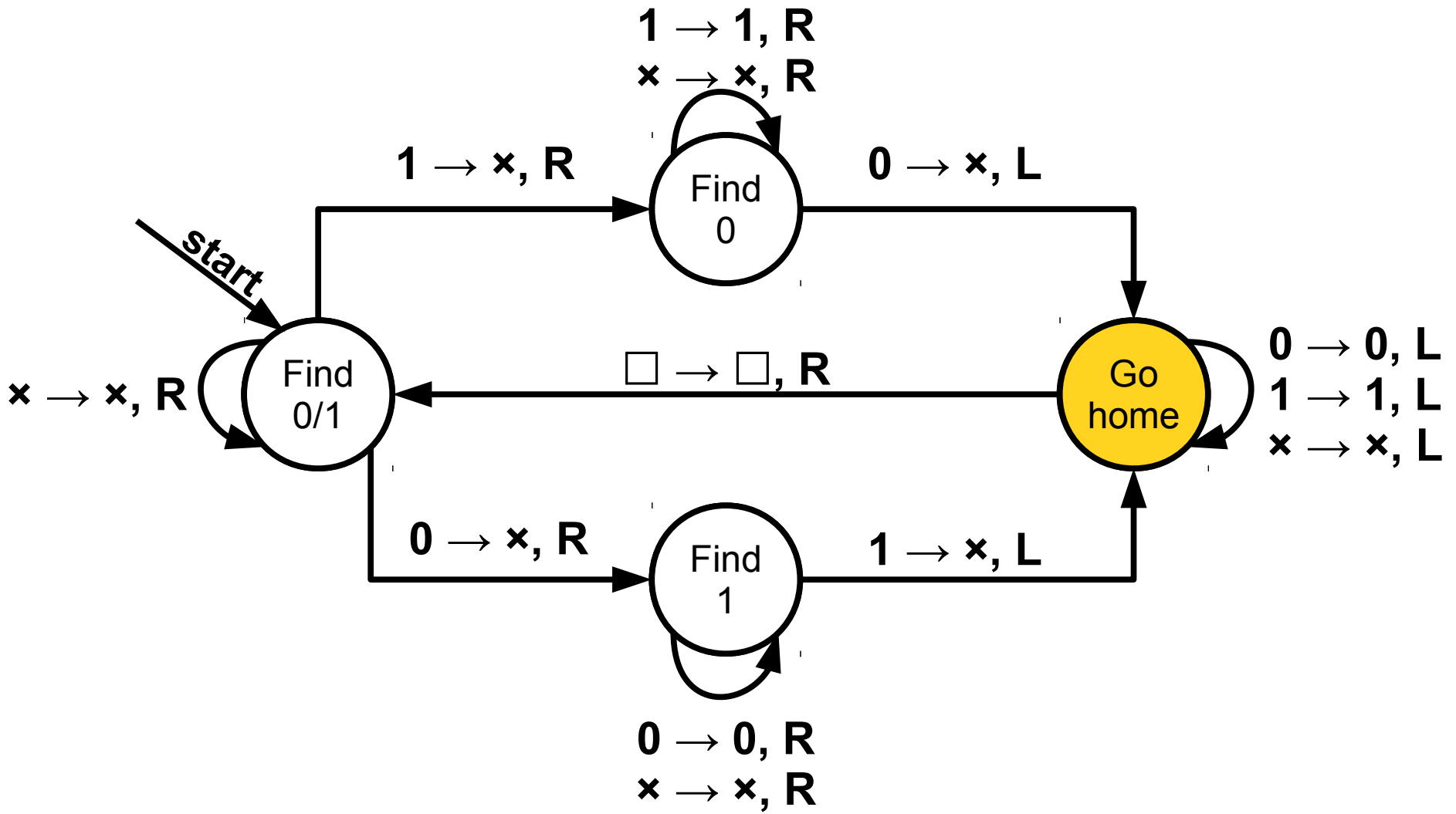

…
×
×
×
×
×
1
0
0
…

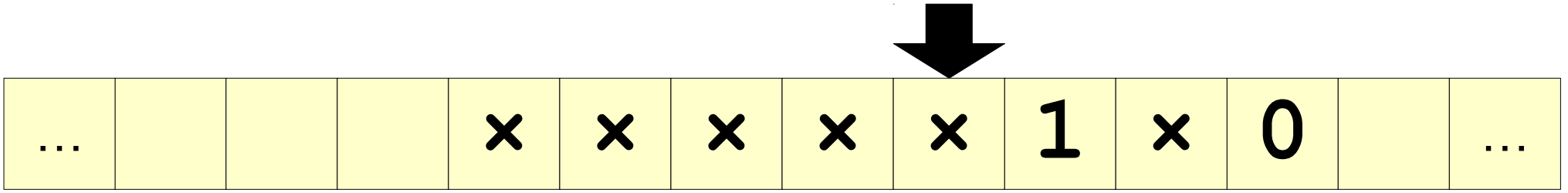

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

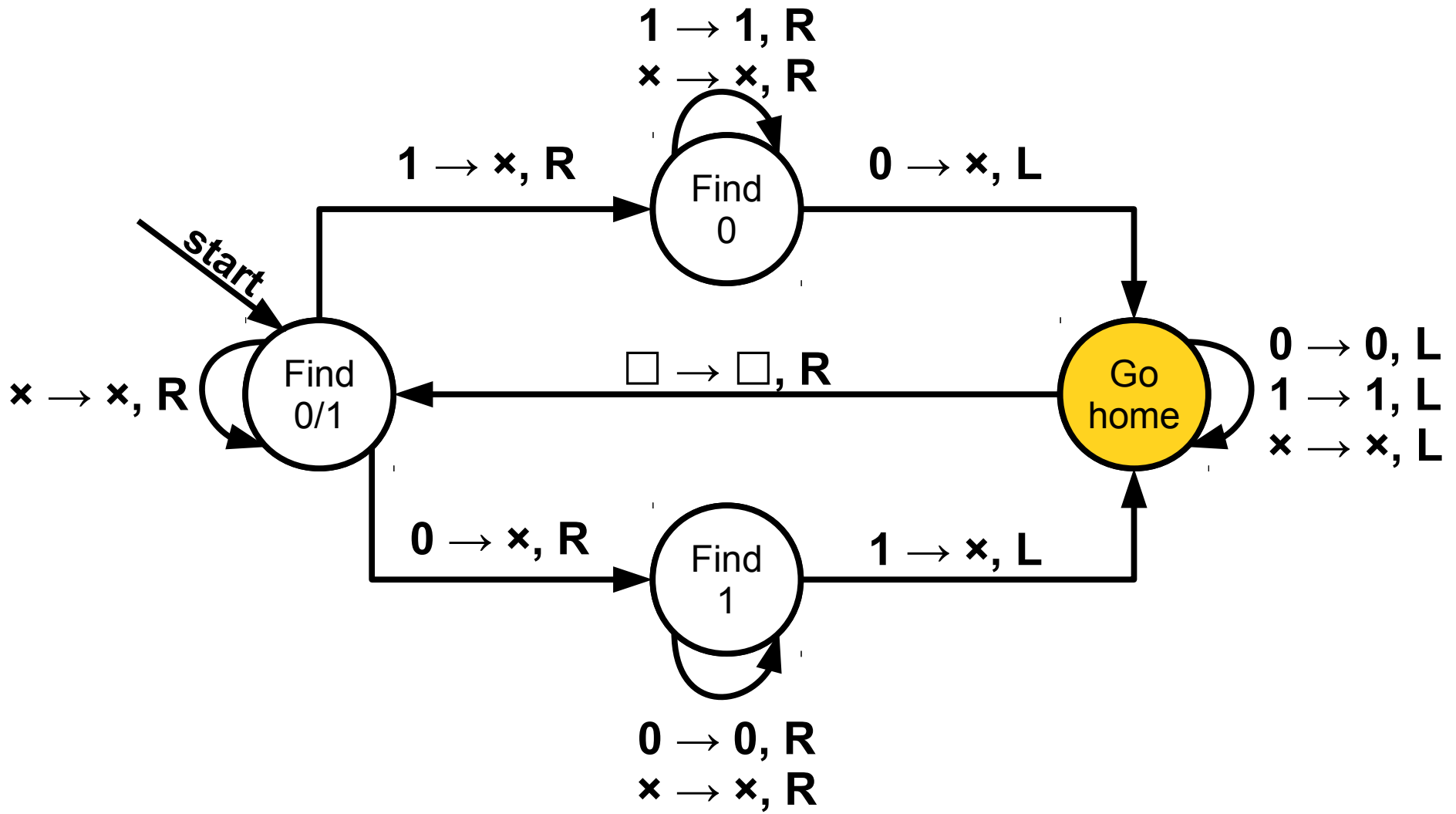

…
×
×
×
×
×
1
0
0
…

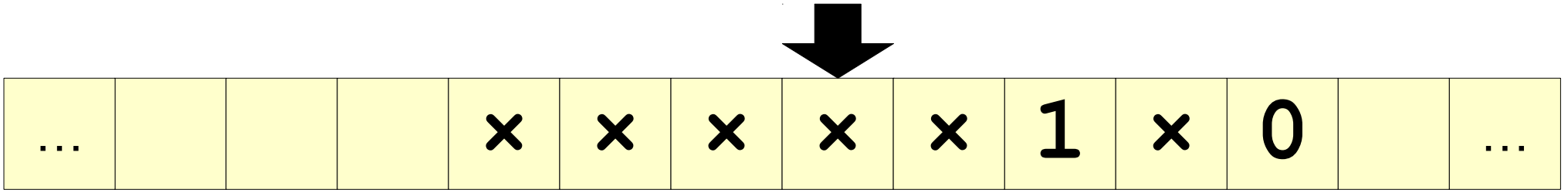

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

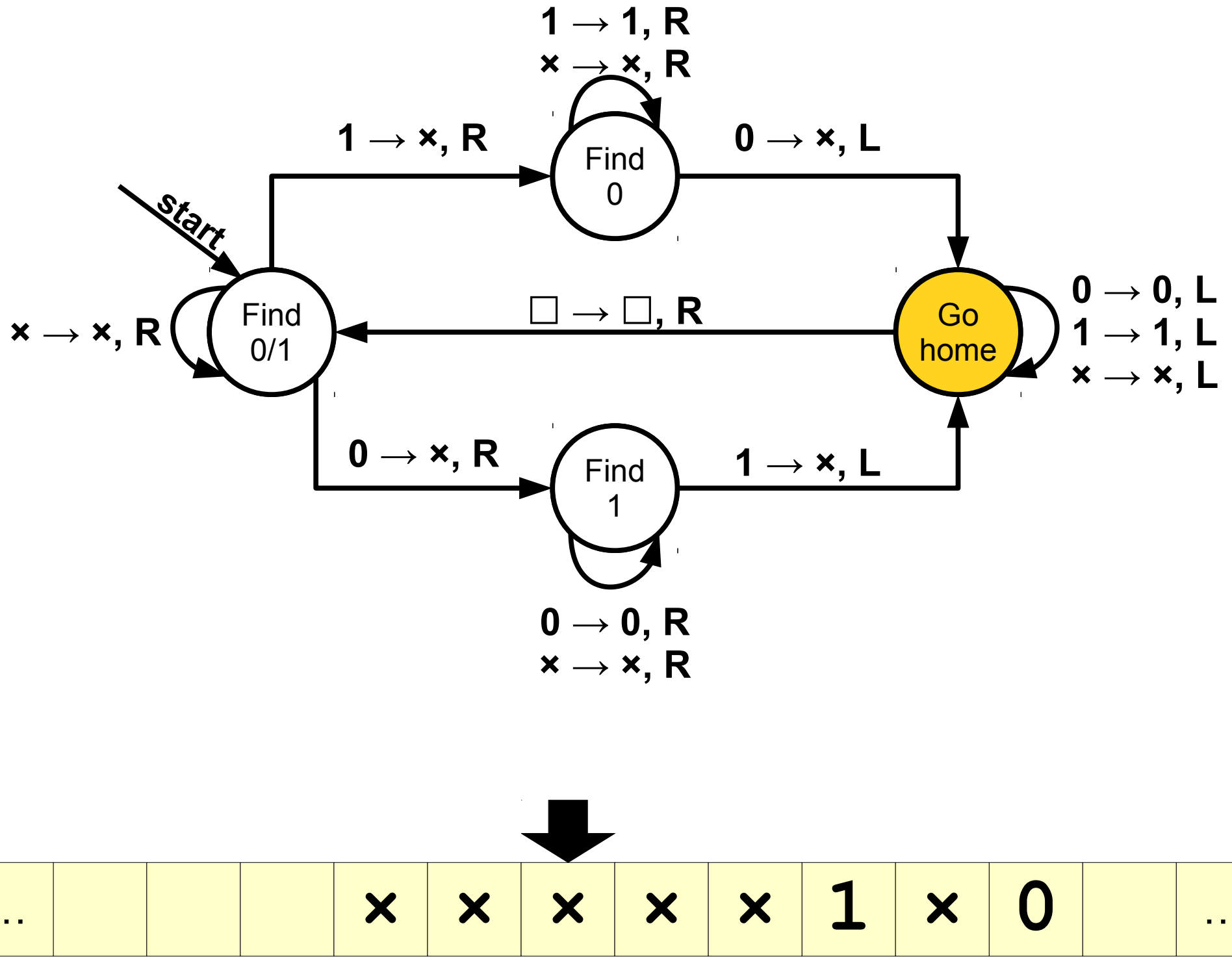

…
×
×
×
×
×
1
×
0
…

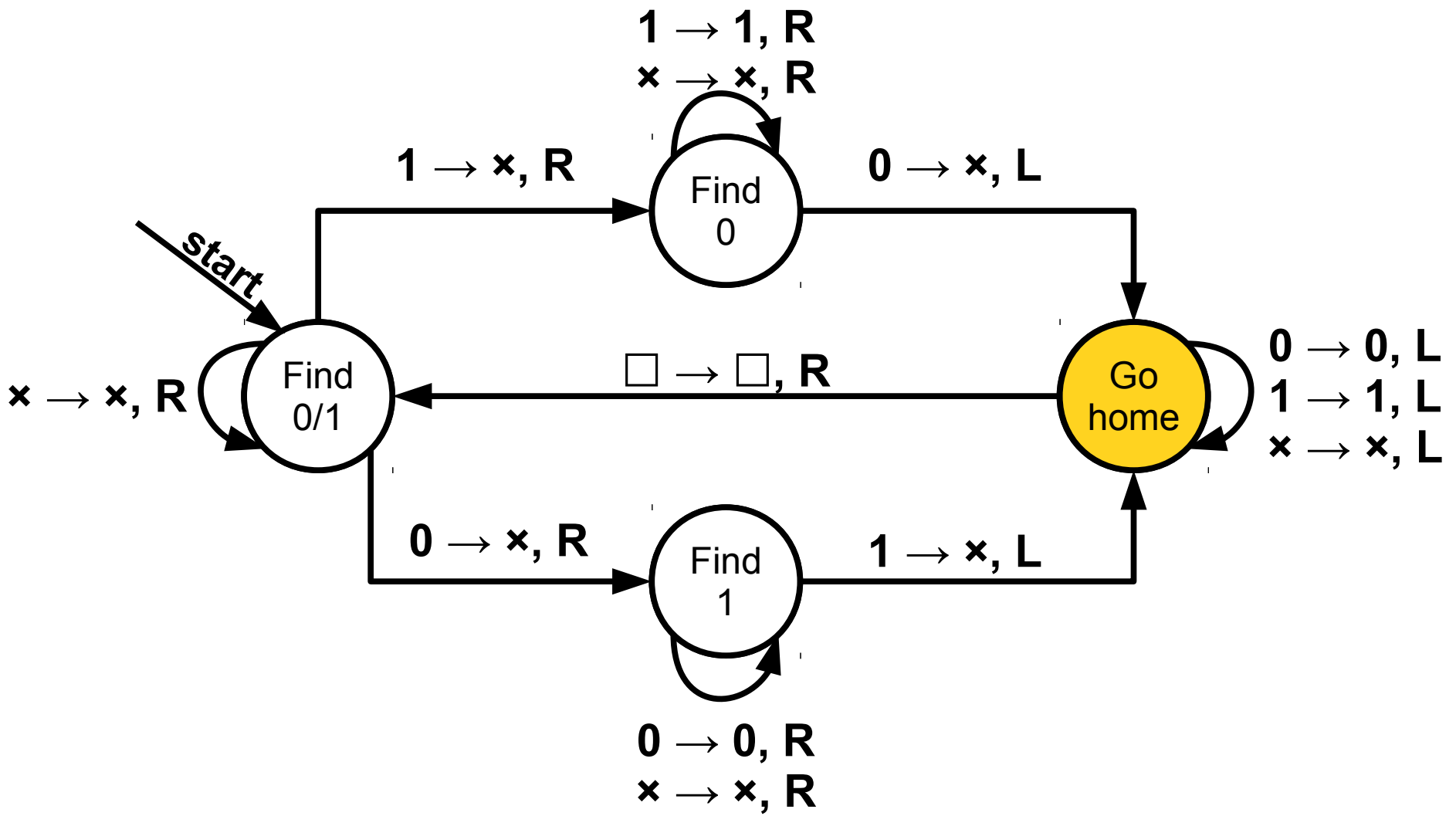

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

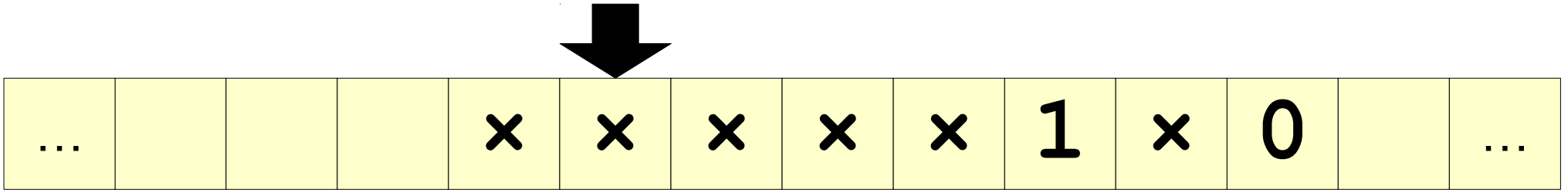

…
×
×
×
×
×
1
×
0
…

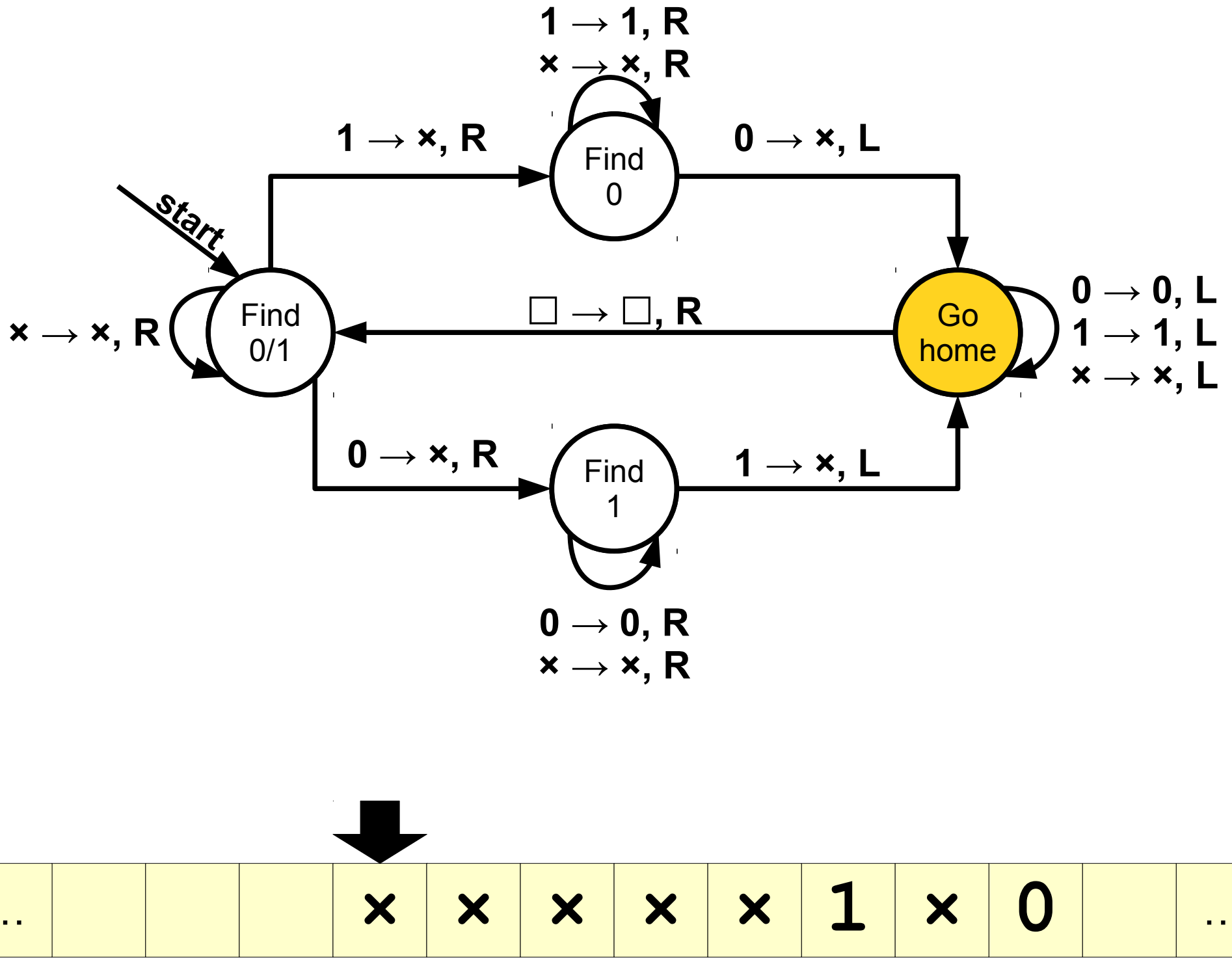

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R

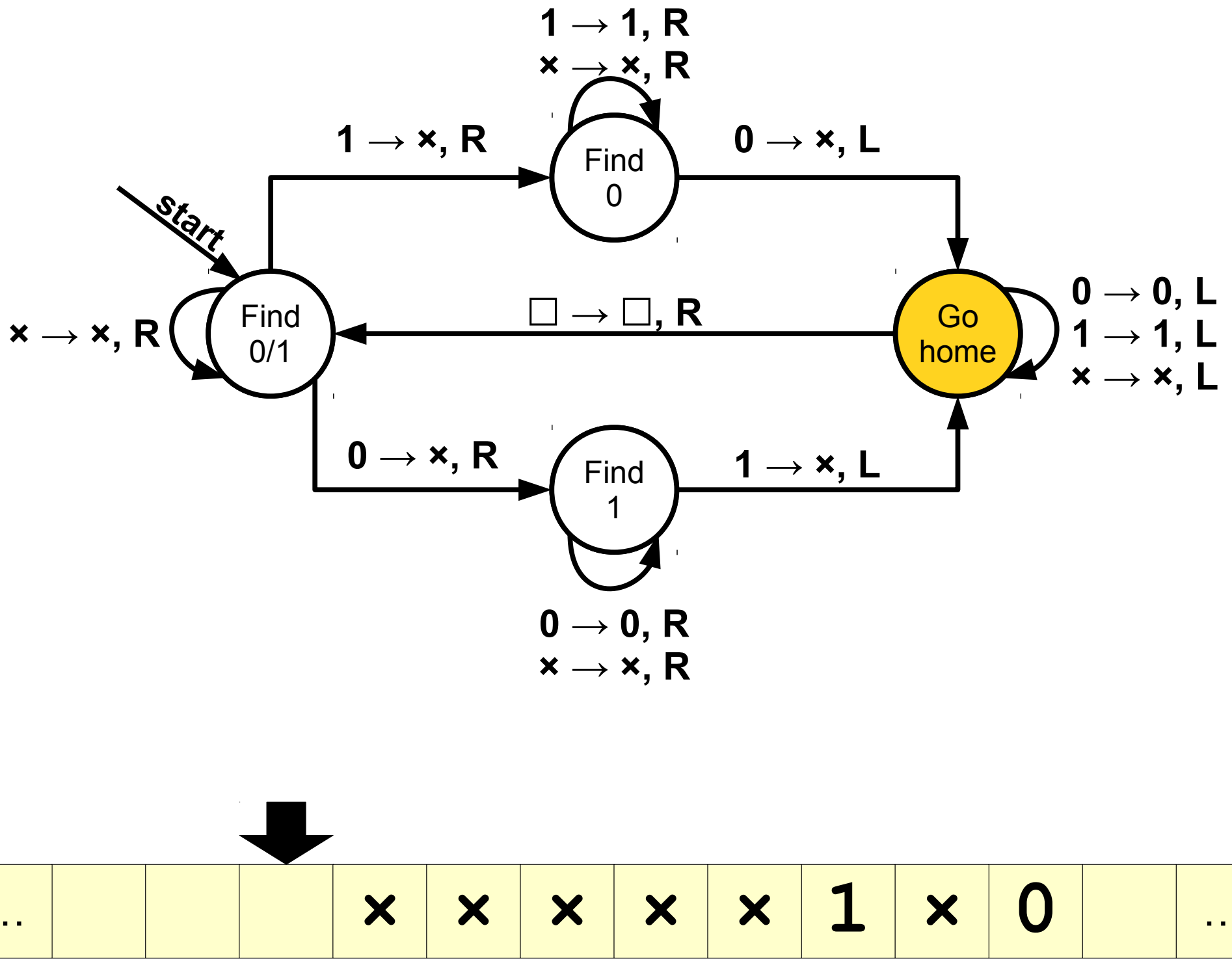

…
×
×
×
×
×
1
×
0
…

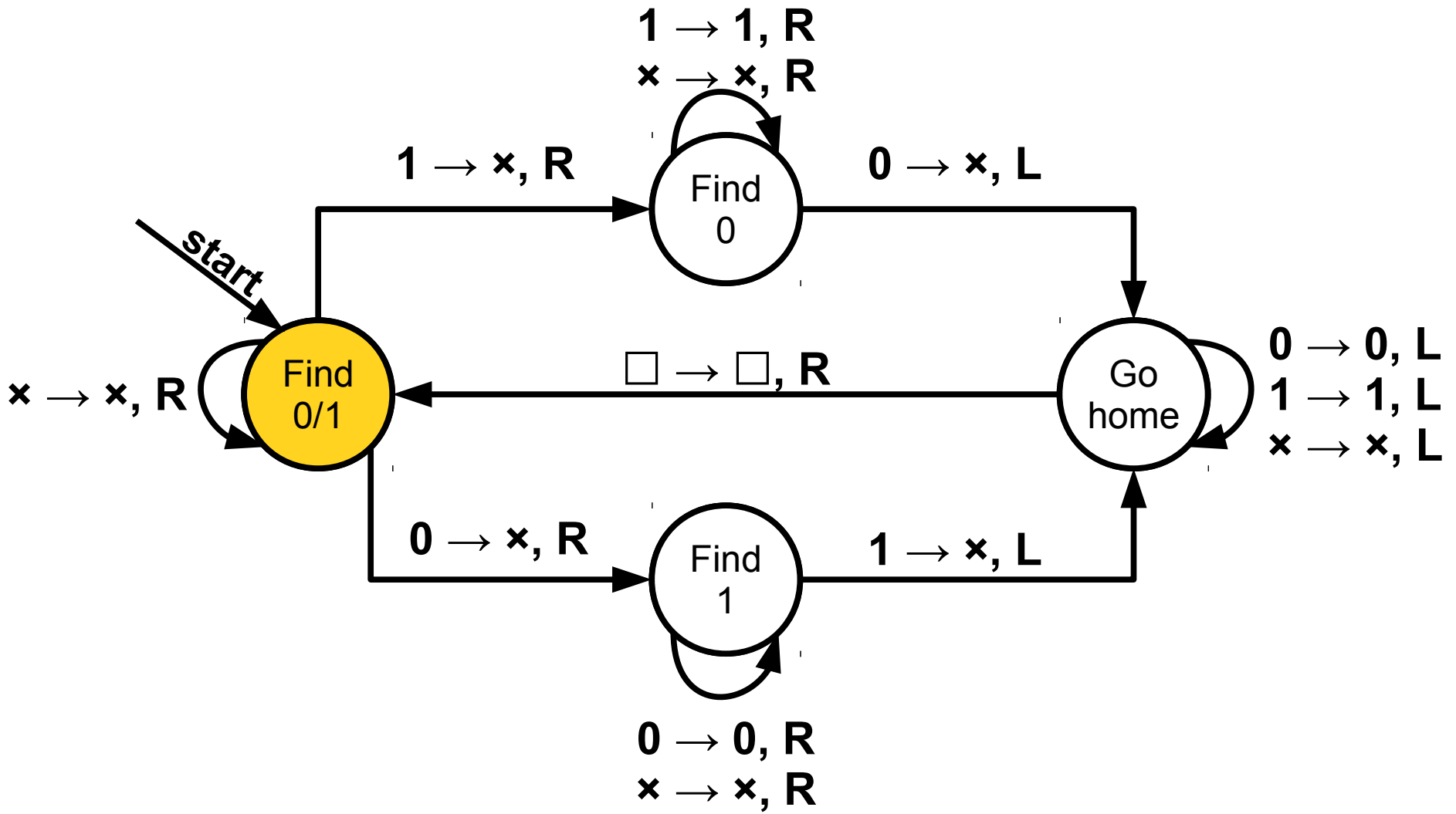

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R
… × × × × × 1 × 0 …

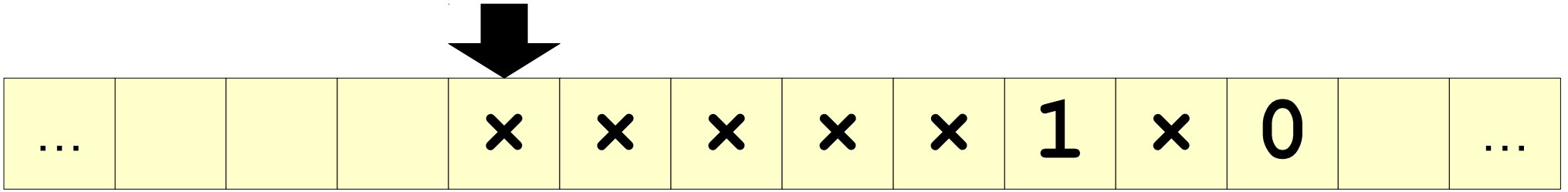

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

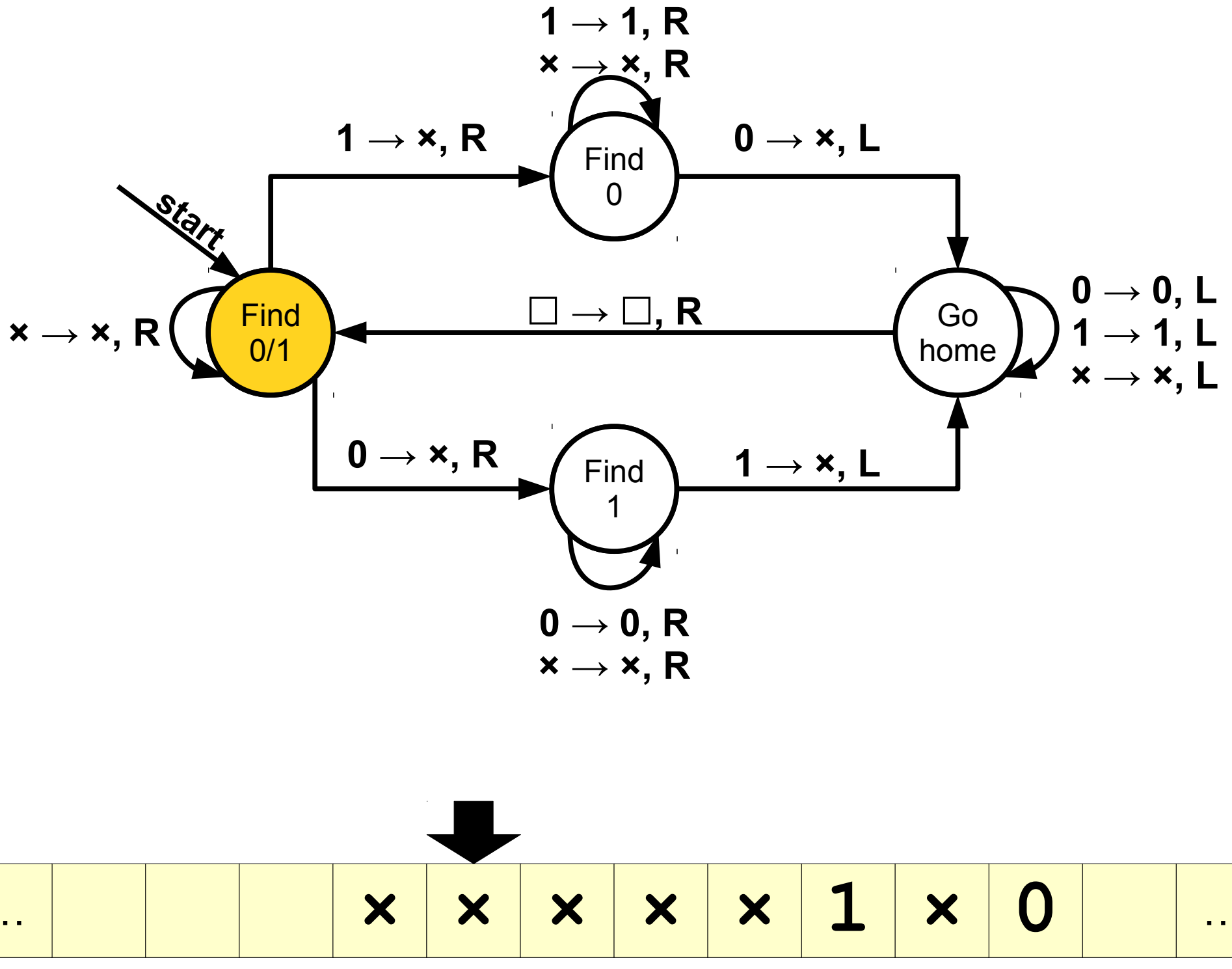

…
×
×
×
×
×
1
×
0
…

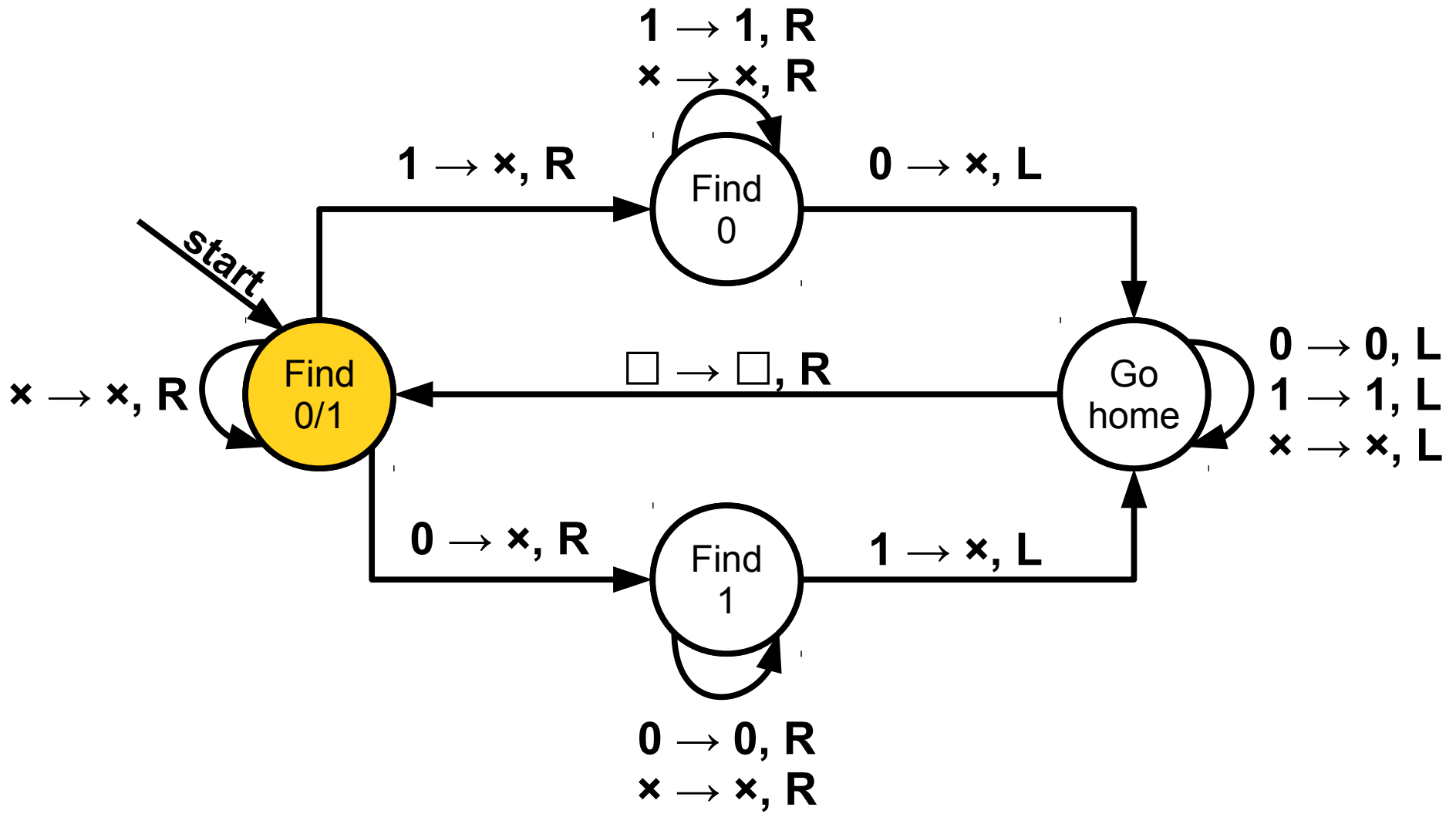

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R
…
×
×
×
×
×
1
×
0
…

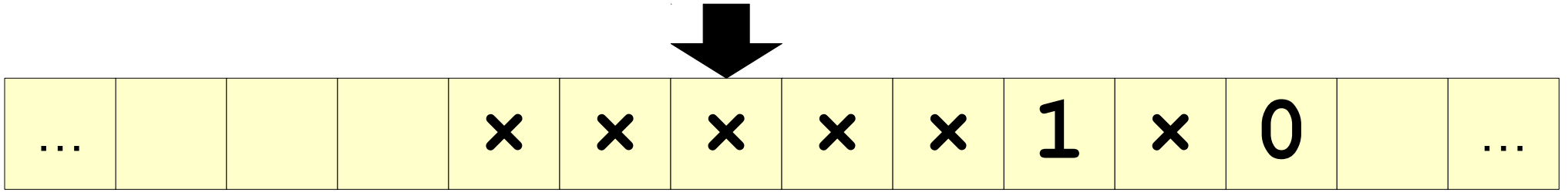

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R
…
×
×
×
×
×
1
×
0
…

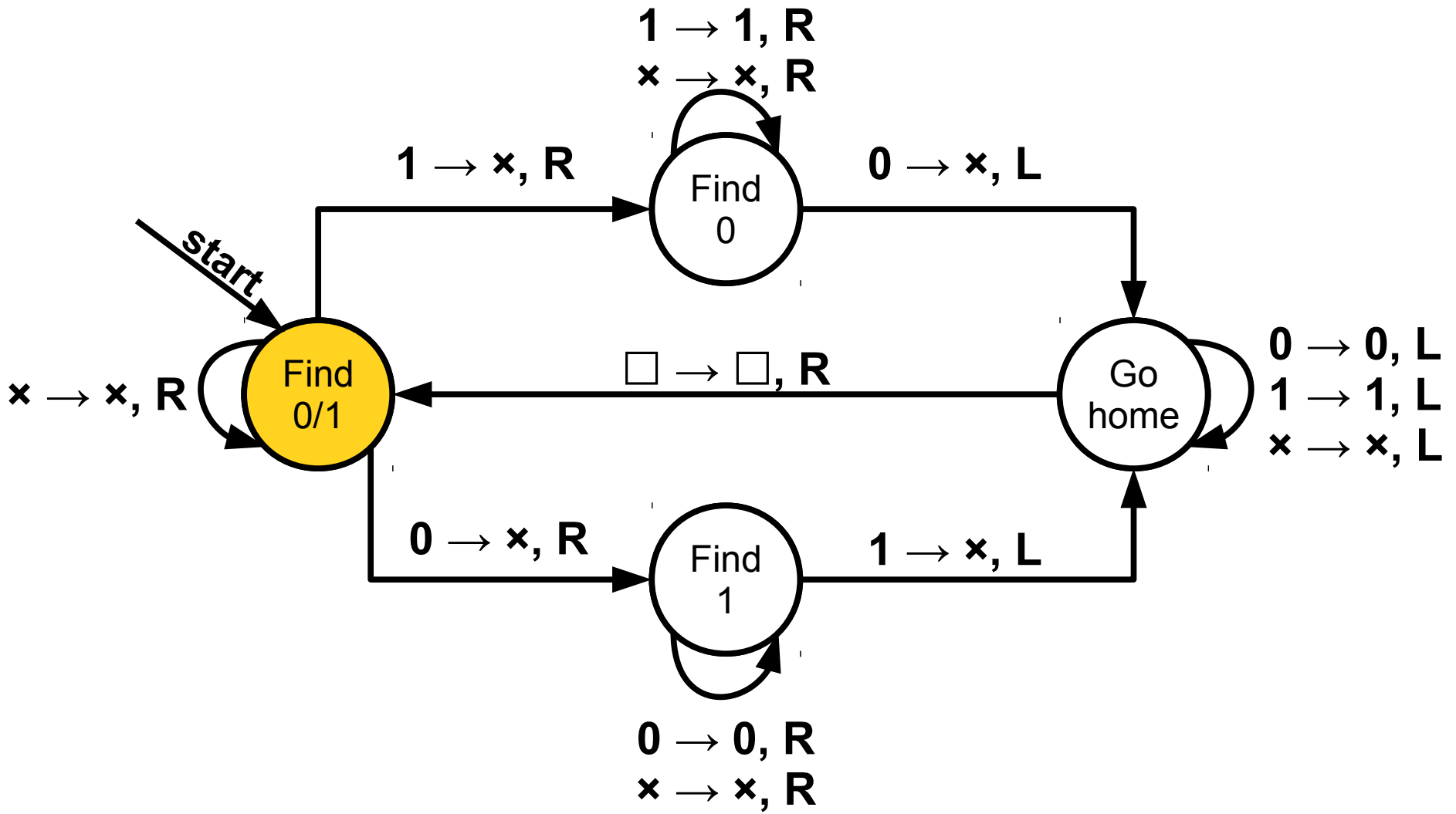

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find 1
1 → ×, L
0 → 0, R
× → ×, R

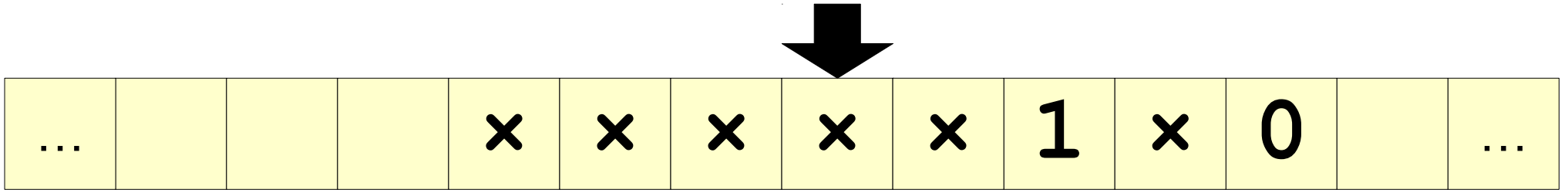

…
×
×
×
×
×
1
×
0
…

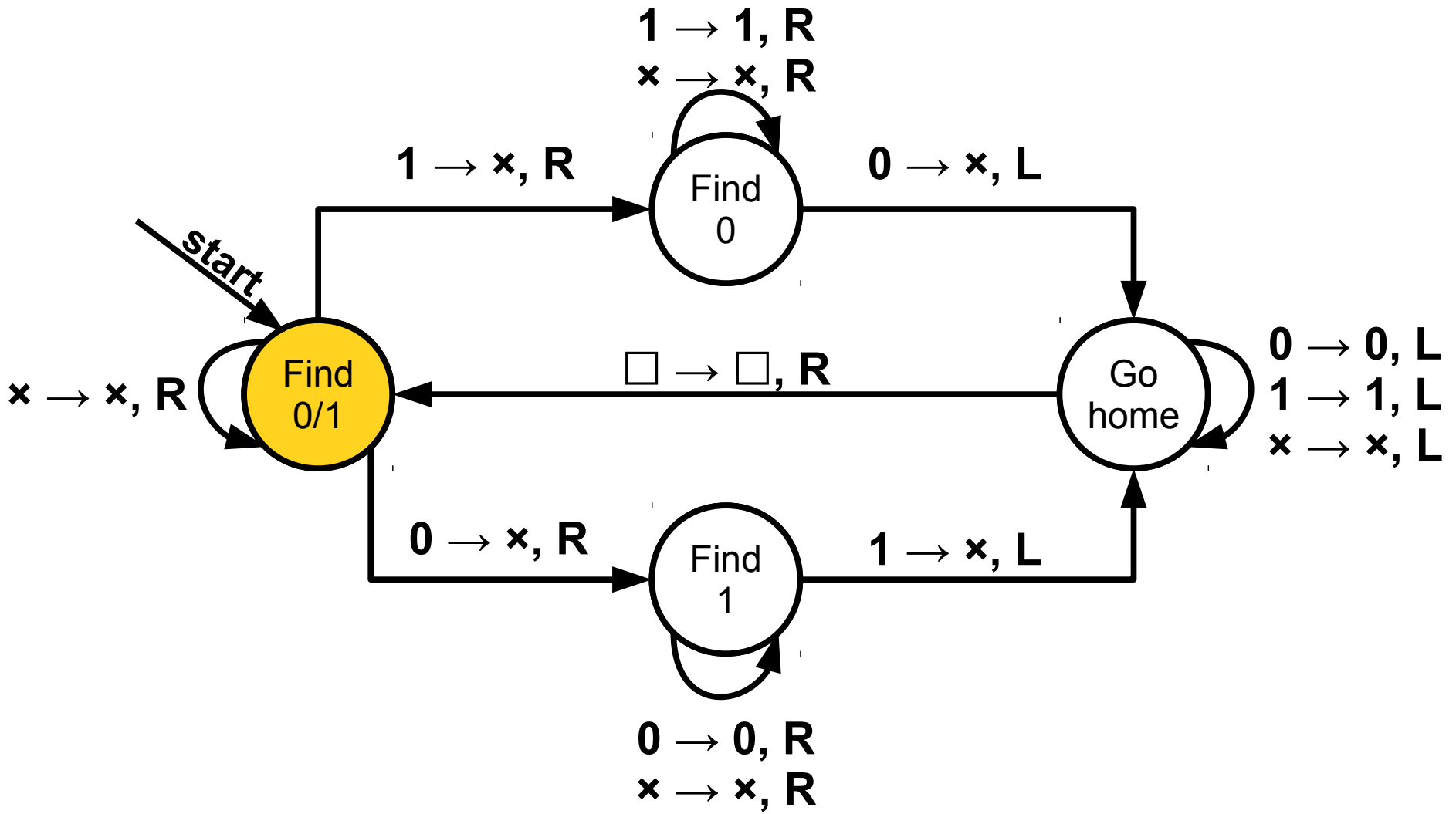

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R
… × × × × × 1 × 0 …

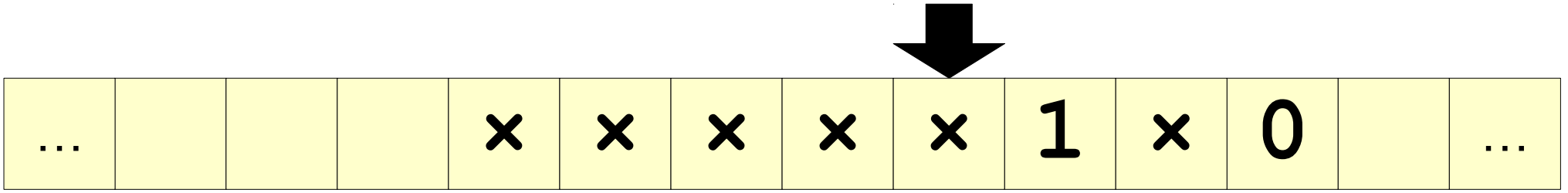

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

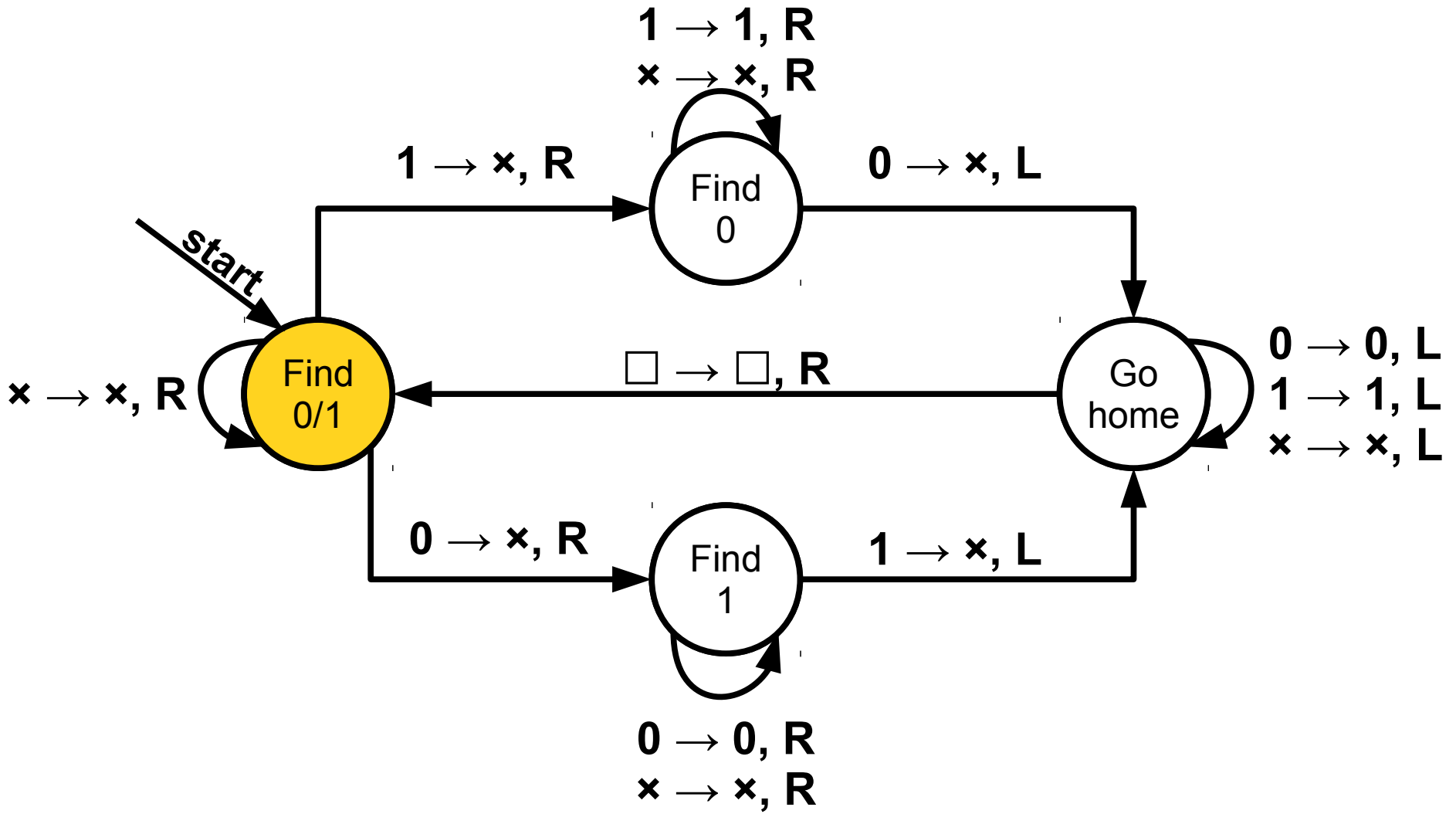

…
× × × × × 1 × 0
…

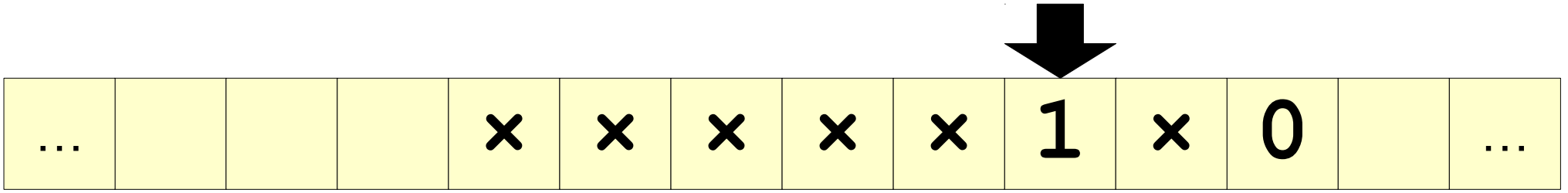

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

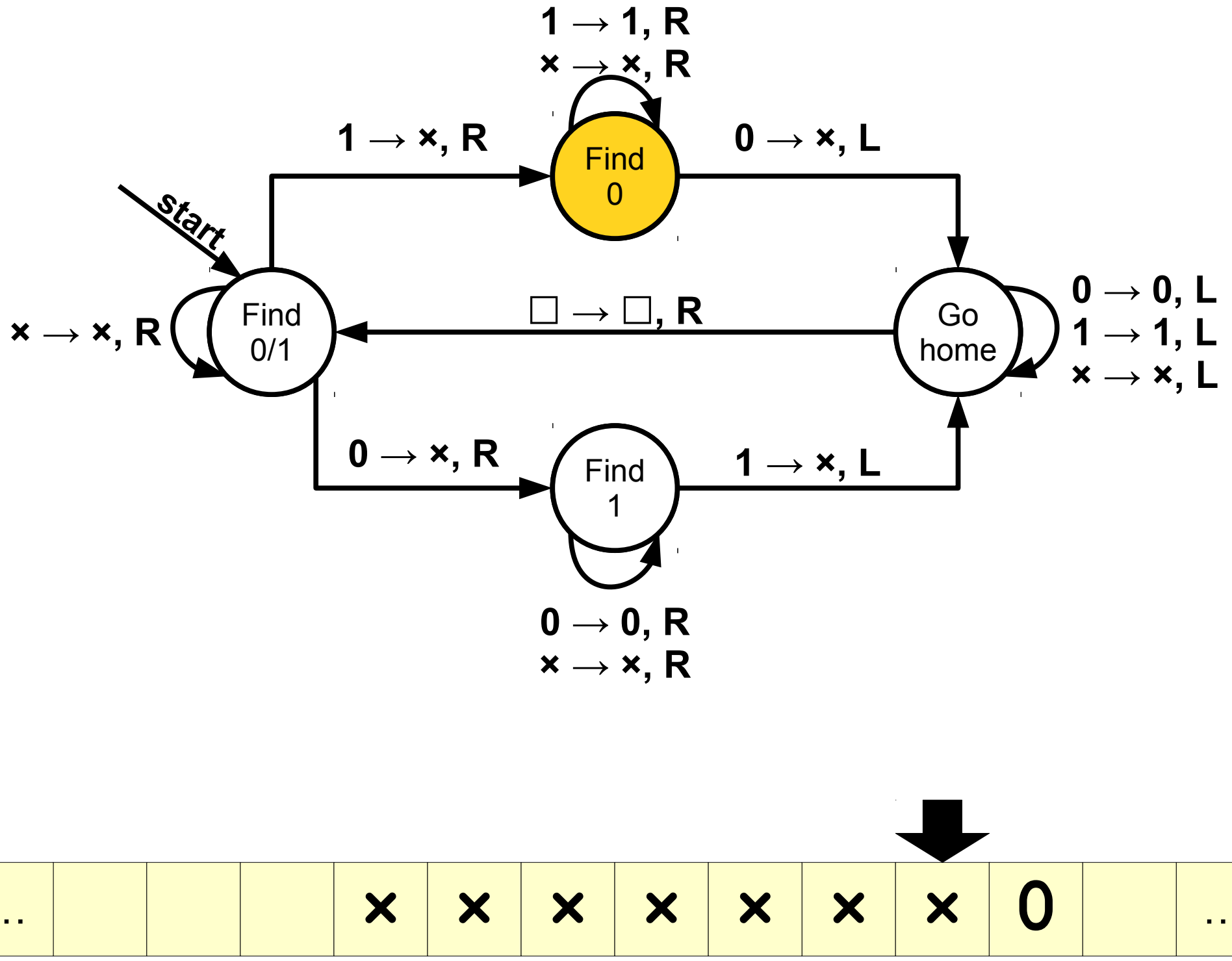

…
×
×
×
×
×
1
×
0
…

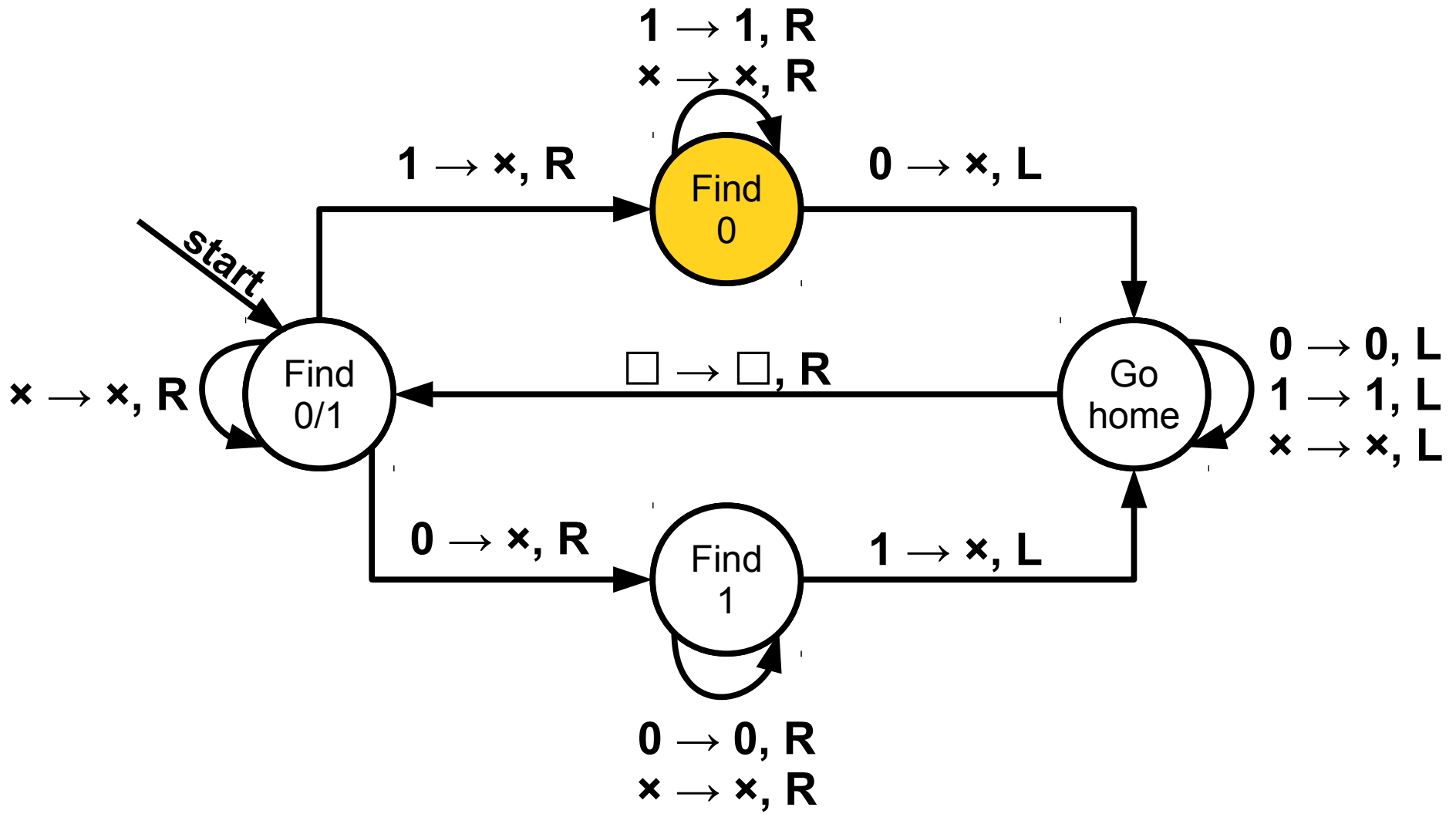

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

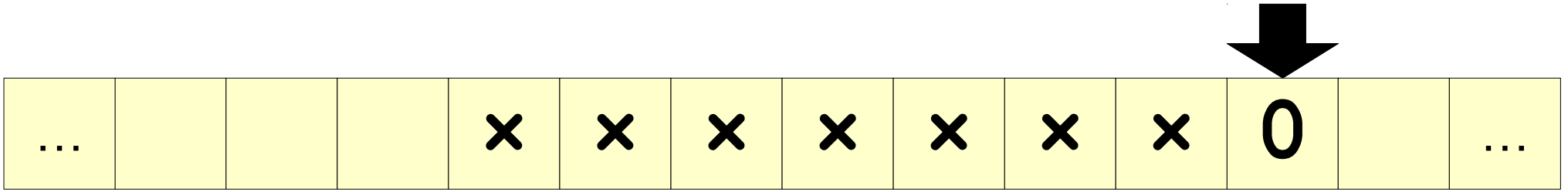

…
×
×
×
×
×
1
×
0
…

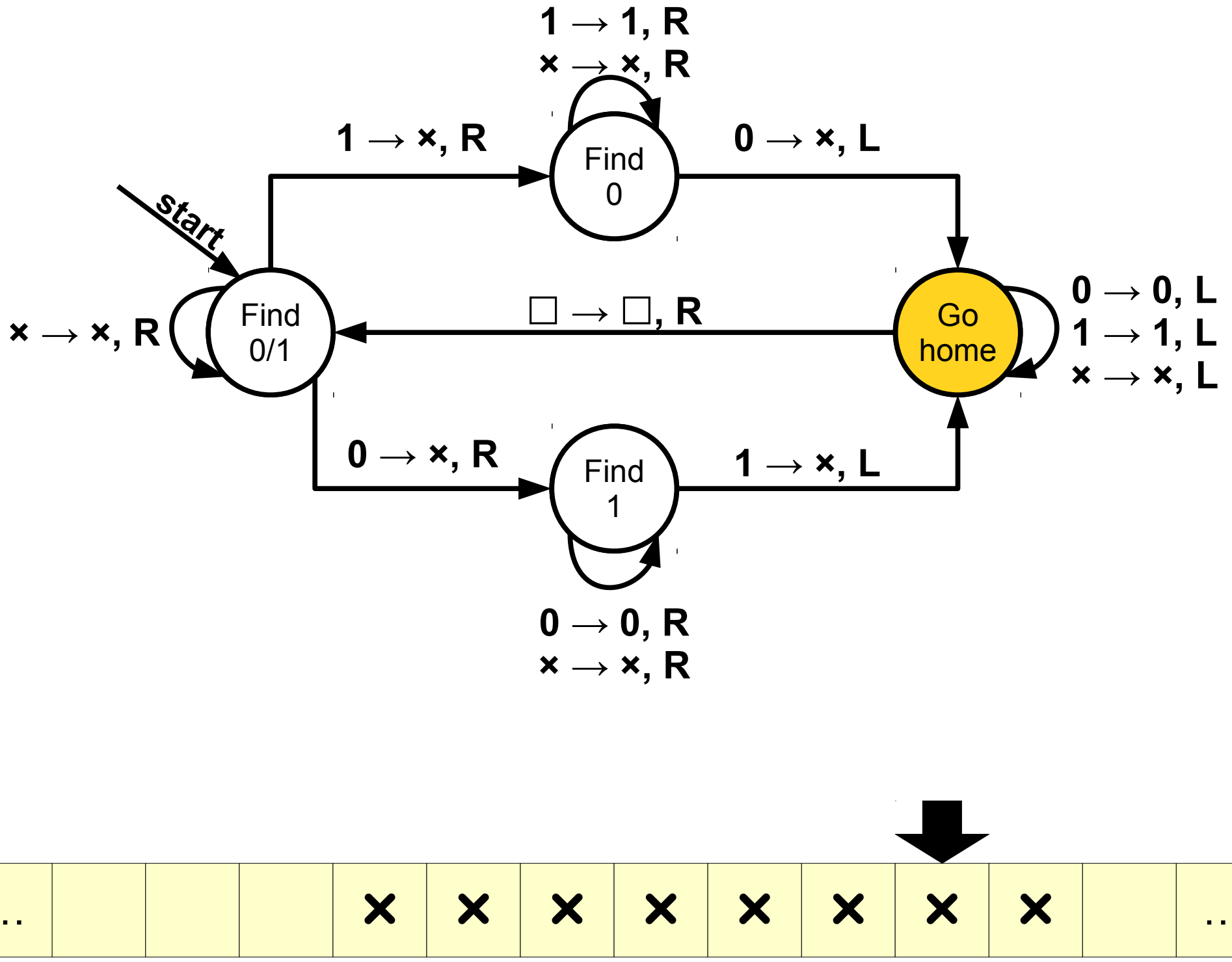

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

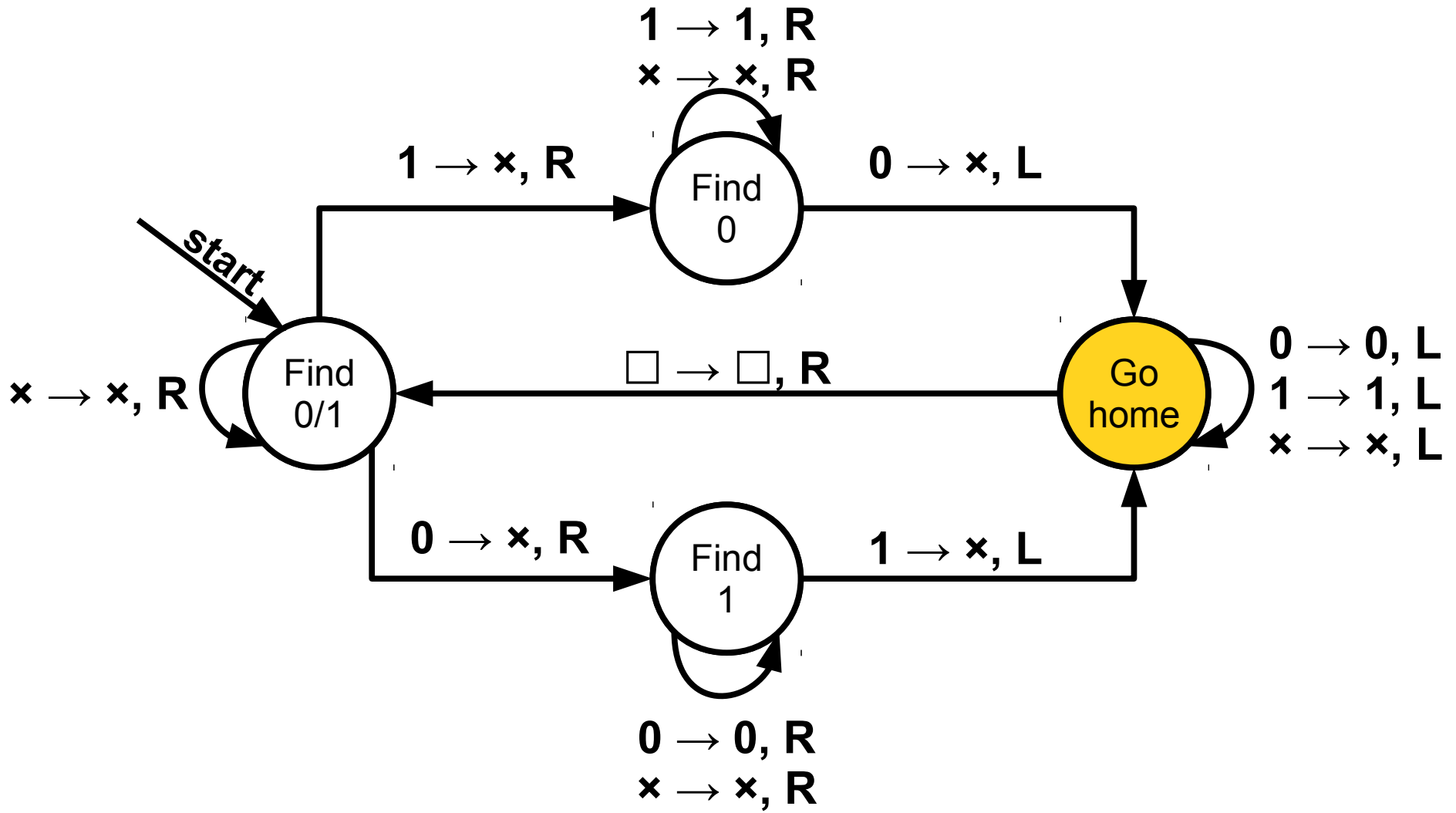

…
×
×
×
×
×
1
×
0
…

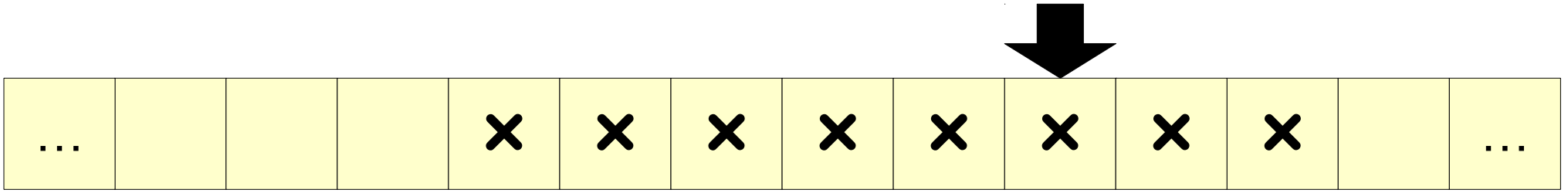

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R
…
×
×
×
×
×
×
×
0
…

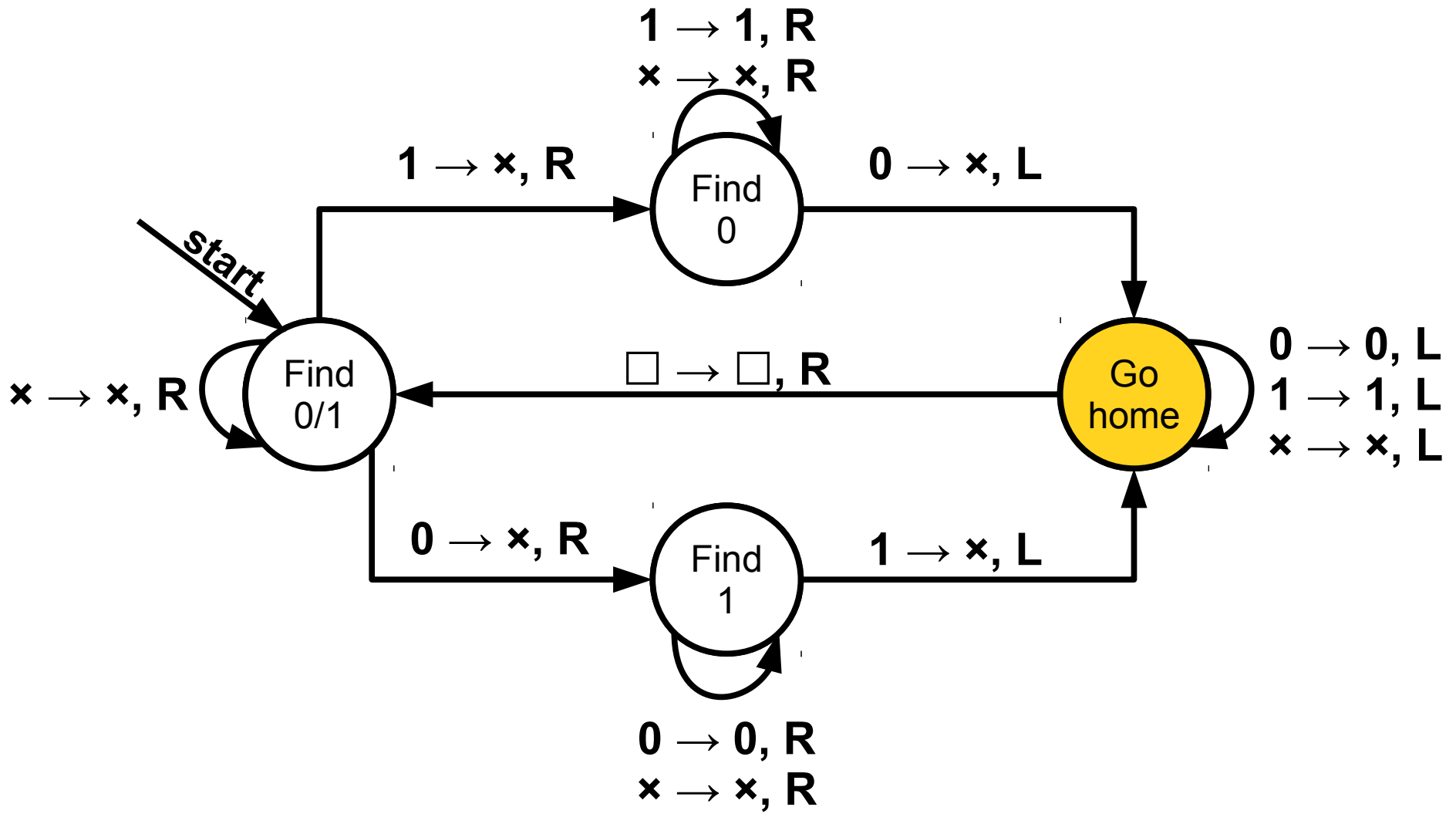

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

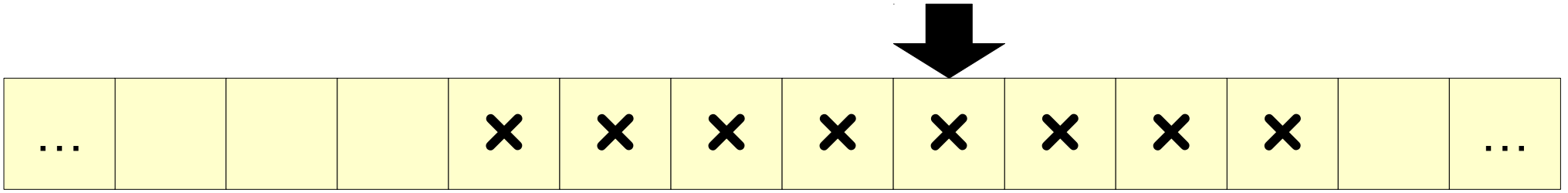

…
×
×
×
×
×
×
×
0
…

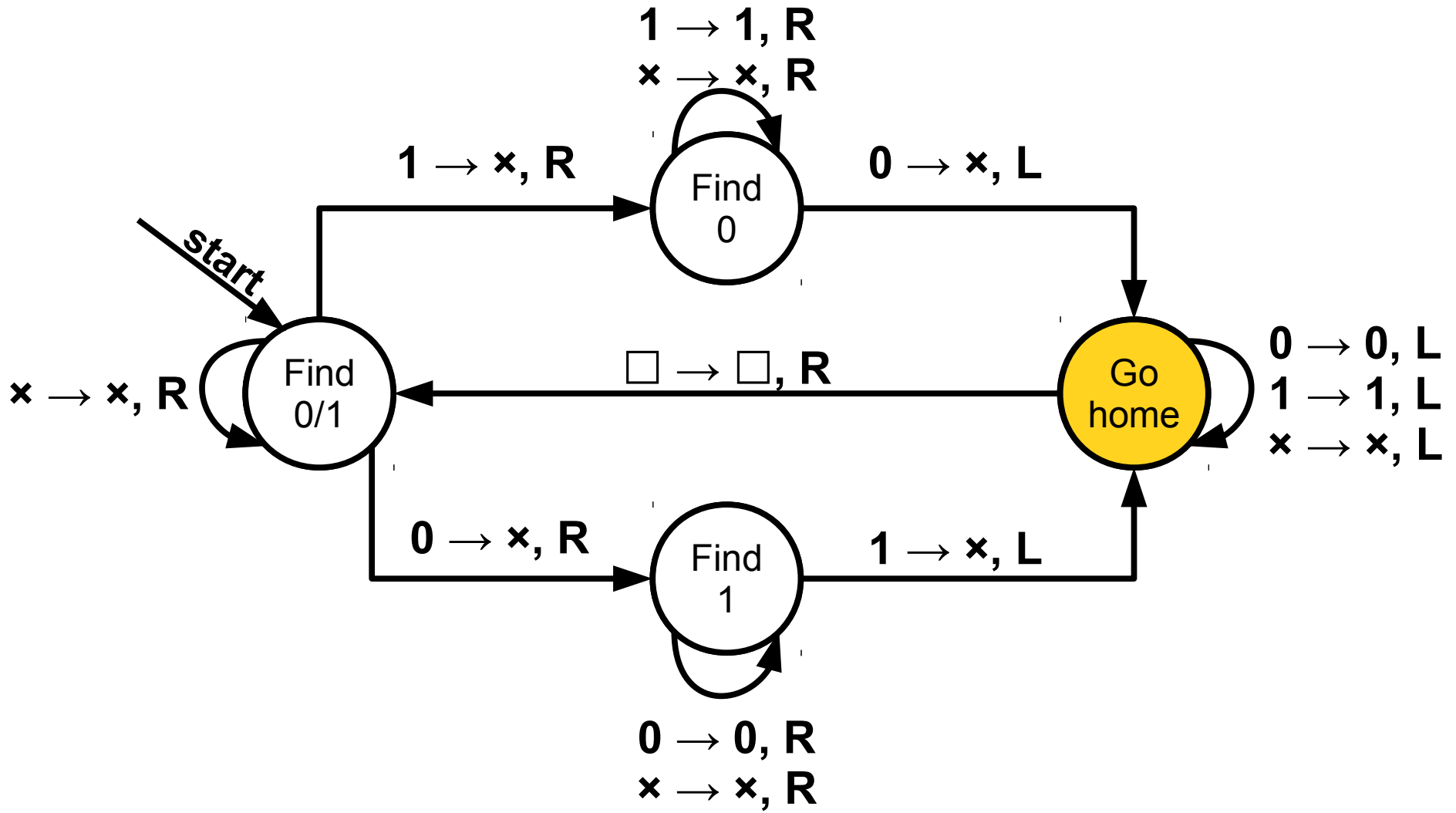

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R
…
…

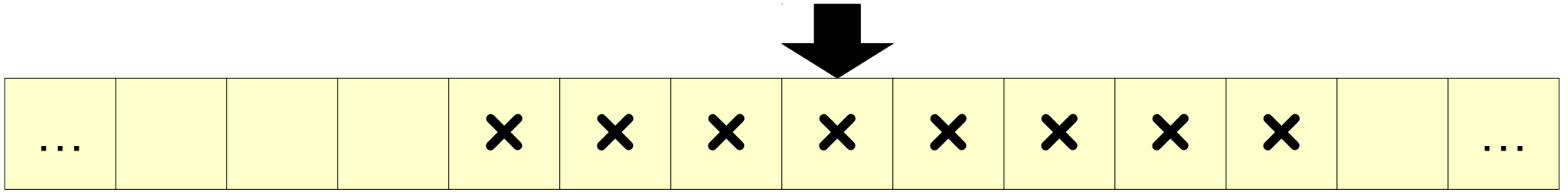

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

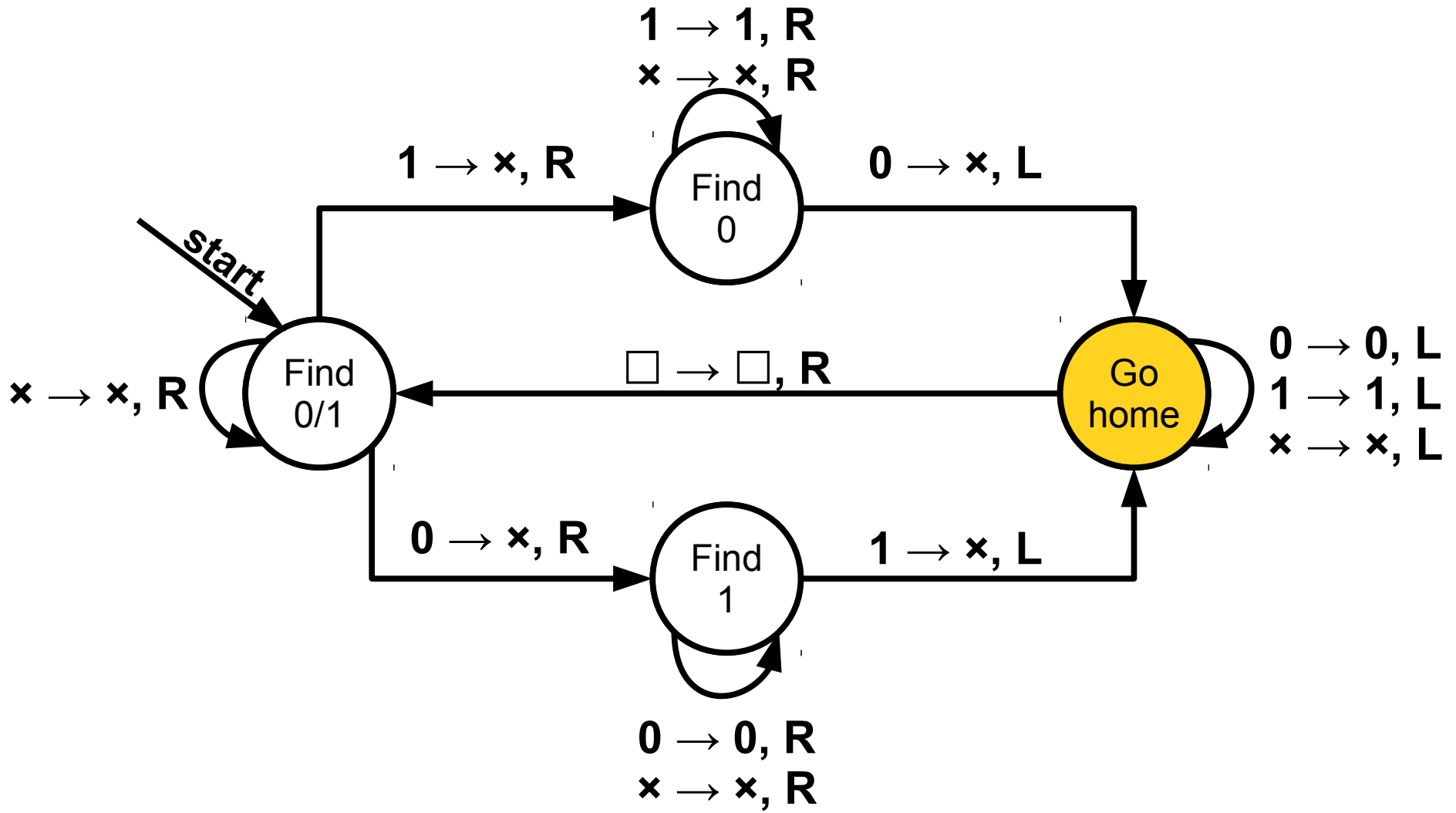

…
×
×
×
×
×
×
×
×
…

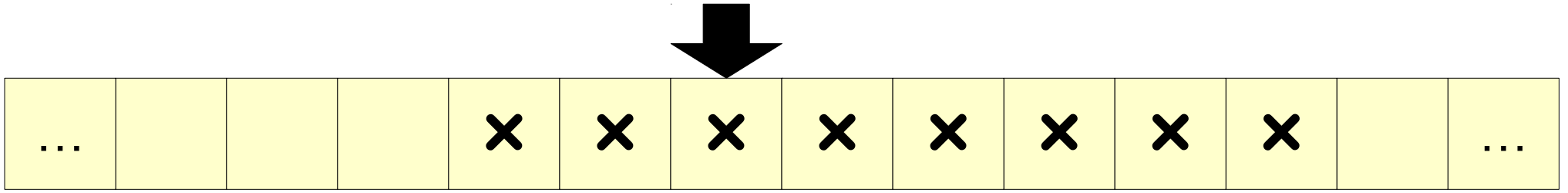

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

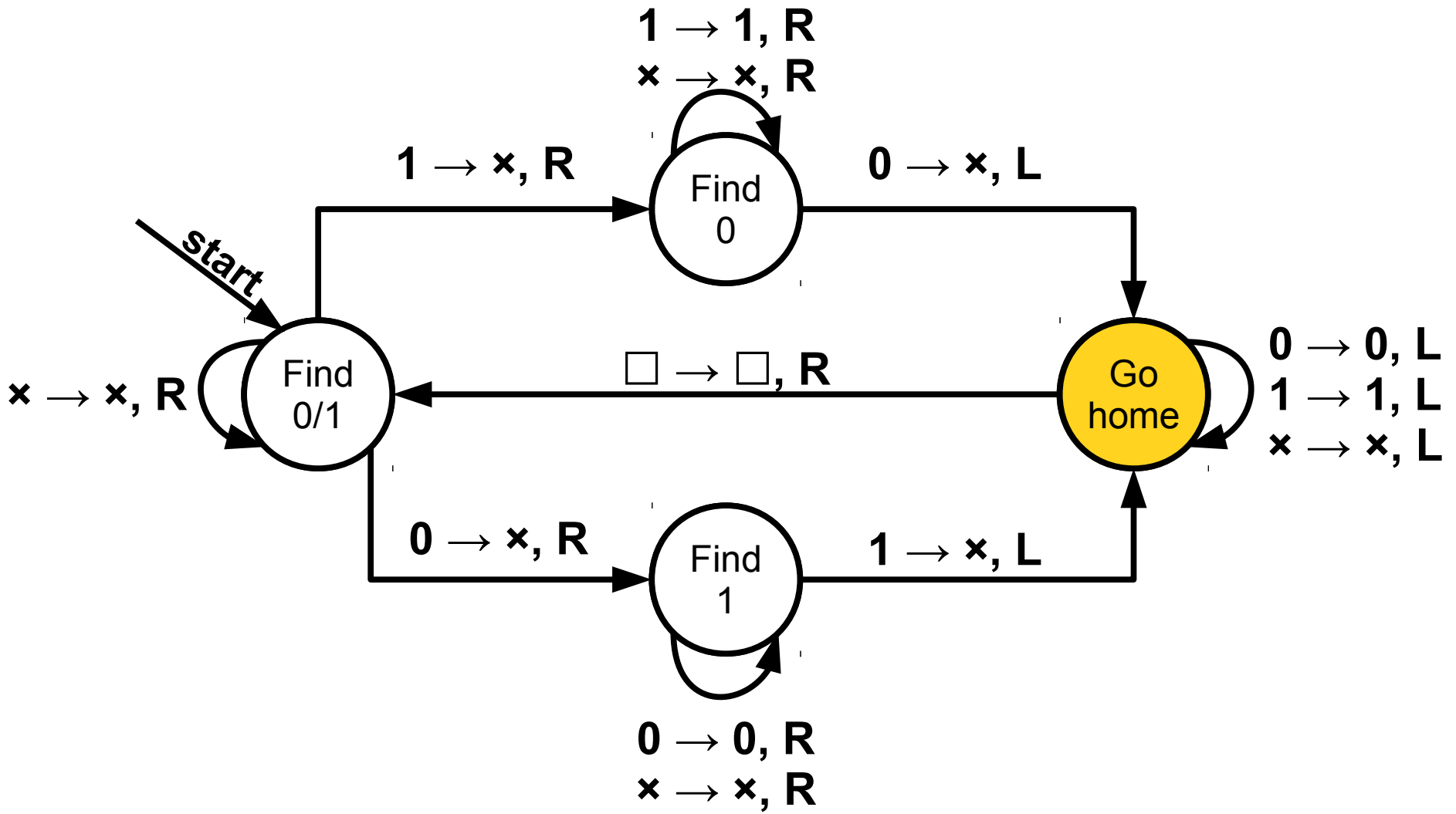

…
×
×
×
×
×
×
×
×
…

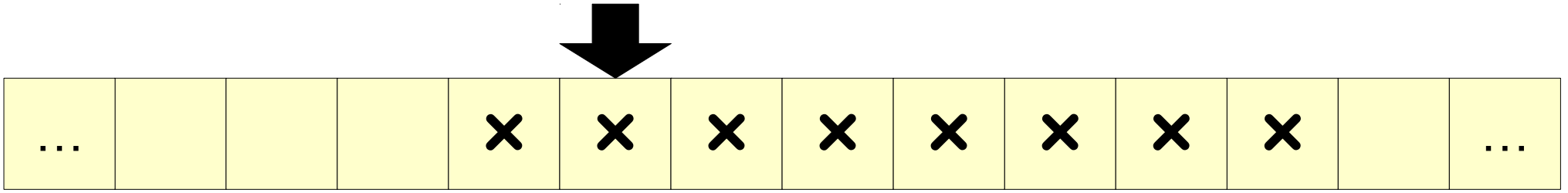

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

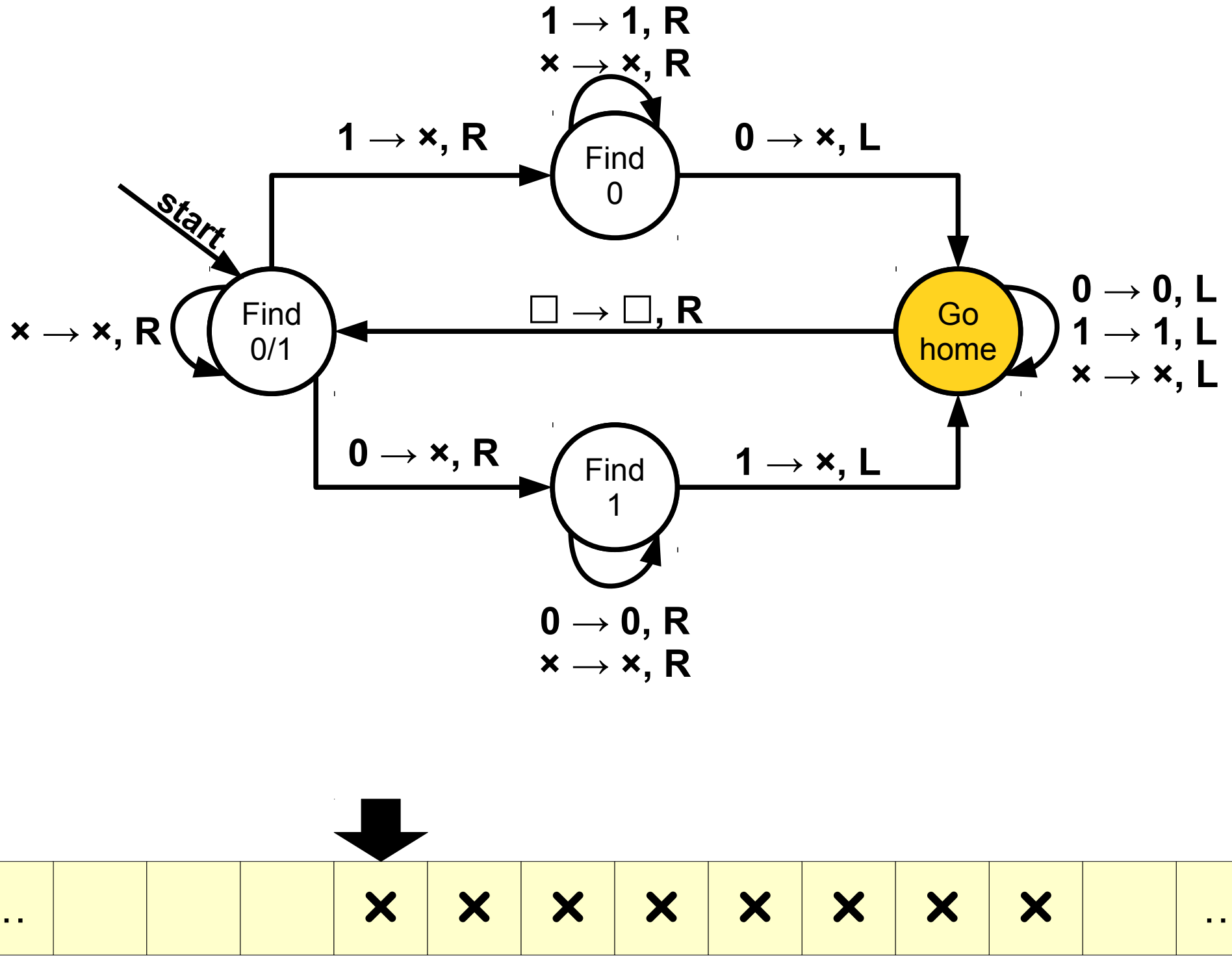

…
×
×
×
×
×
×
×
×
…

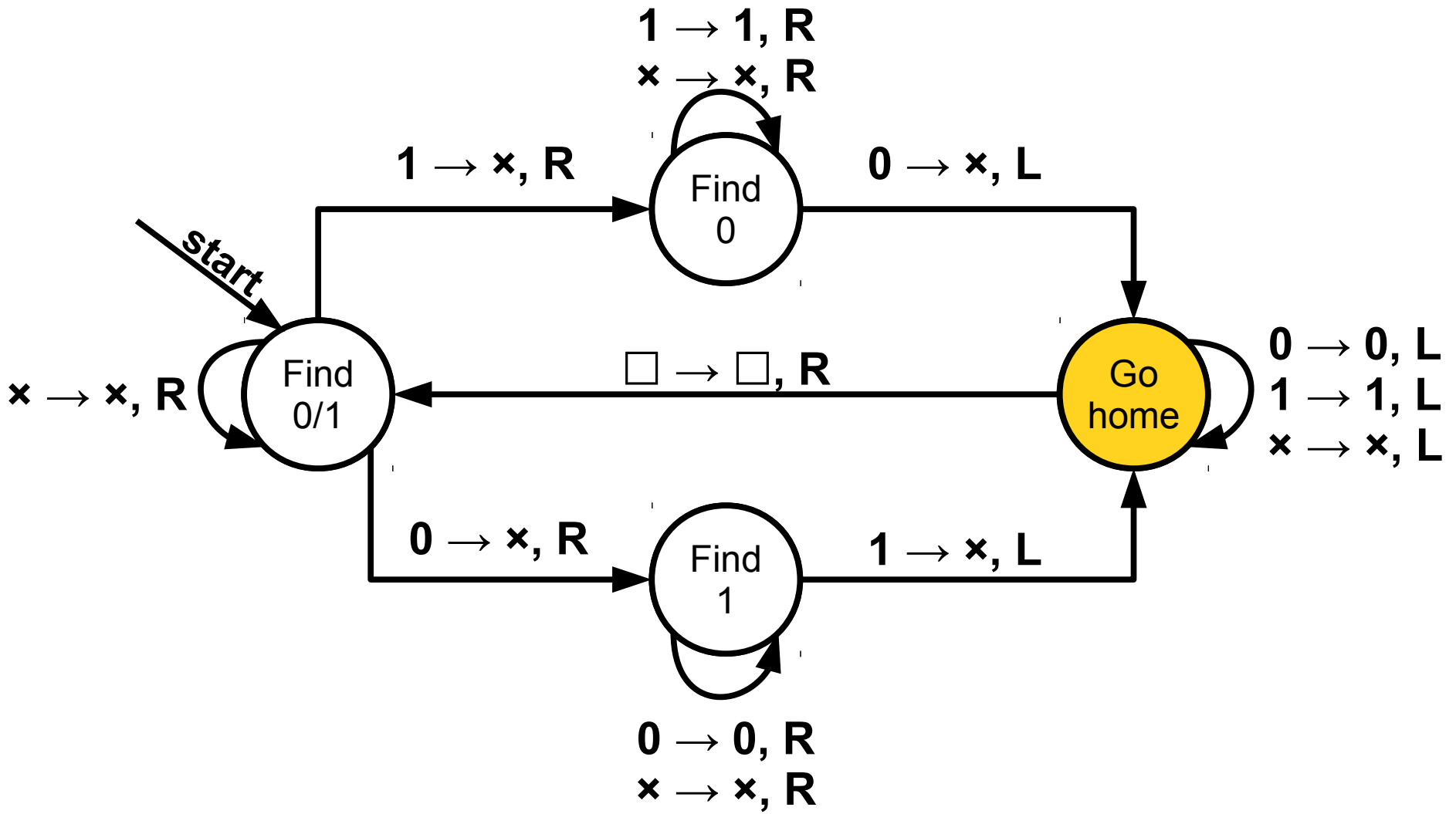

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

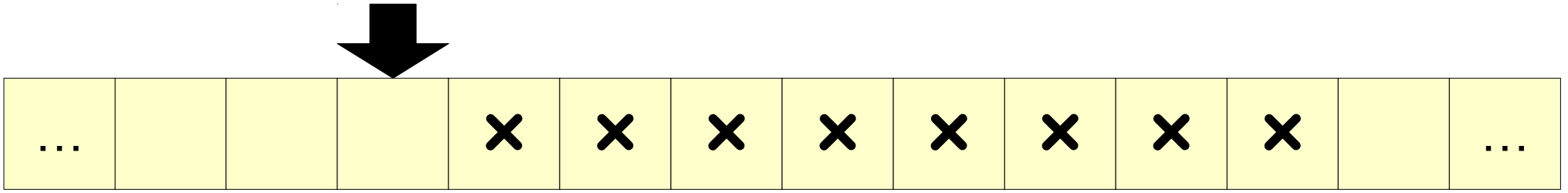

…
×
×
×
×
×
×
×
×
…

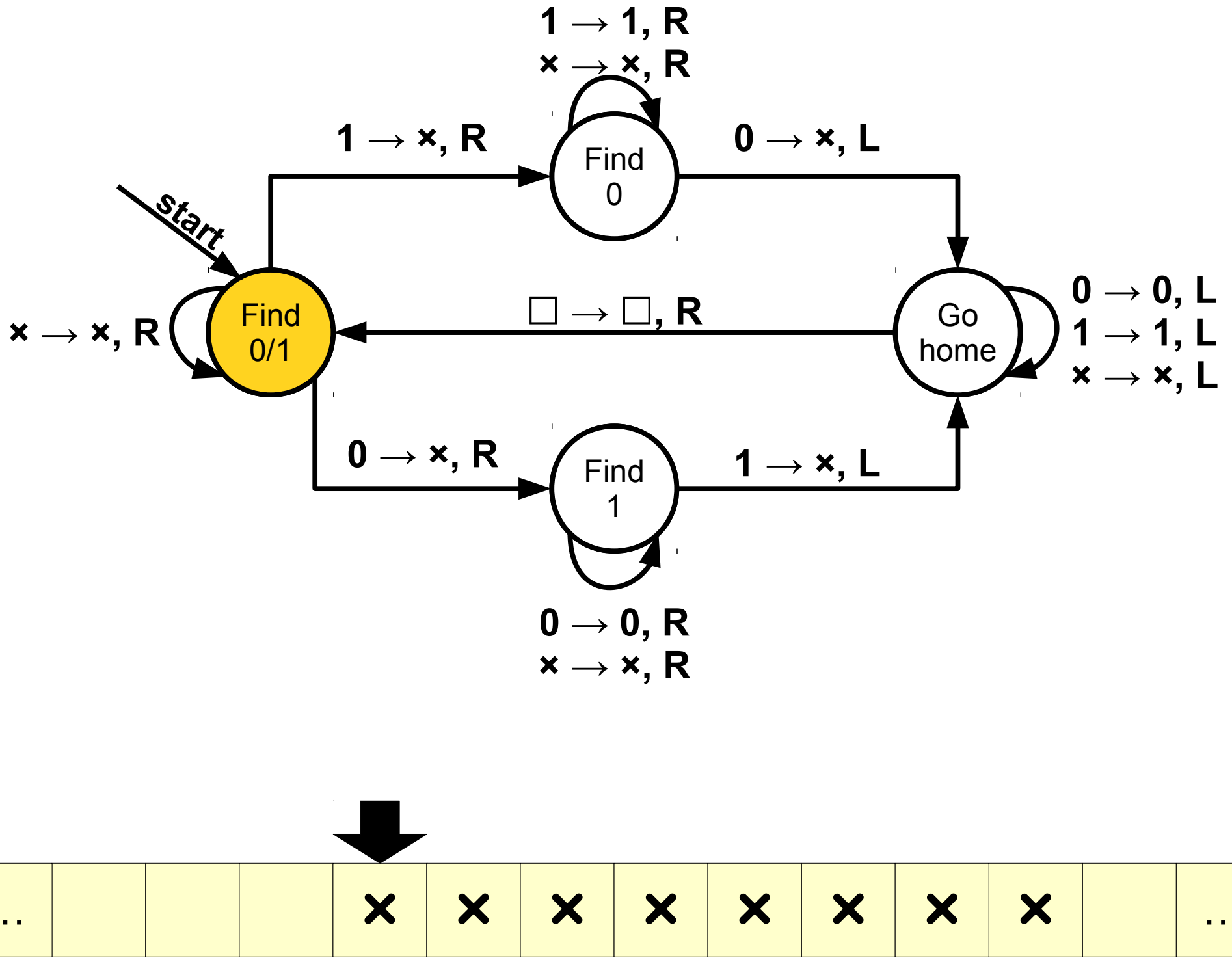

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

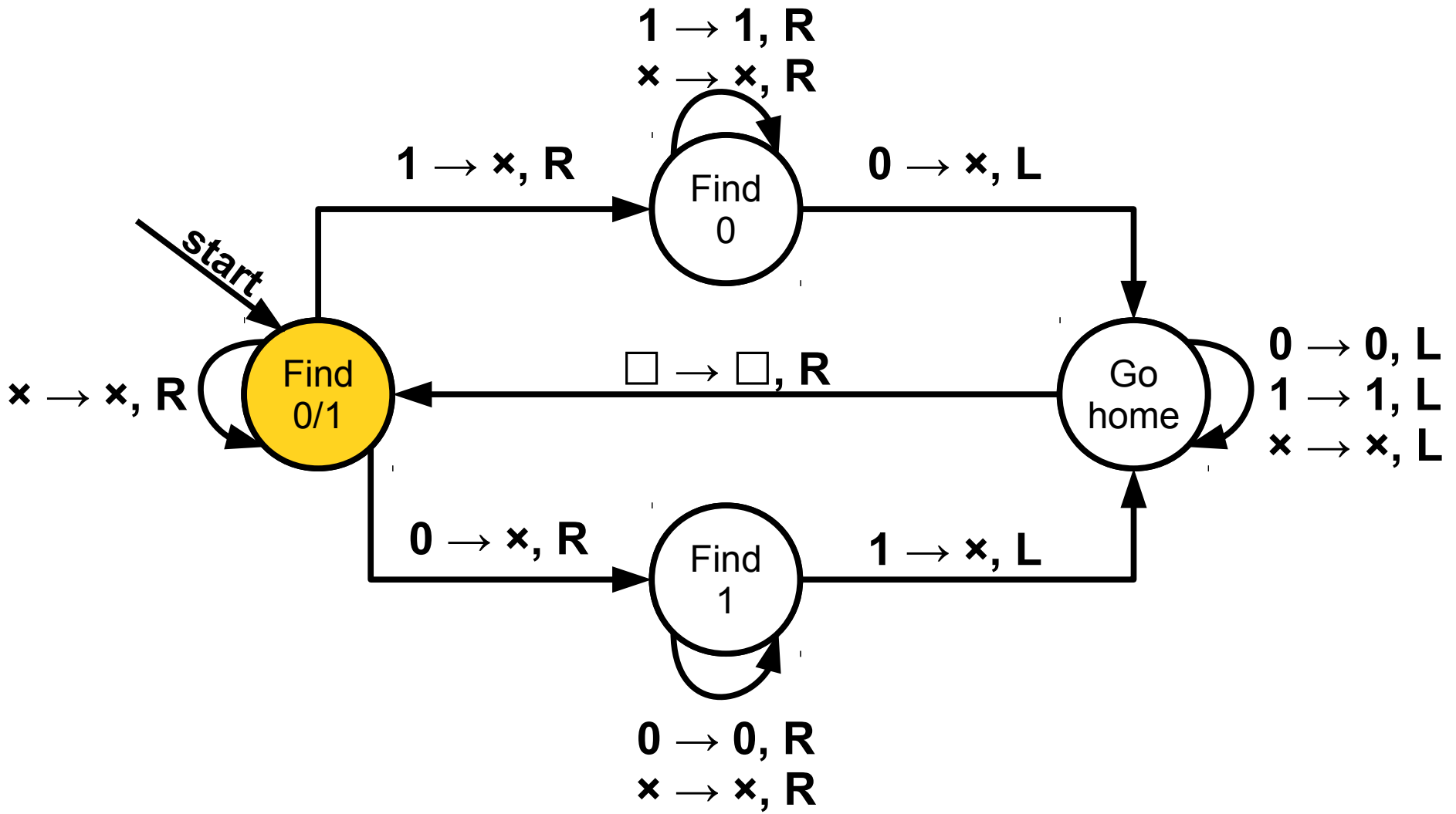

…
×
×
×
×
×
×
×
×
…

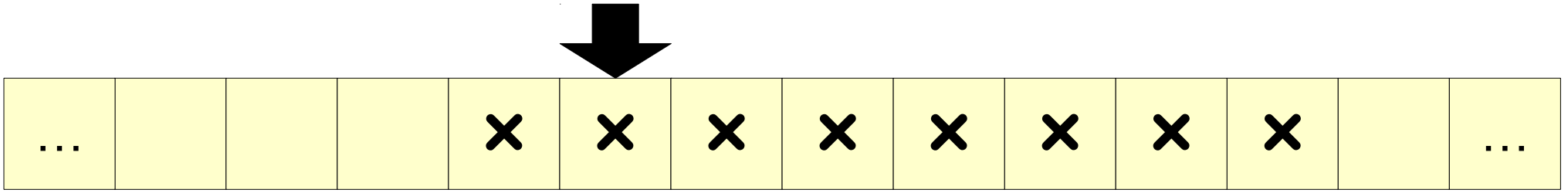

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find 1
1 → ×, L
0 → 0, R
× → ×, R
…
×
×
×
×
×
×
×
×
…

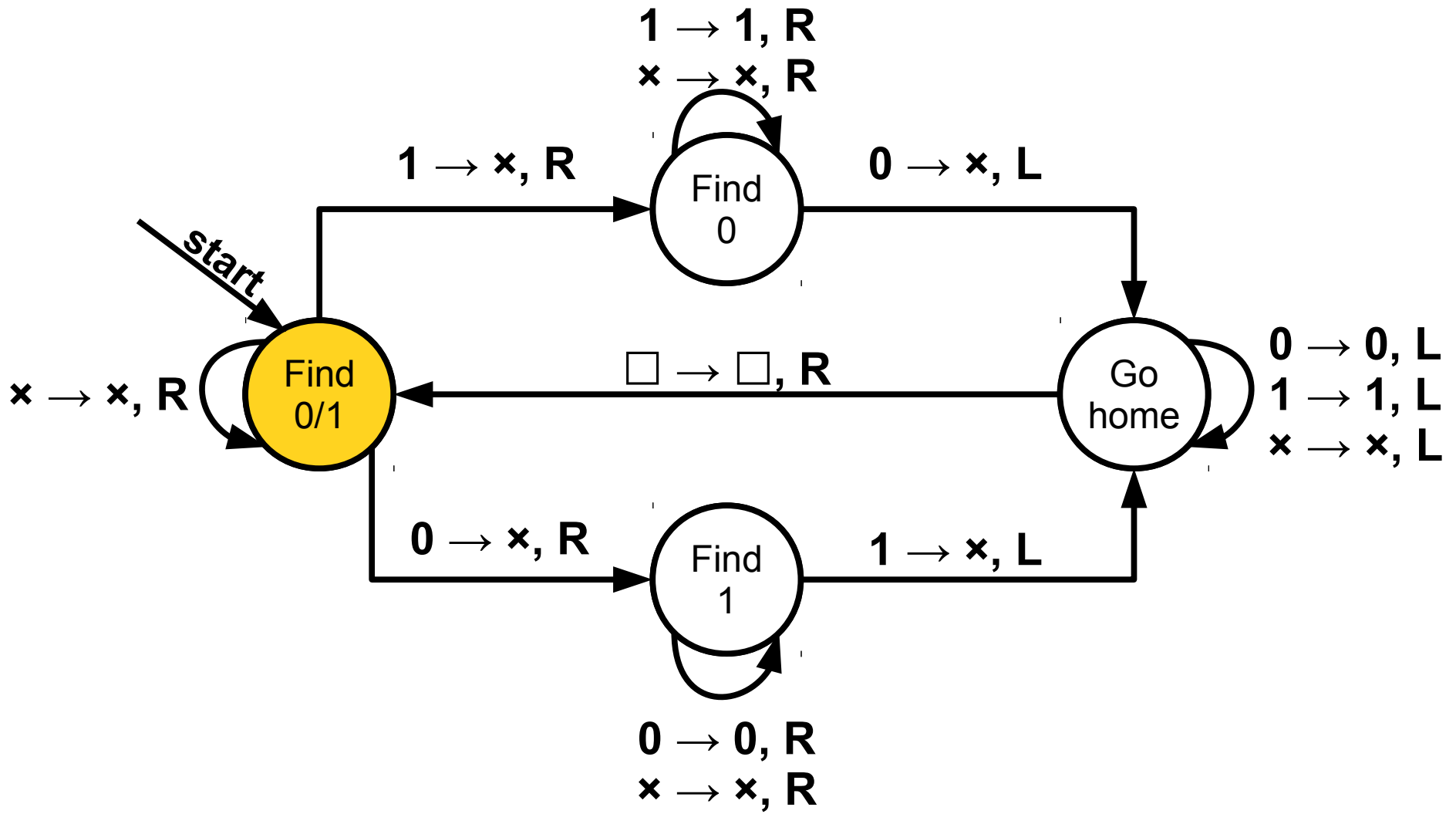

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R

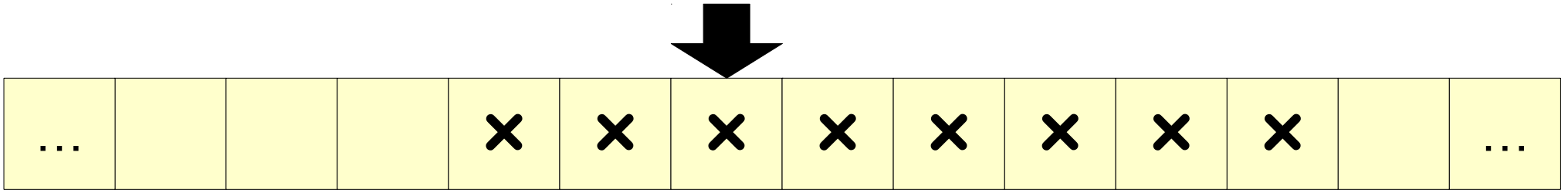

…
×
×
×
×
×
×
×
×
…

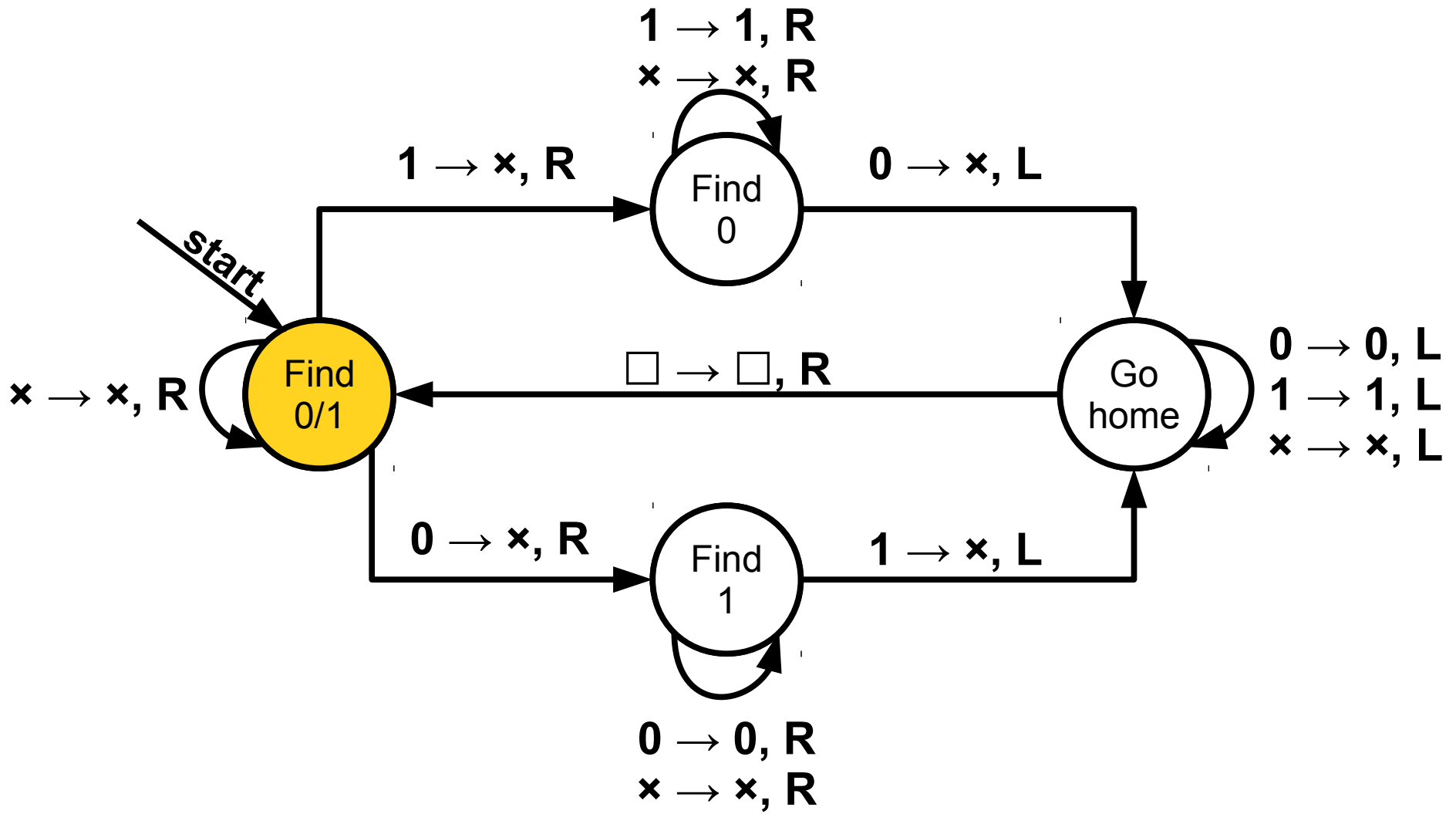

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R
…
×
×
×
×
×
×
×
×
…

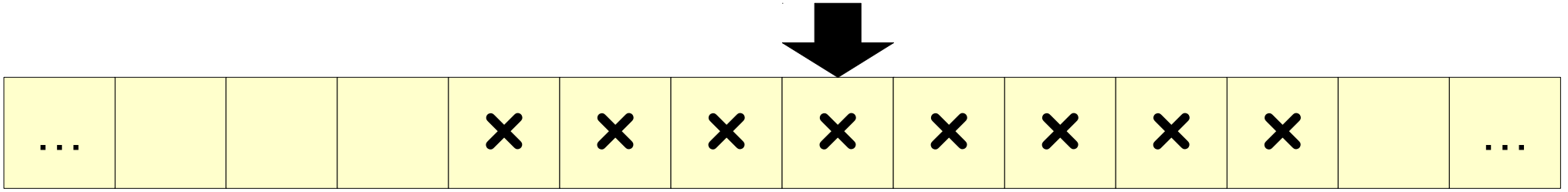

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R

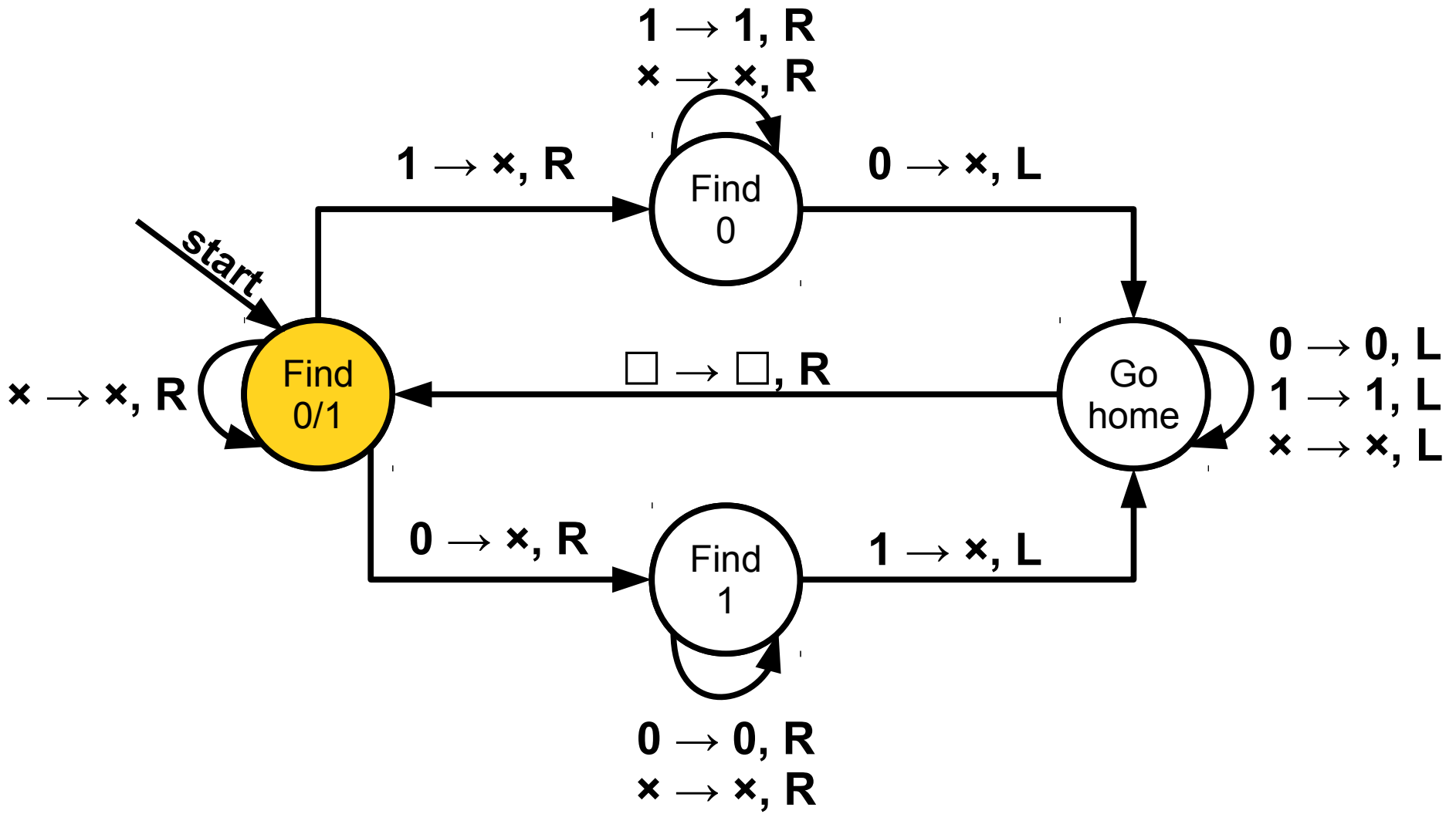

…
×
×
×
×
×
×
×
×
…

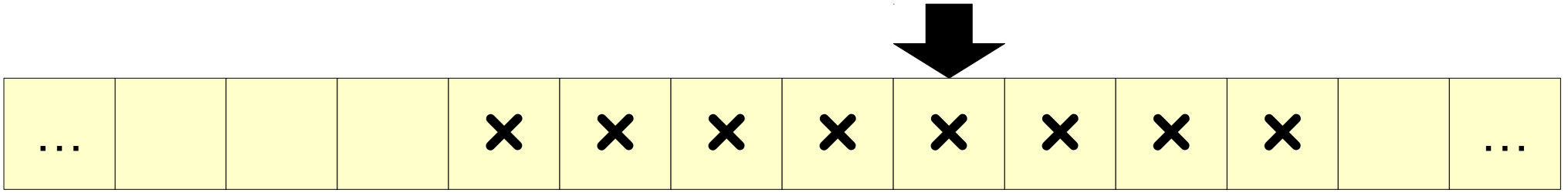

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R

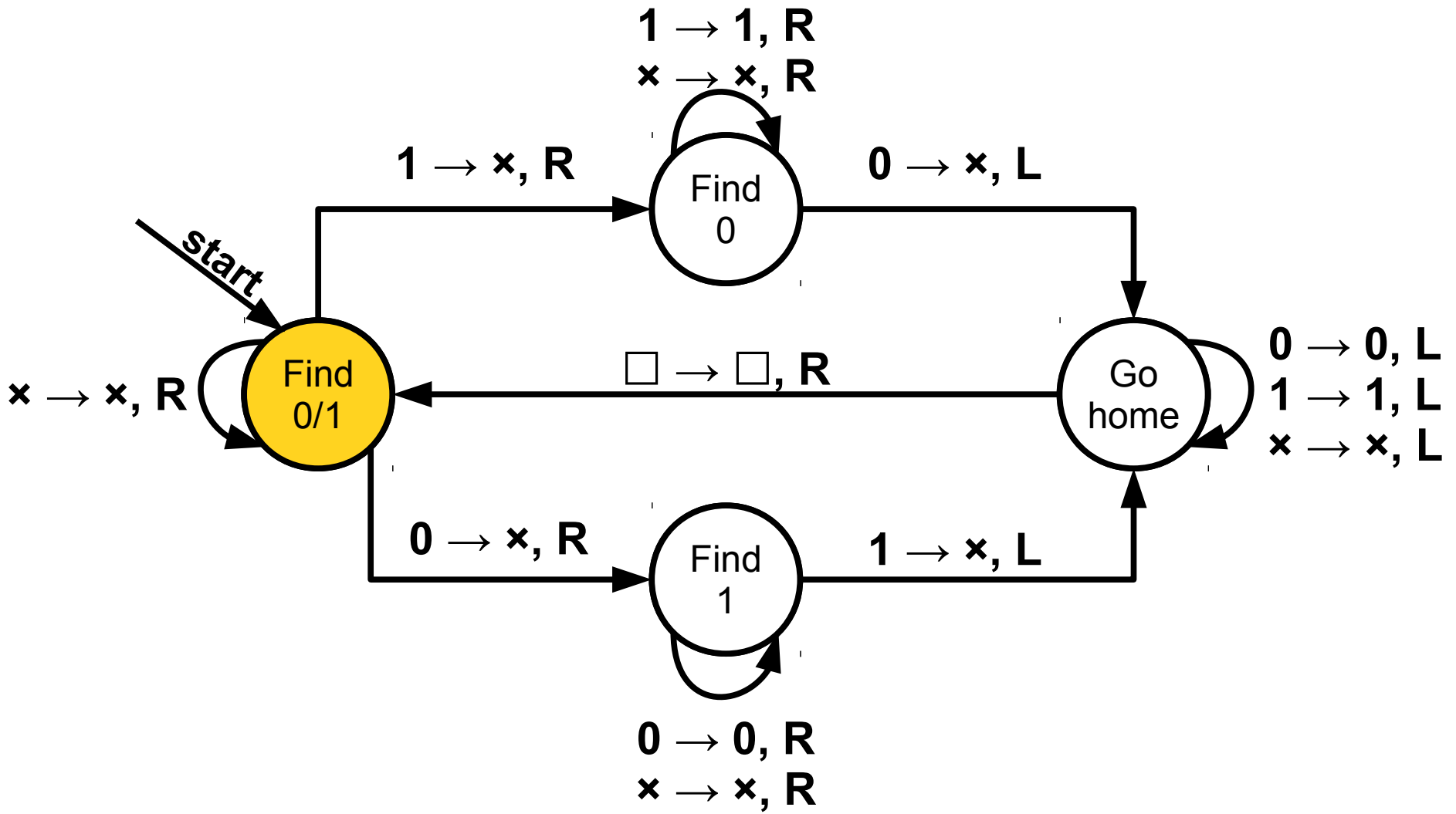

…
×
×
×
×
×
×
×
×
…

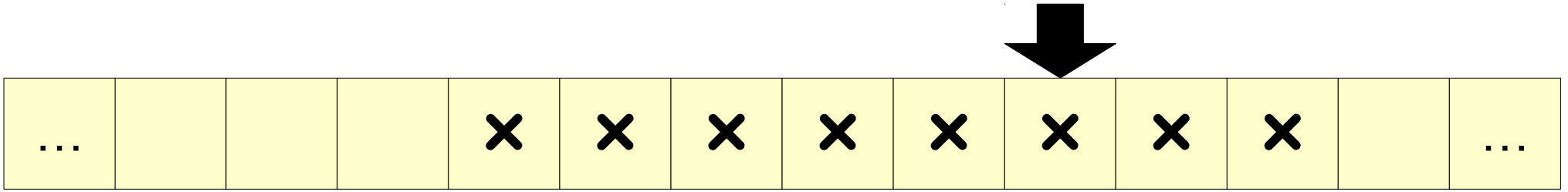

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

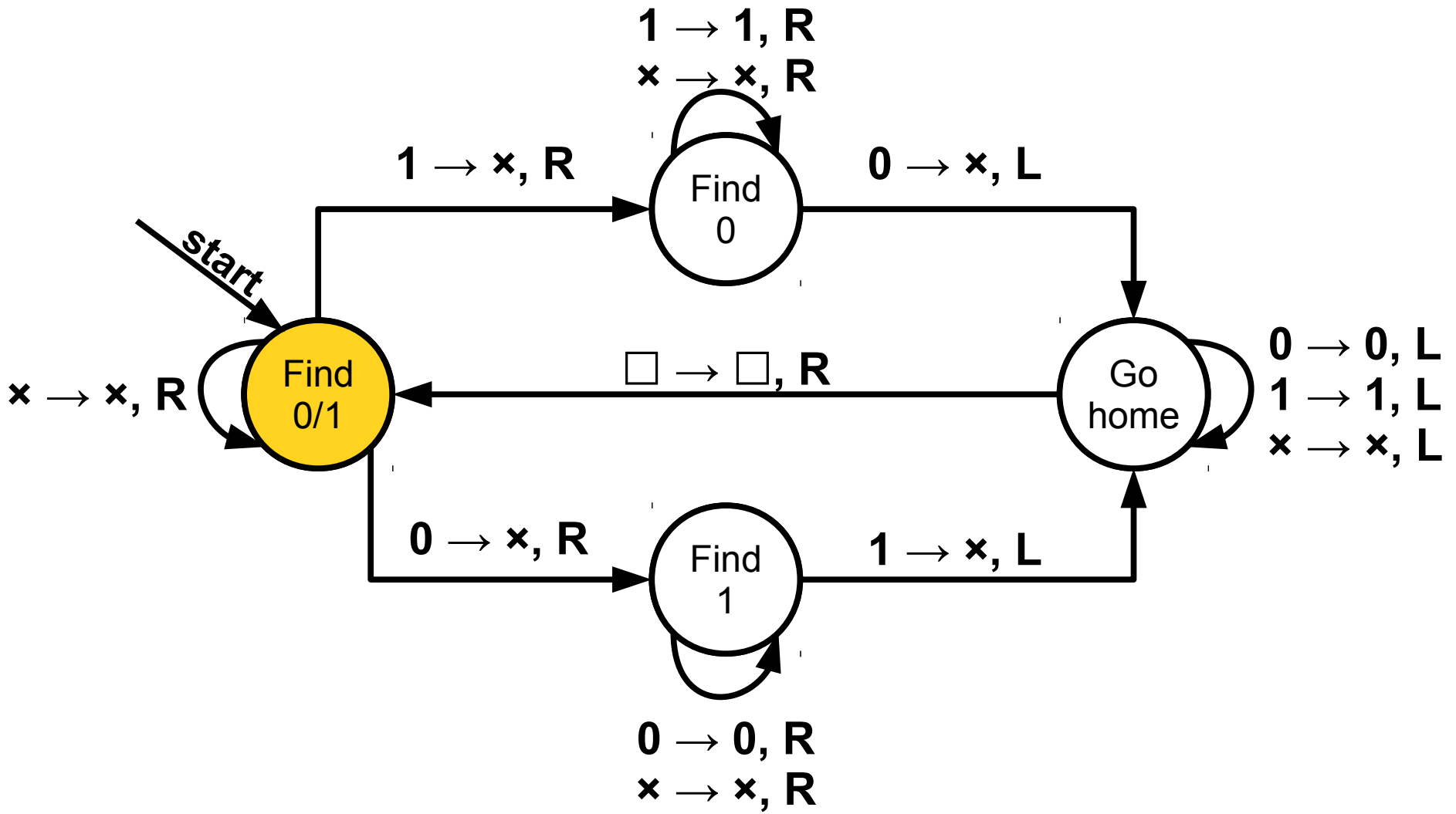

…
×
×
×
×
×
×
×
×
…

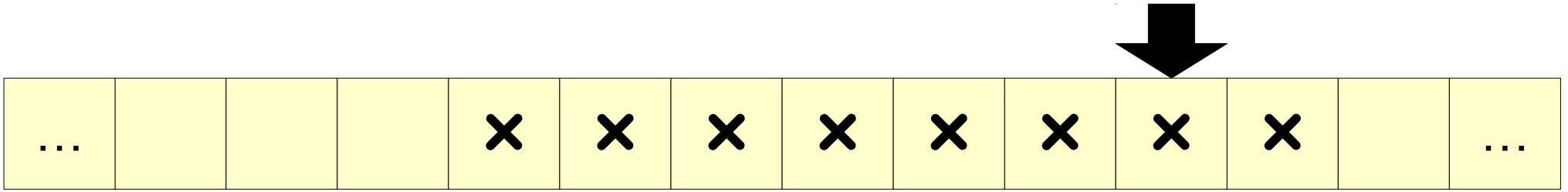

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R

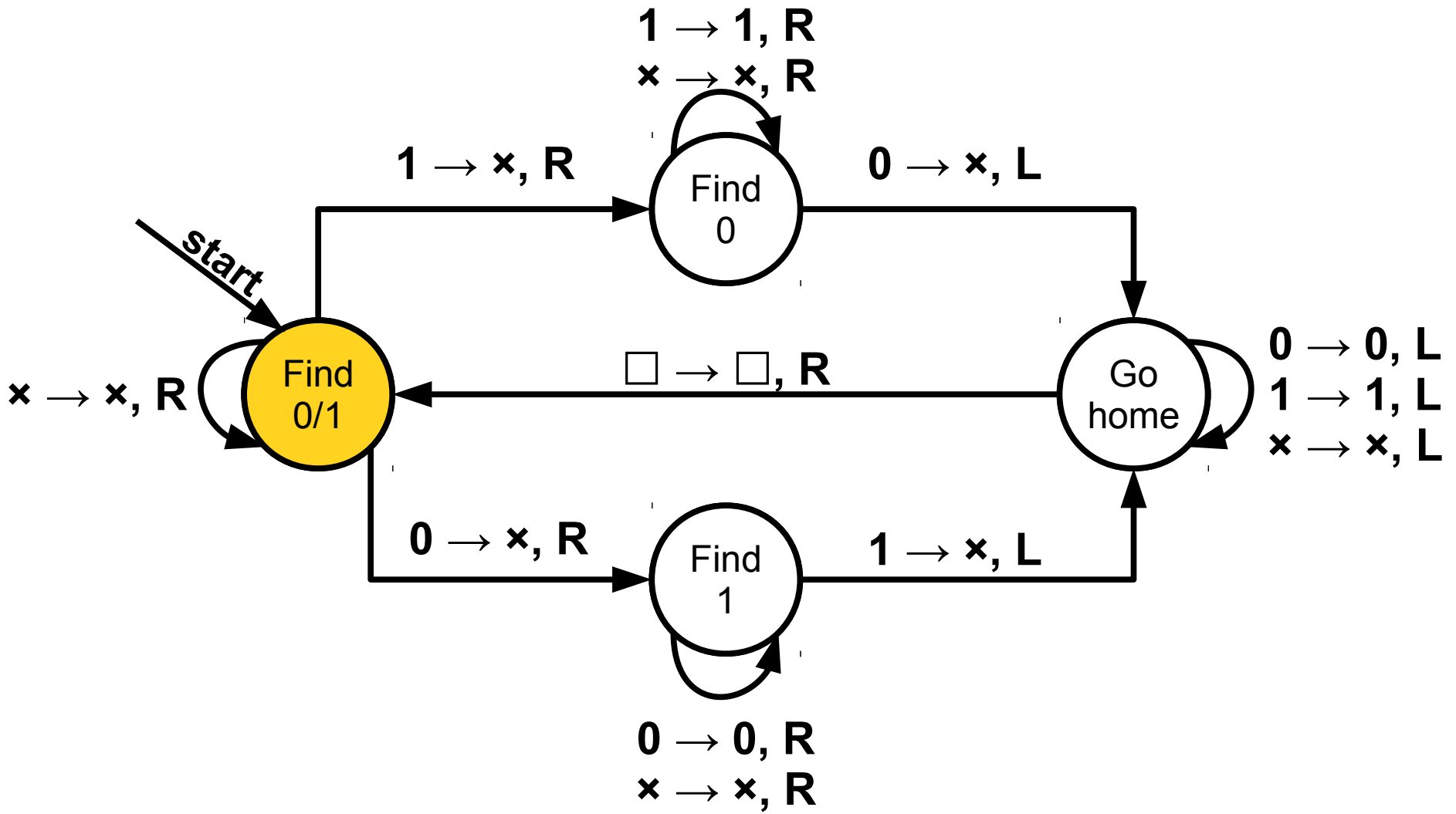

…
×
×
×
×
×
×
×
×
…

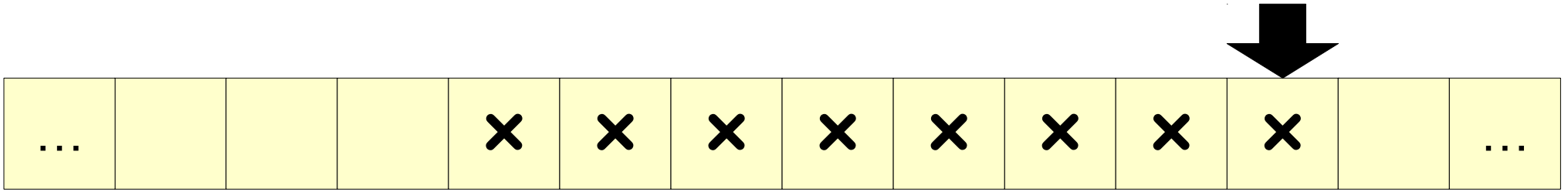

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

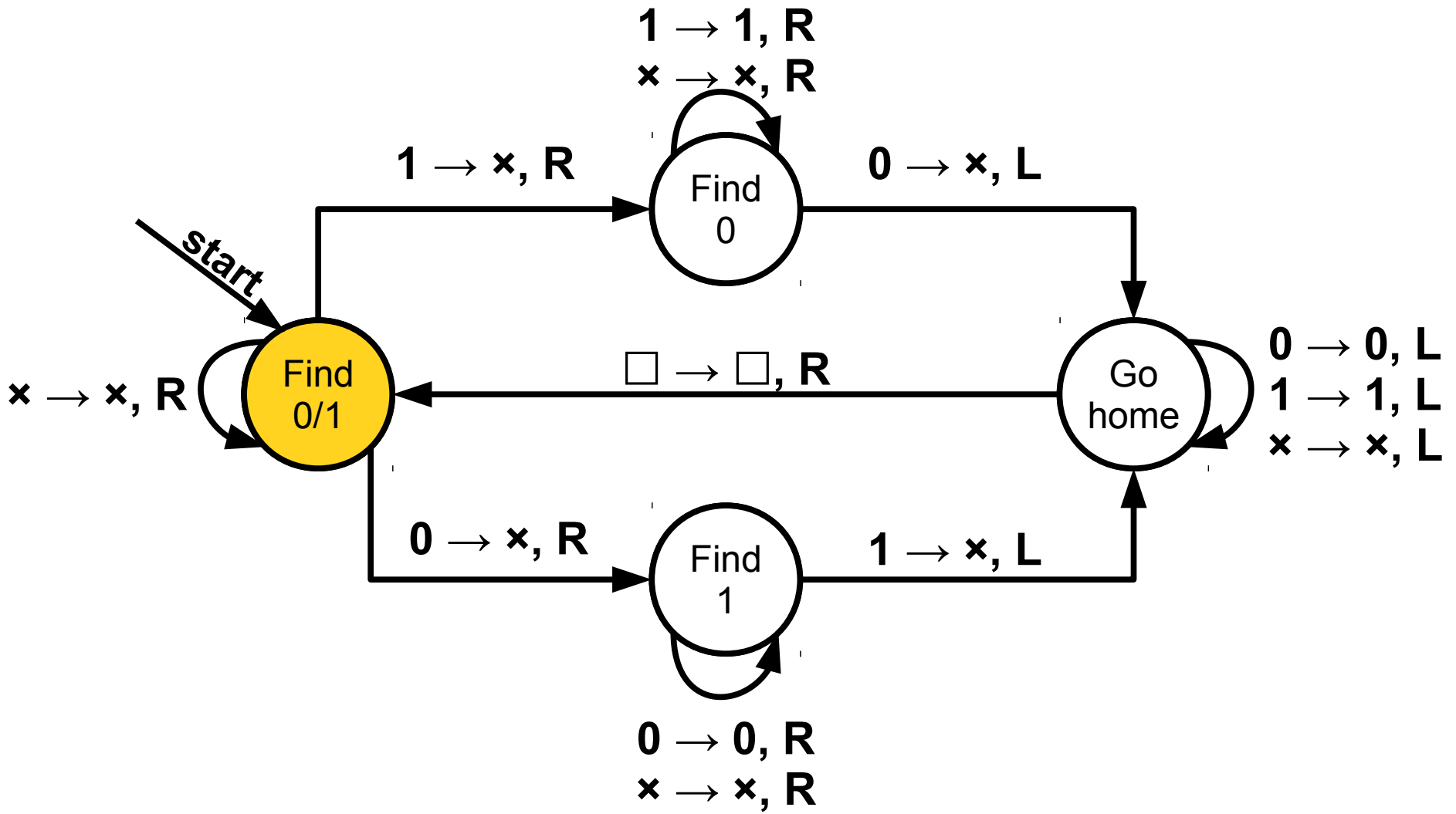

…
×
×
×
×
×
×
×
×
…

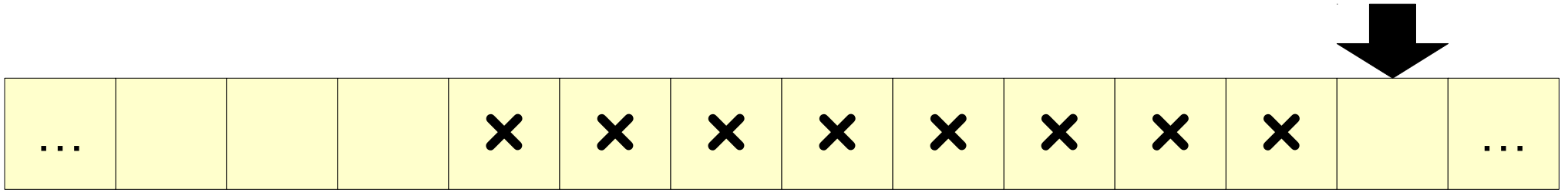

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find 1
0 → 0, R
× → ×, R

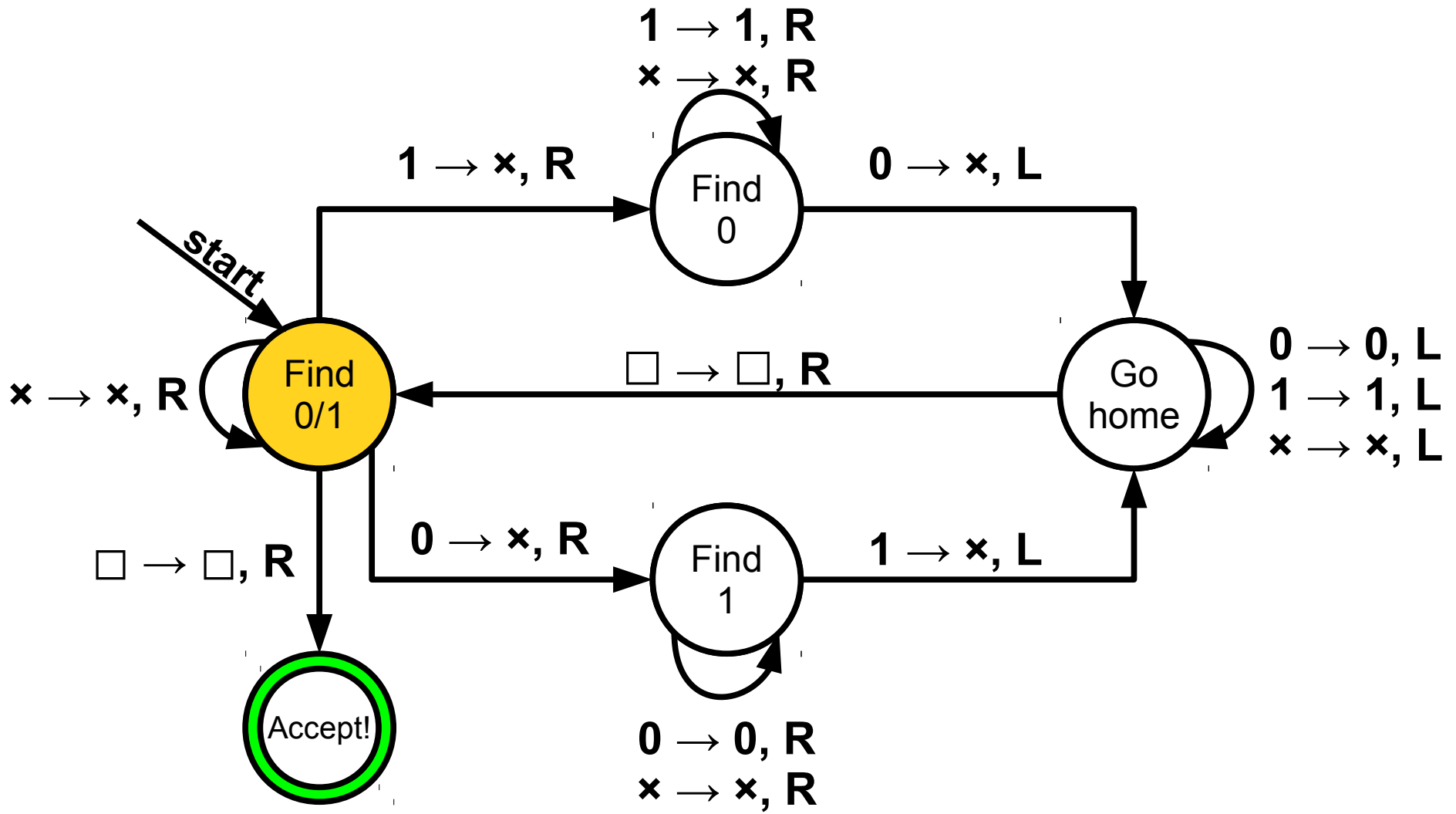

…
×
×
×
×
×
×
×
×
…

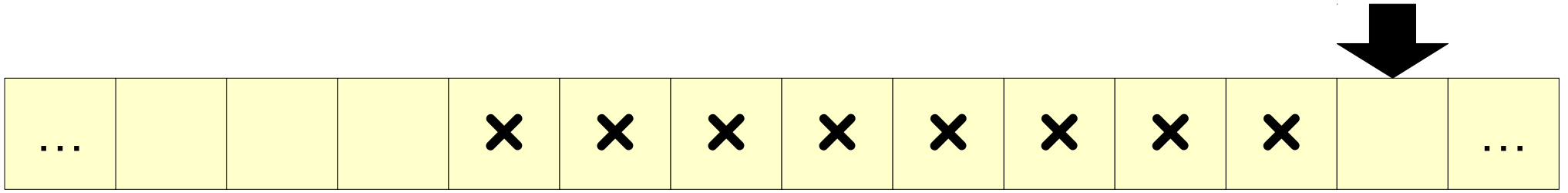

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find
1
1 → ×, L
0 → 0, R
× → ×, R

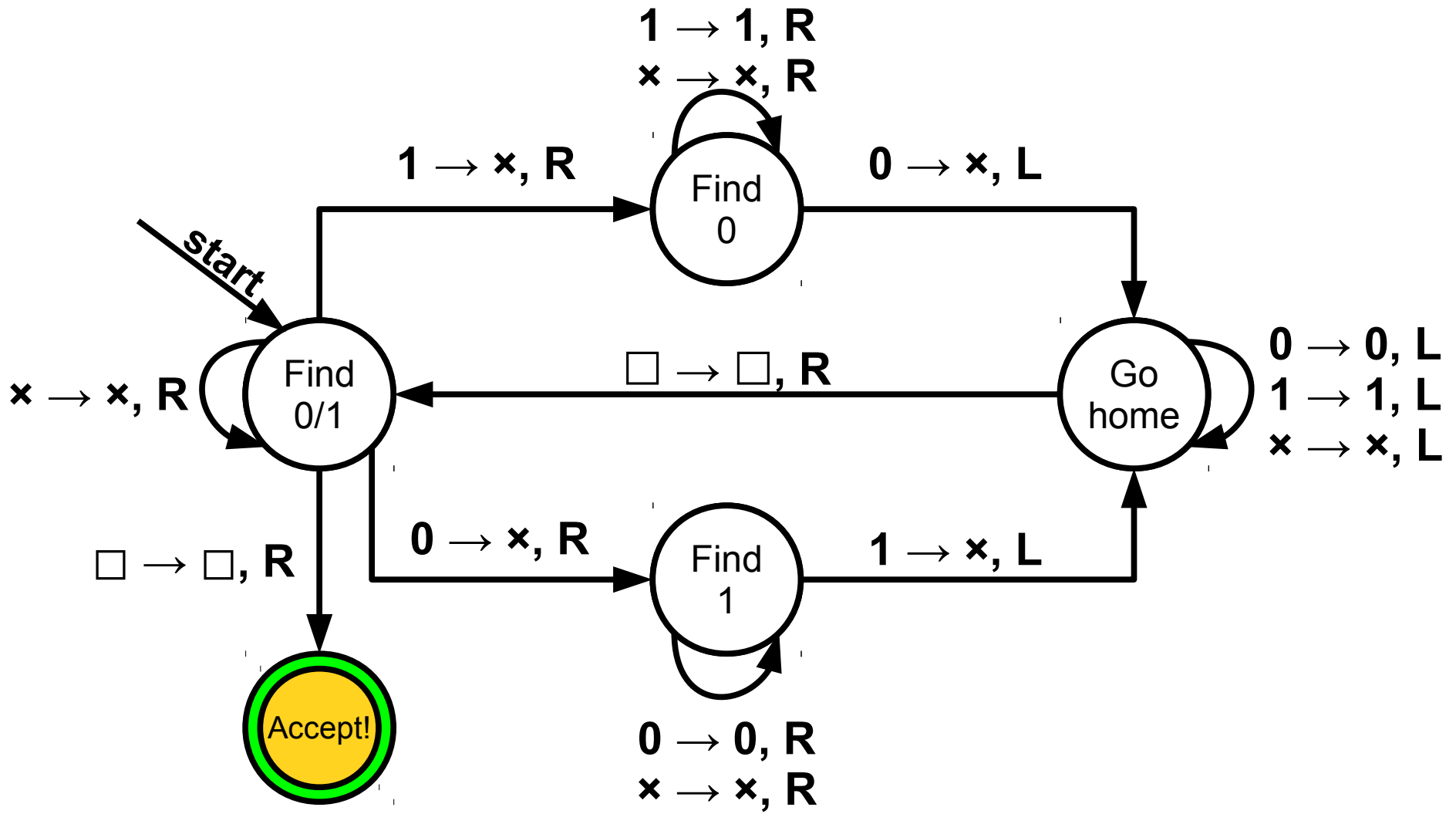

…
×
×
×
×
×
×
×
×
…

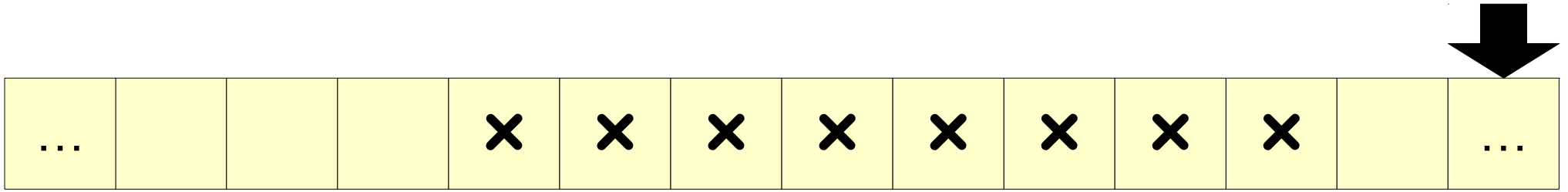

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find
0
Find
0/1
× → ×, R
□ → □, R
Go
home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
Find
1
0 → 0, R
× → ×, R

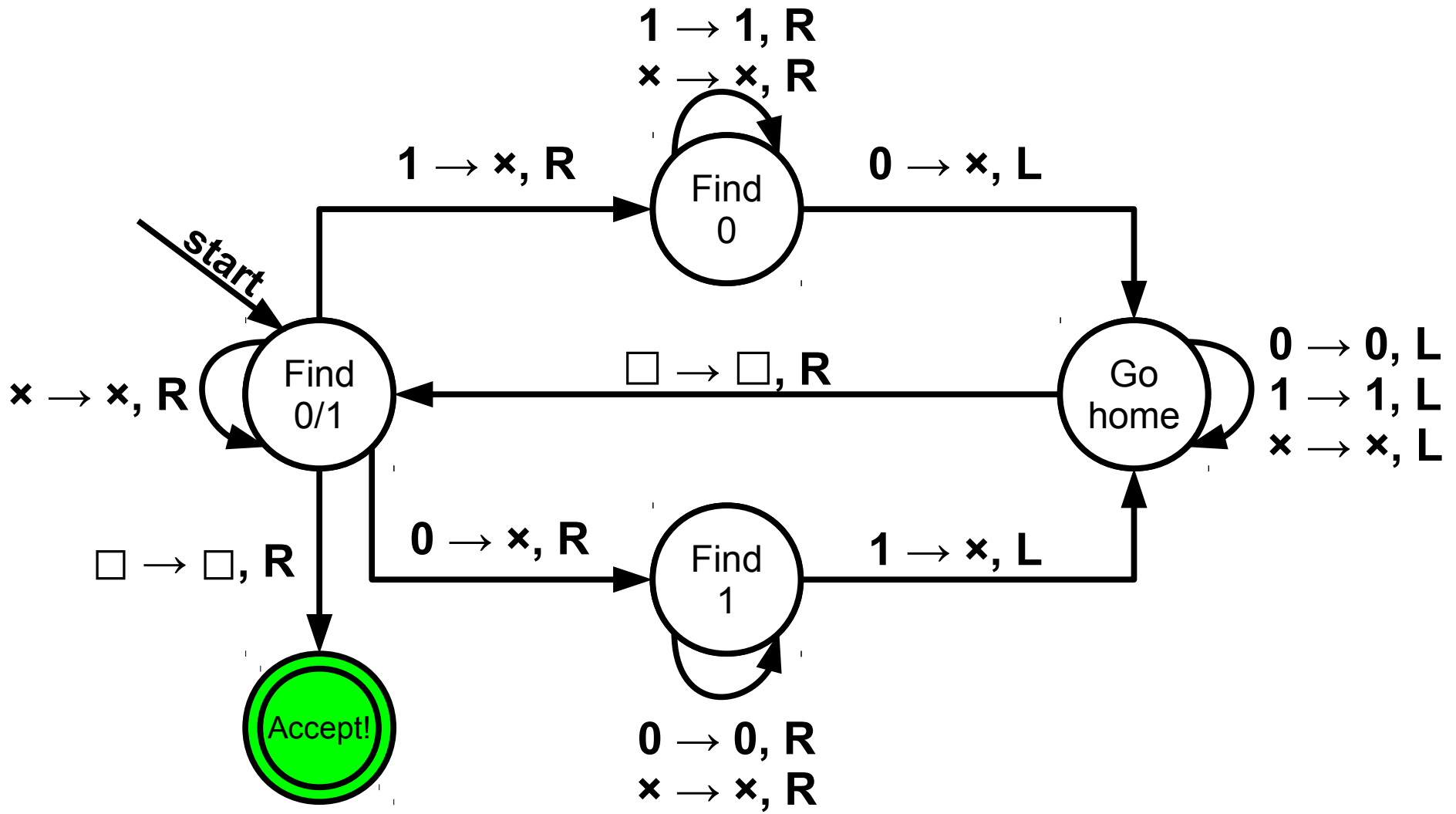

…
×
×
×
×
×
×
×
×
…

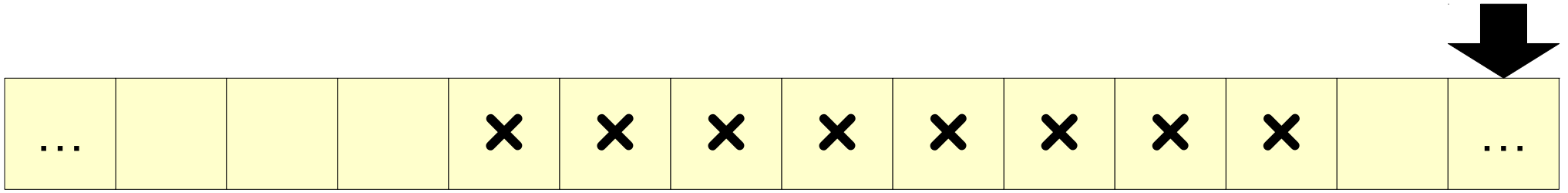

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
Go home
× → ×, R
□ → □, R
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
1 → ×, L
□ → □, R
Find 1
Accept!
0 → 0, R
× → ×, R

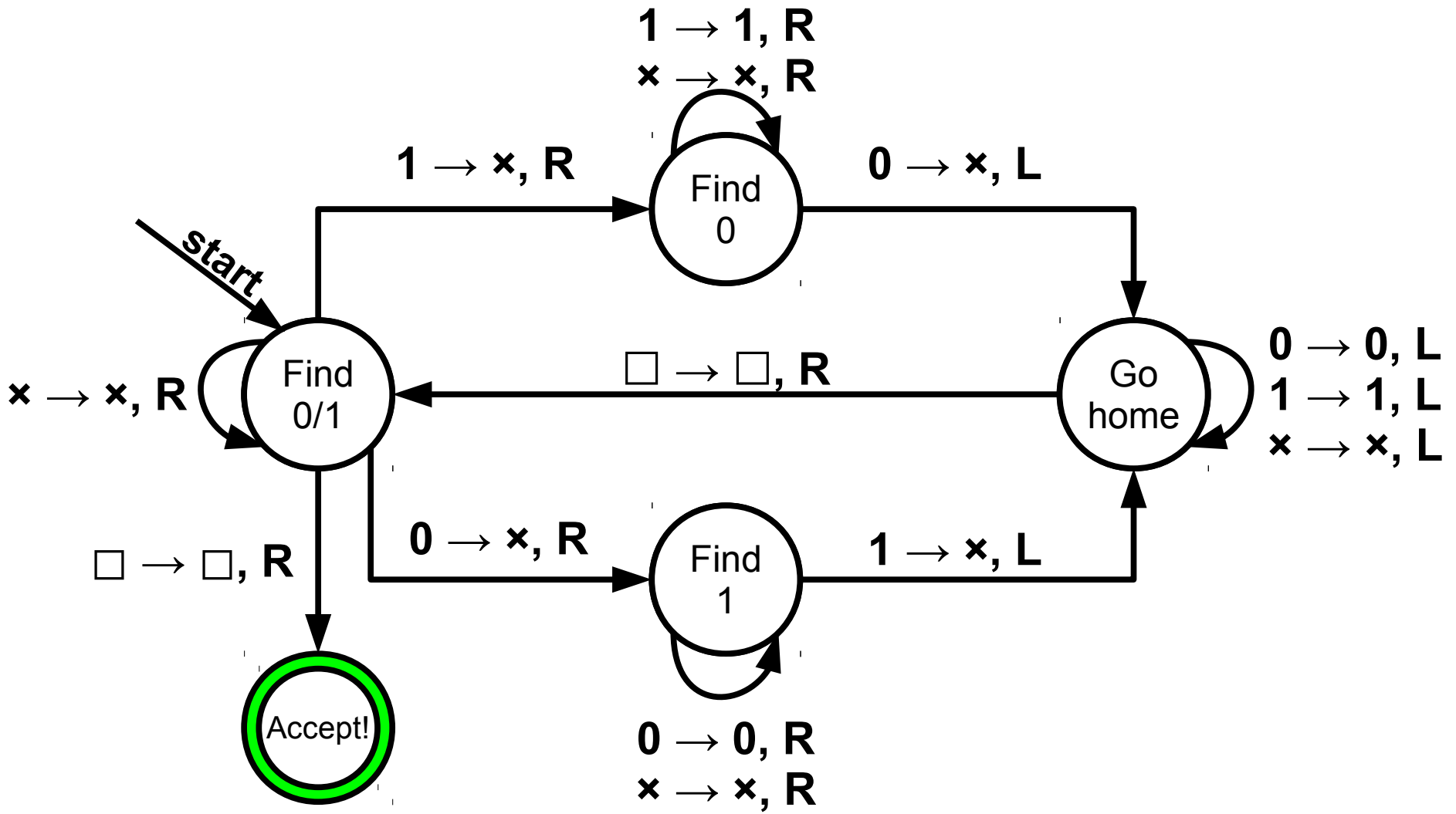

…
×
×
×
×
×
×
×
×
…

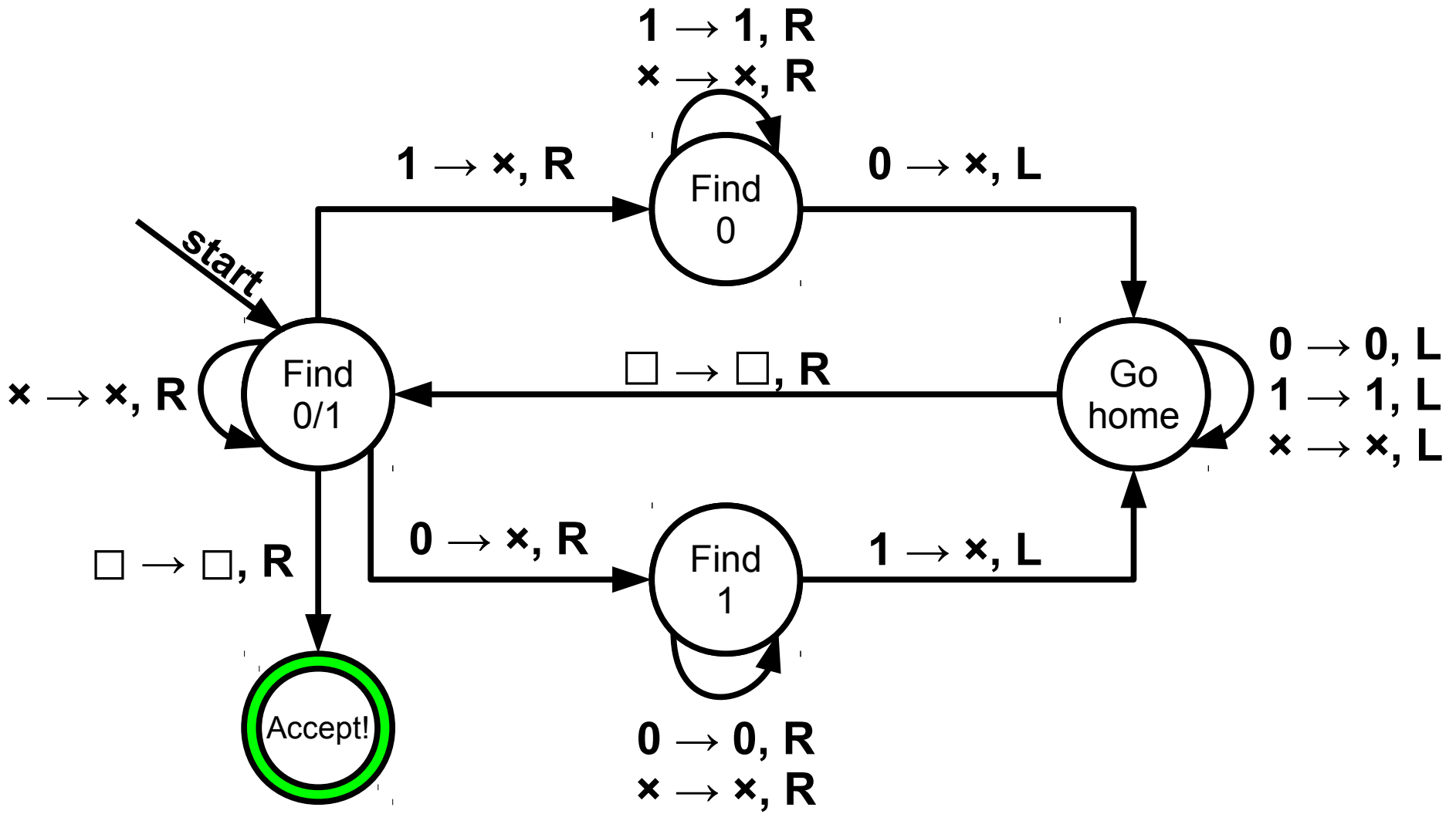

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
Go home
× → ×, R
□ → □, R
0 → 0, L
1 → 1, L
× → ×, L
□ → □, R
0 → ×, R
Find 1
1 → ×, L
Accept!
0 → 0, R
× → ×, R


…
×
×
×
×
×
×
×
×
…

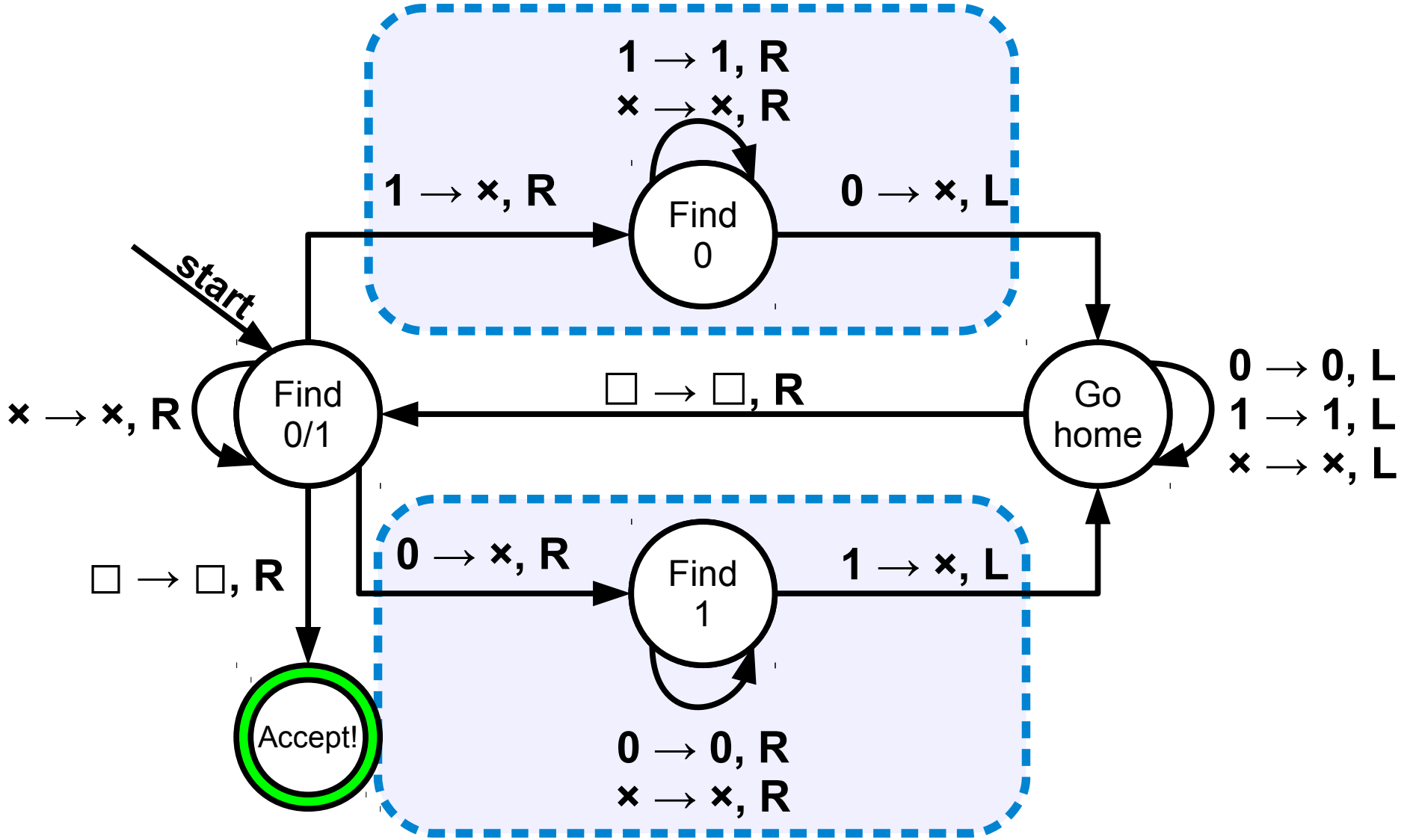

1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
Go home
× → ×, R
□ → □, R
0 → 0, L
1 → 1, L
× → ×, L
□ → □, R
0 → ×, R
Find 1
1 → ×, L
Accept!
0 → 0, R
× → ×, R


…
×
×
×
×
×
×
×
×
…


1 → 1, R
× → ×, R
1 → ×, R
0 → ×, L
start
Find 0
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
□ → □, R
0 → ×, R
Find 1
1 → ×, L
Accept!
0 → 0, R
× → ×, R



For now, let's ignore the missing transitions and pretend that they implicitly reject.


1 → 1, R
× → ×, R
1 → ×, R
Find 0
0 → ×, L
start
Find 0/1
× → ×, R
□ → □, R
Go home
0 → 0, L
1 → 1, L
× → ×, L
0 → ×, R
Find 1
1 → ×, L
□ → □, R
Accept!
0 → 0, R
× → ×, R

Constant Storage

- Sometimes, a TM needs to remember some additional information that can't be put on the tape.
- In this case, you can use similar techniques from DFAs and introduce extra states into the TM's finite-state control.
- The finite-state control can only remember one of finitely many things, but that might be all that you need!

Time-Out for Announcements!

Midterm Logistics

- Second midterm exam is next **Thursday, November 13** from **7PM – 10PM**, locations TBA.
  - Covers material up through and including PS6, with a focus on topics from PS4 – PS6.
  - Closed-book, closed-computer. You can have one 8.5” × 11” sheet of notes with you when you take the exam.
- Practice midterm is next **Monday, November 10** from 7PM – 10PM in **Annenberg Auditorium** (same time and place as last time.)
- We'll release a second set of practice problems on Friday.

Your Questions

“I find myself routinely falling below the median on homework submissions. The issue is not a lack of understanding the material, but rather losing points here and there due to wording. Do you have any suggestions on how I can improve?”

“I find myself routinely falling below the median on homework submissions. The issue is not a lack of understanding the material, but rather losing points here and there due to wording. Do you have any suggestions on how I can improve?”

“I don't do well on problem sets; it isn't because I don't understand the material, but because I lose one point here and there and it adds up fast. I'm worried about the effect this will have on my final grade – how are final grades being calculated?”

“I find myself routinely falling below the median on homework submissions. The issue is not a lack of understanding the material, but rather losing points here and there due to wording. Do you have any suggestions on how I can improve?”

“I don't do well on problem sets; it isn't because I don't understand the material, but because I lose one point here and there and it adds up fast. I'm worried about the effect this will have on my final grade – how are final grades being calculated?”

“The exams are curved, but is the class as a whole? Or the raw score just be the curved exams averaged with our p-sets, so if you have an 82 point average, you'll have a B-?”

“I find myself routinely falling below the median on homework submissions. The issue is not a lack of understanding the material, but rather losing points here and there due to wording. Do you have any suggestions on how I can improve?”

“I don't do well on problem sets; it isn't because I don't understand the material, but because I lose one point here and there and it adds up fast. I'm worried about the effect this will have on my final grade – how are final grades being calculated?”

“~~The exams are curved,~~ but is the class as a whole? Or the raw score just be the curved exams averaged with our p-sets, so if you have an 82 point average, you'll have a B-?”

“I sometimes feel like I am bad test-taker because I like to think through everything before starting a problem and sometimes allocate too much time on a single problem. So, how do you become a good test-taker? (and become a faster test-taker?)”

“What are benefits of doing research here vs summer internships? You mentioned CURIS and am interested, but am weighing options.”

“What are benefits of doing research here vs summer internships? You mentioned CURIS and am interested, but am weighing options.”

“I'm currently in the process of interviewing for summer internships. I usually do pretty well on the online coding tests, but not so well on phone interviews. How do I practice coding so that I'll do better when a real person asks me a question?”

Back to CS103!

Another TM Design

- Consider the following language over $\Sigma = \{\mathtt{0}, \mathtt{1}\}$:

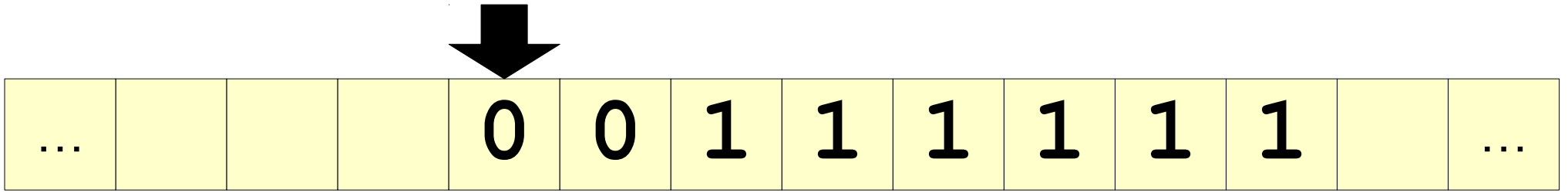
$$L = \{\mathtt{0}^n\mathtt{1}^m \mid n, m \in \mathbb{N} \text{ and } m \text{ is a multiple of } n\}$$

- Is this language regular?
- How might we design a TM for this language?

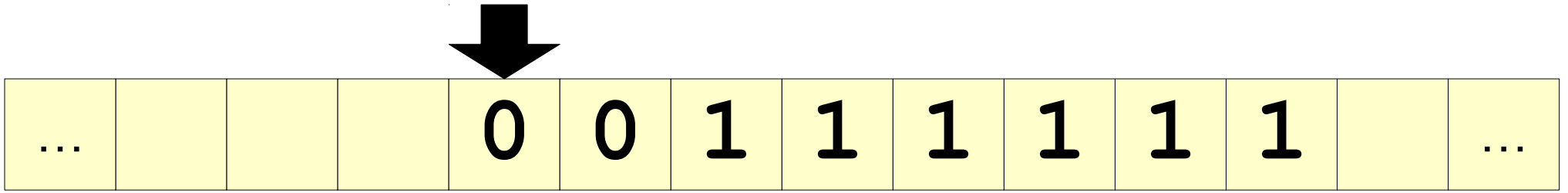
An Observation

- We can recursively describe when one number m is a multiple of n:
  - If $m = 0$, then m is a multiple of n.
  - Otherwise, m is a multiple of n iff $m - n$ is a multiple of n.
- **Idea:** Repeatedly subtract n from m until m becomes zero (good!) or drops below zero (bad!)

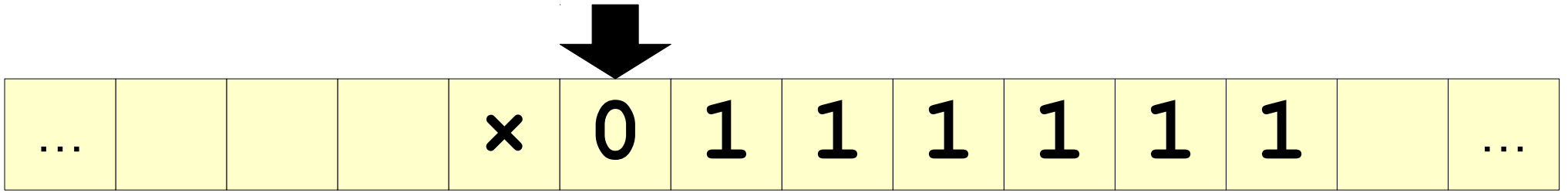
The Challenge



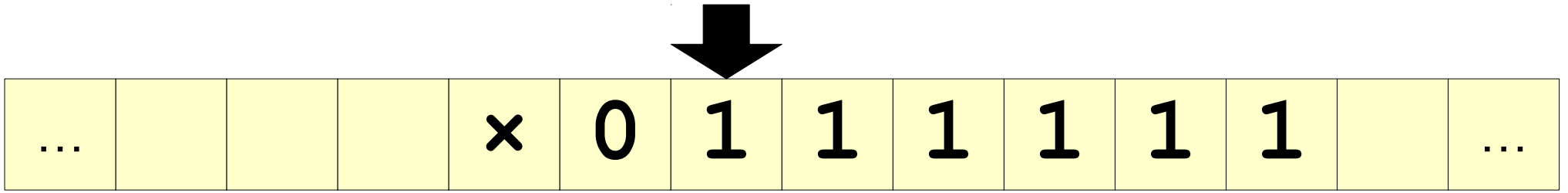
One Solution



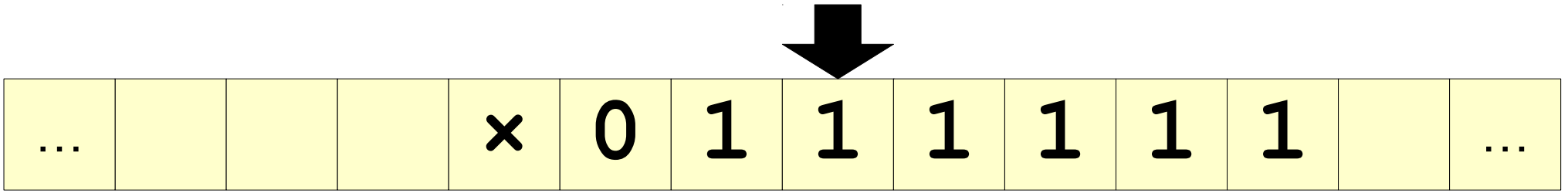
One Solution



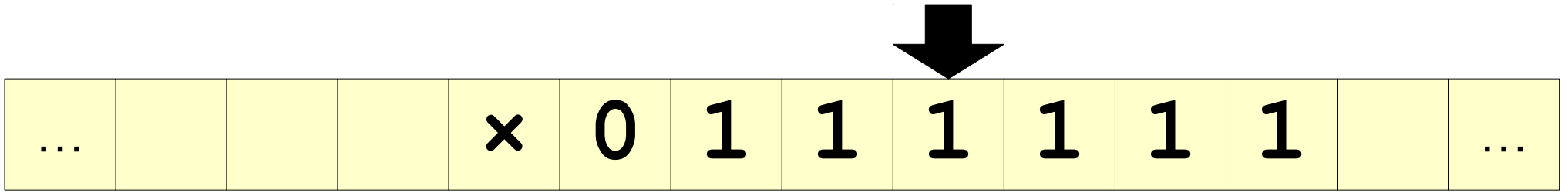
One Solution



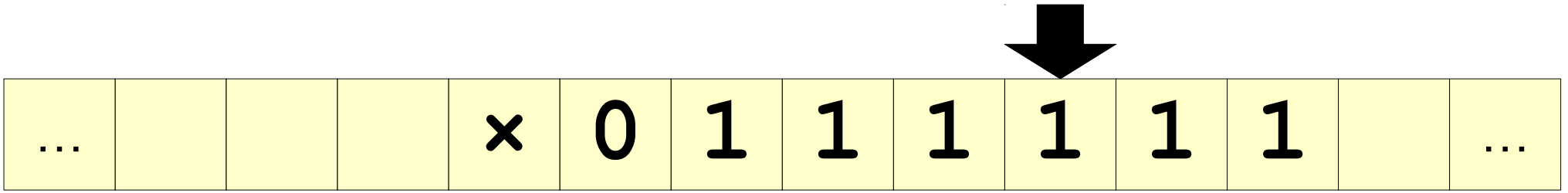
One Solution



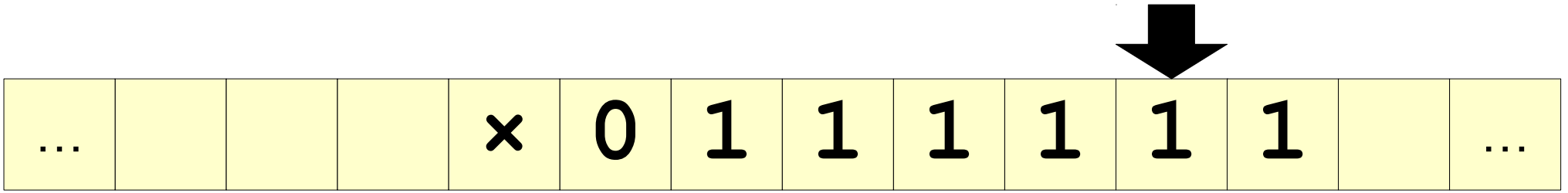
One Solution



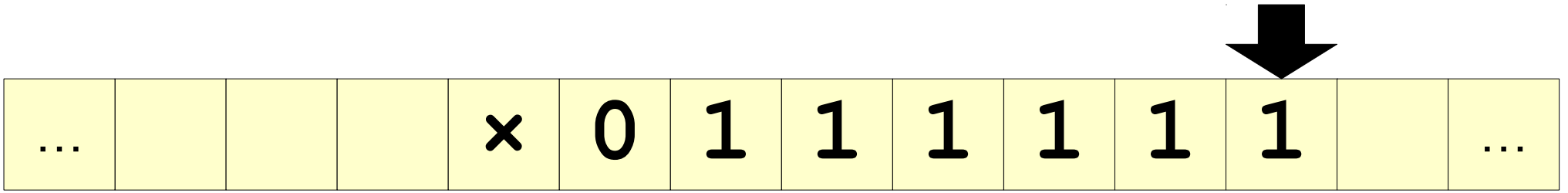
One Solution



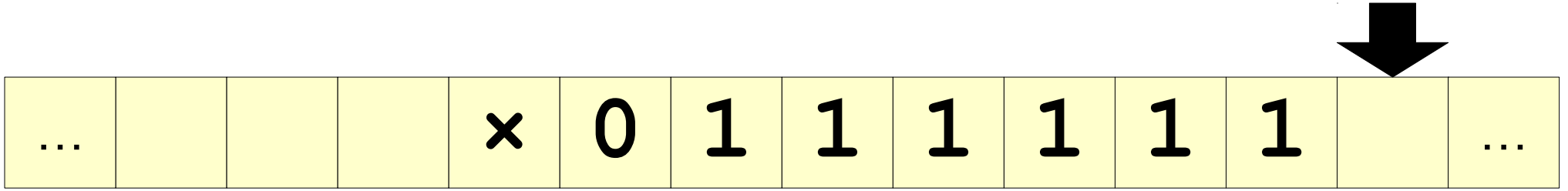
One Solution



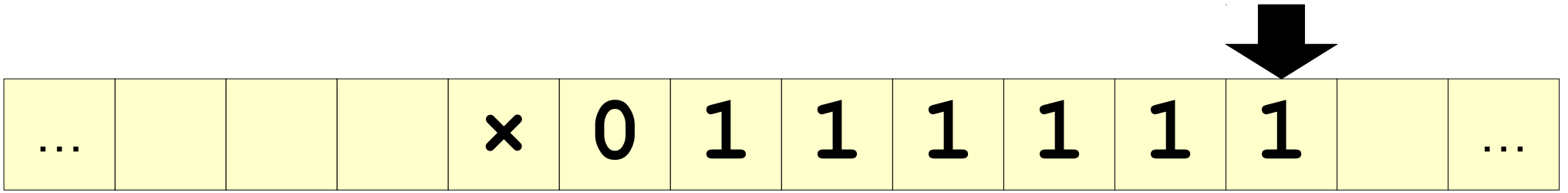
One Solution



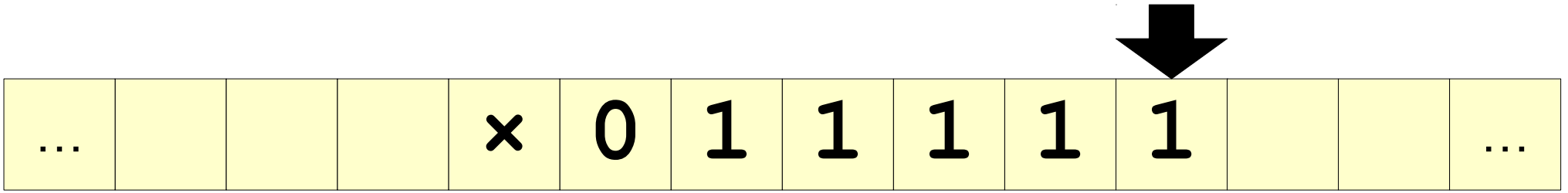
One Solution



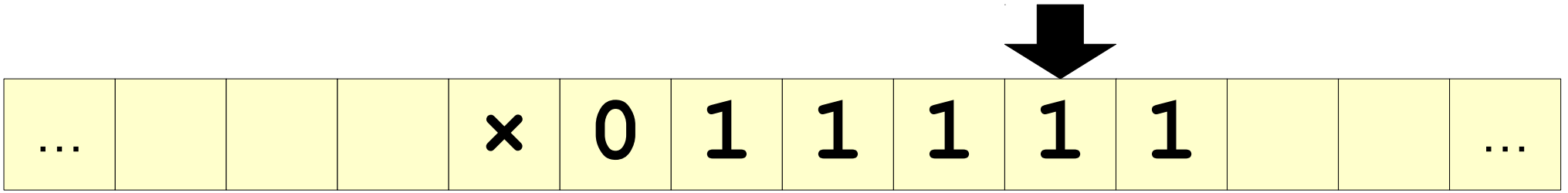
One Solution



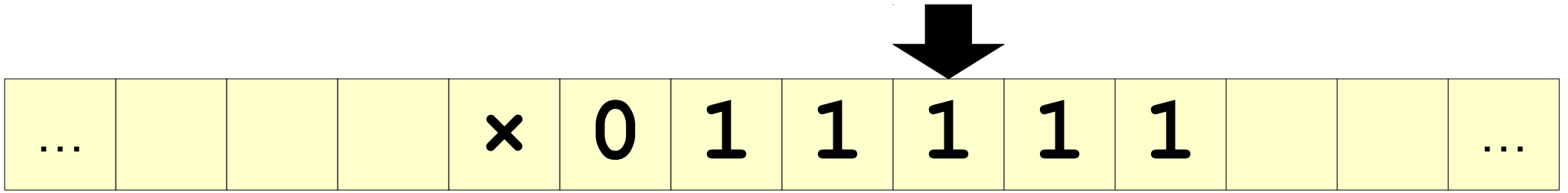
One Solution



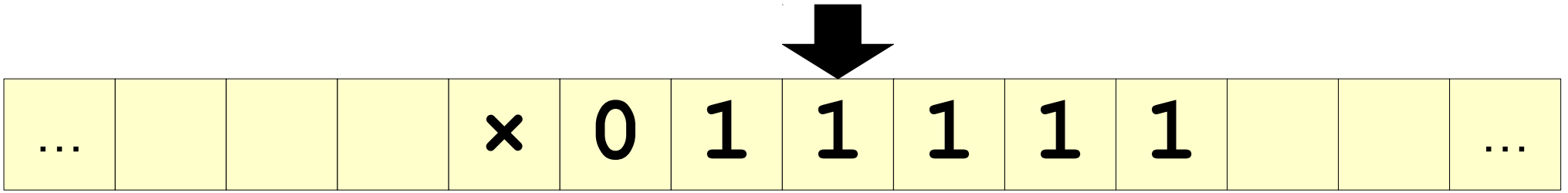
One Solution



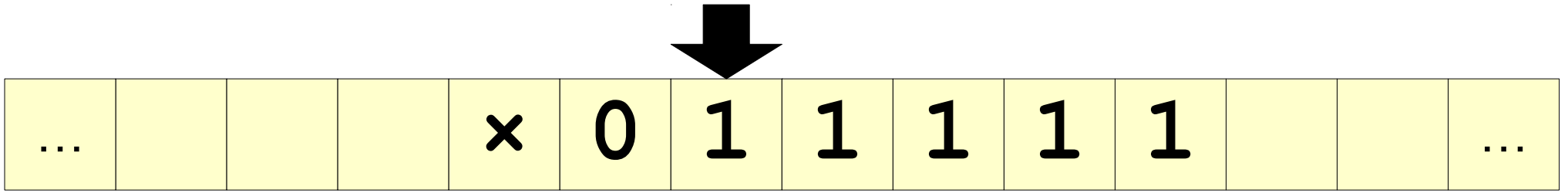
One Solution



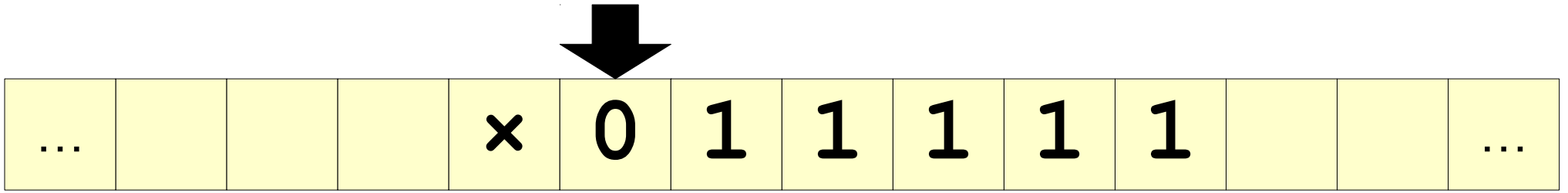
One Solution



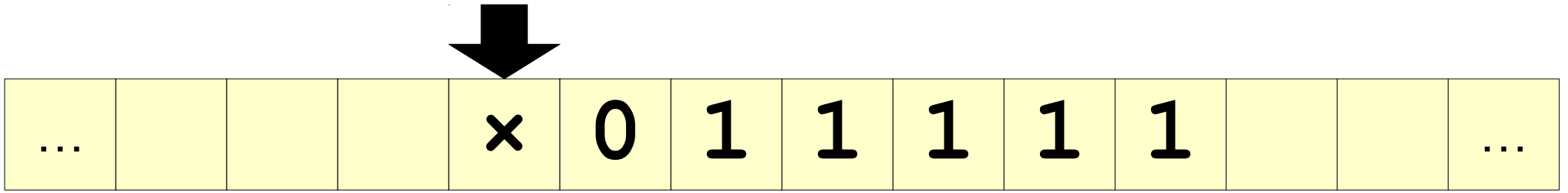
One Solution



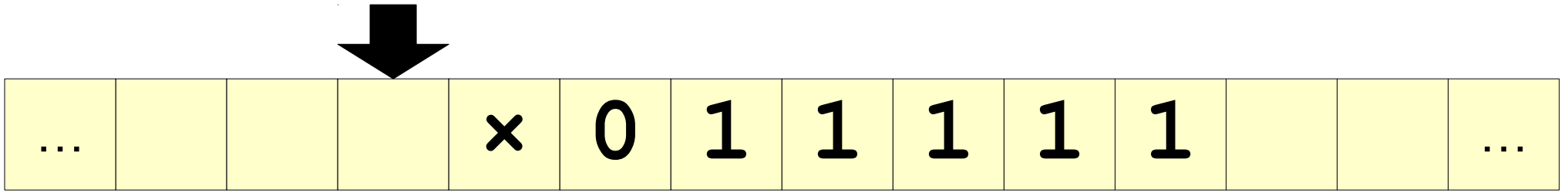
One Solution



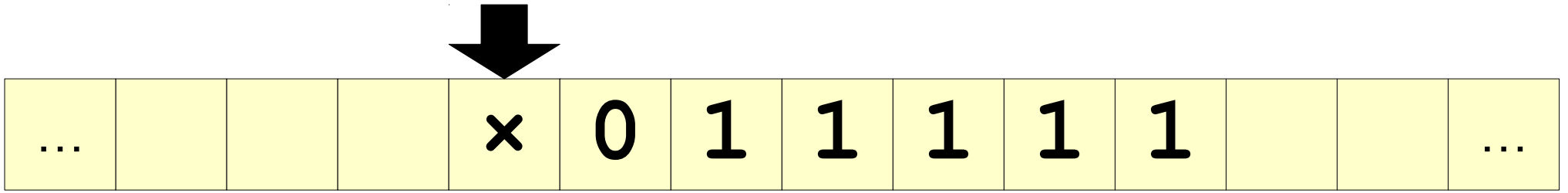
One Solution



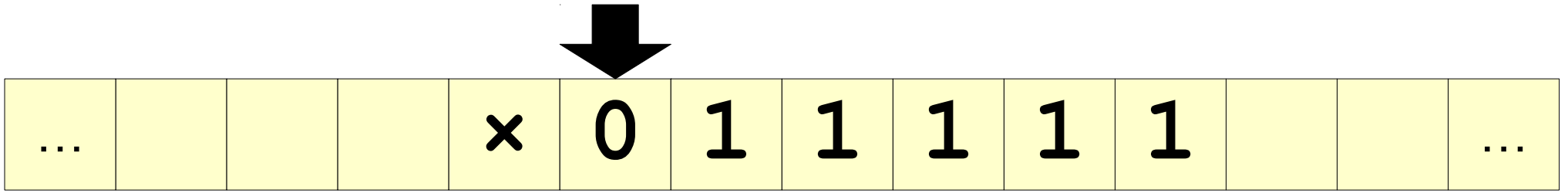
One Solution



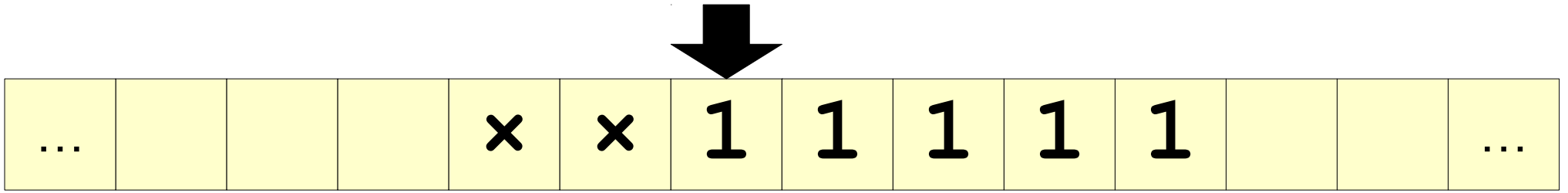
One Solution



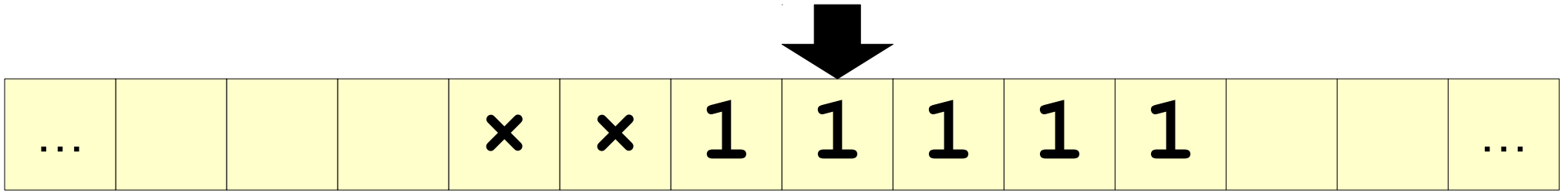
One Solution



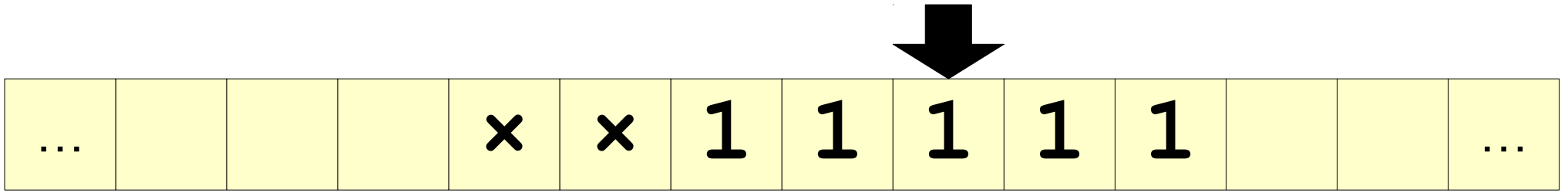
One Solution



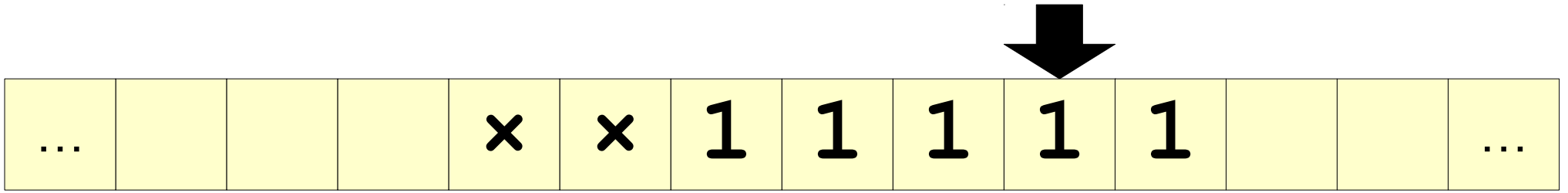
One Solution



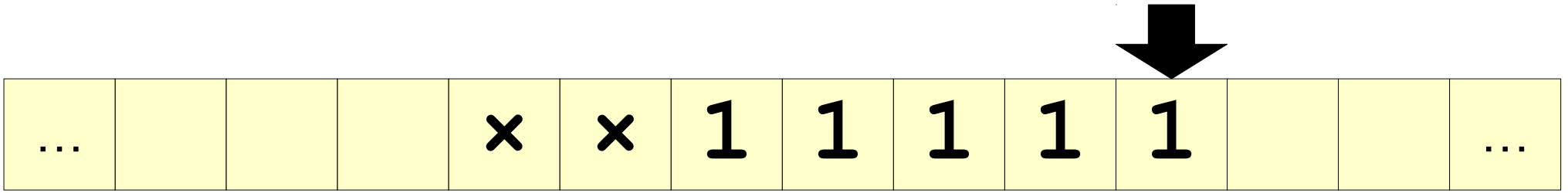
One Solution



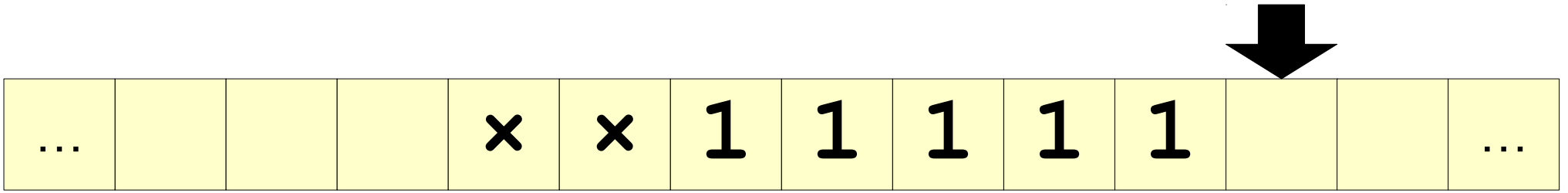
One Solution



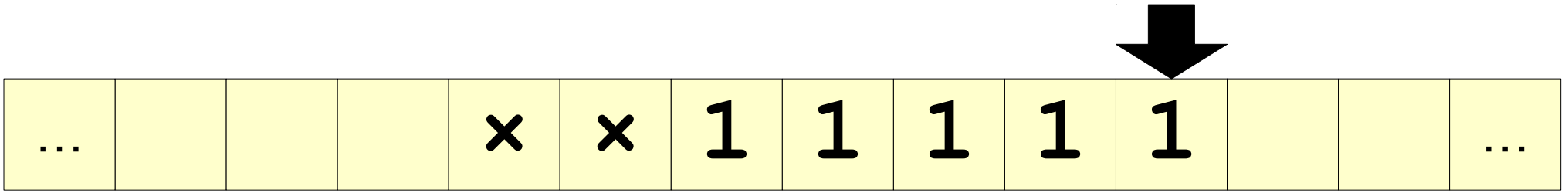
One Solution



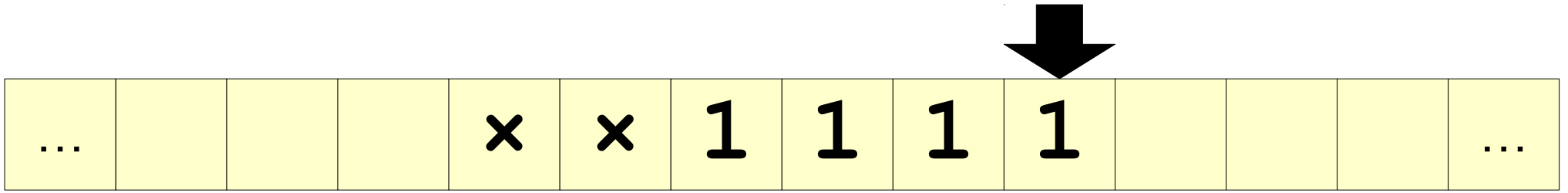
One Solution



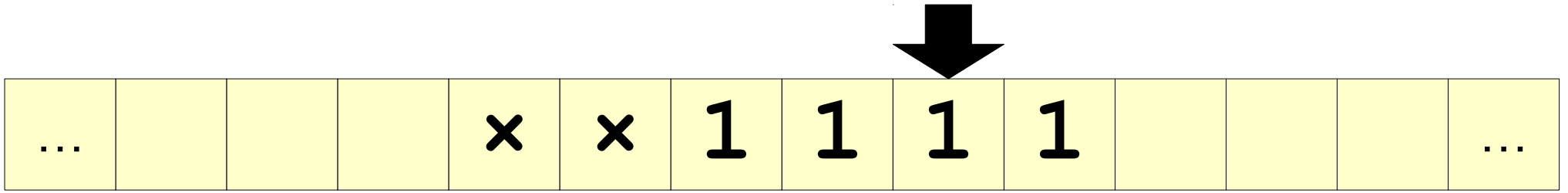
One Solution



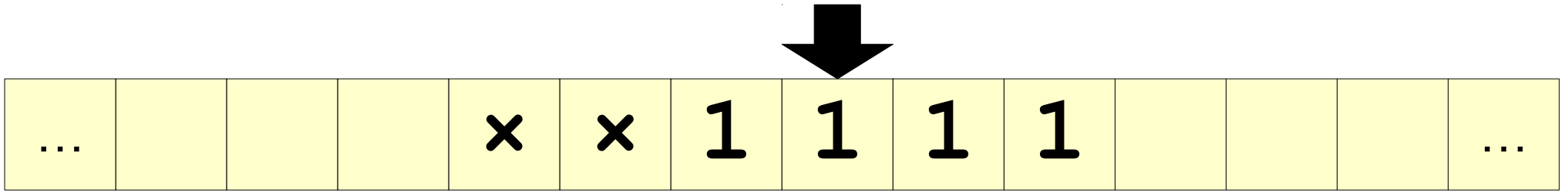
One Solution



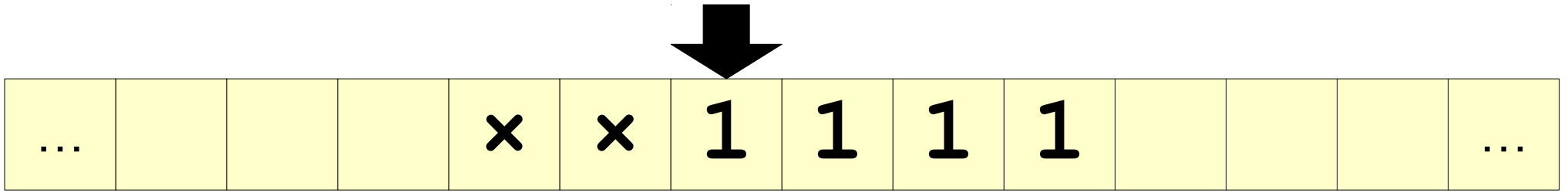
One Solution



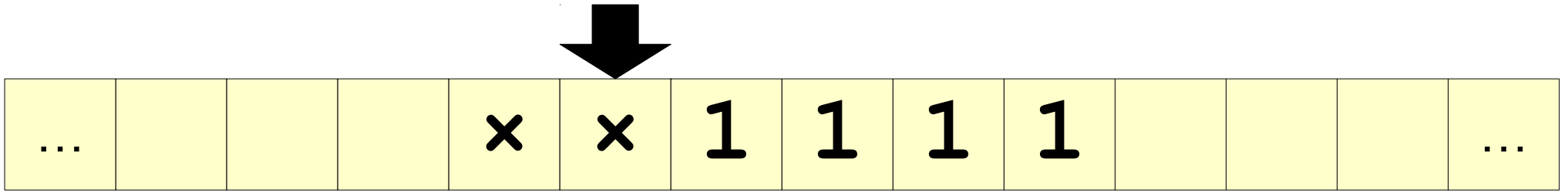
One Solution



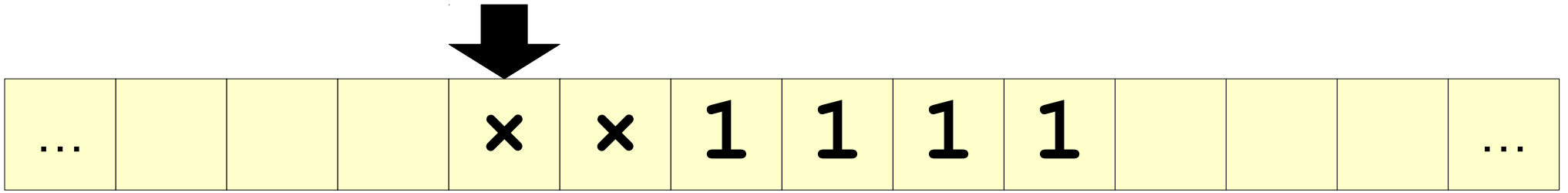
One Solution



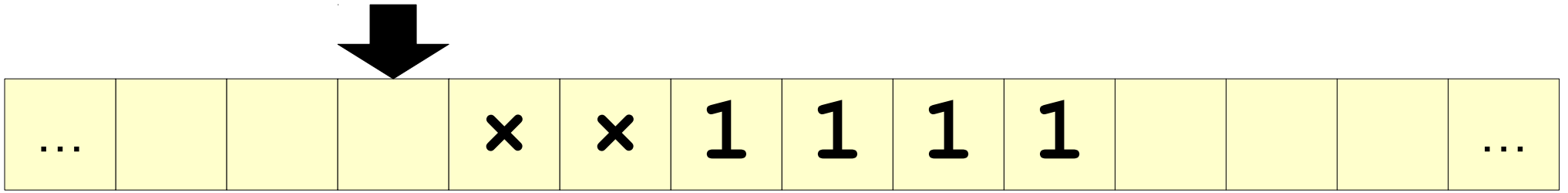
One Solution



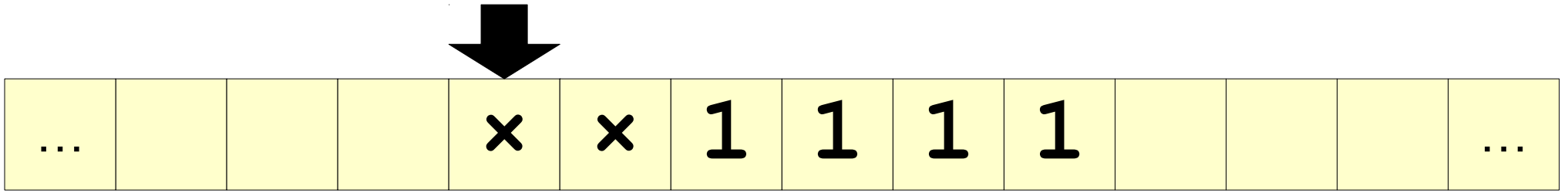
One Solution



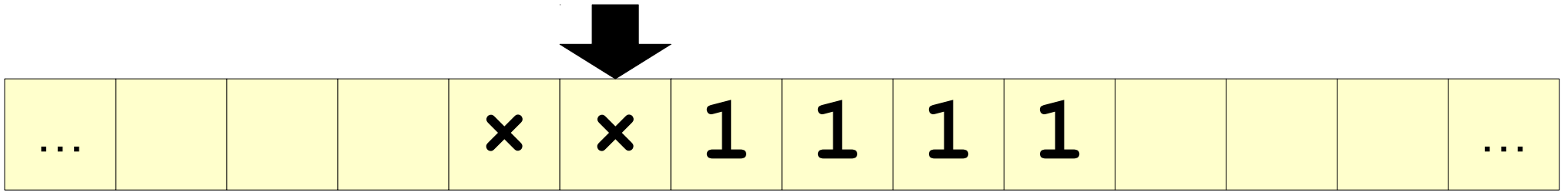
One Solution



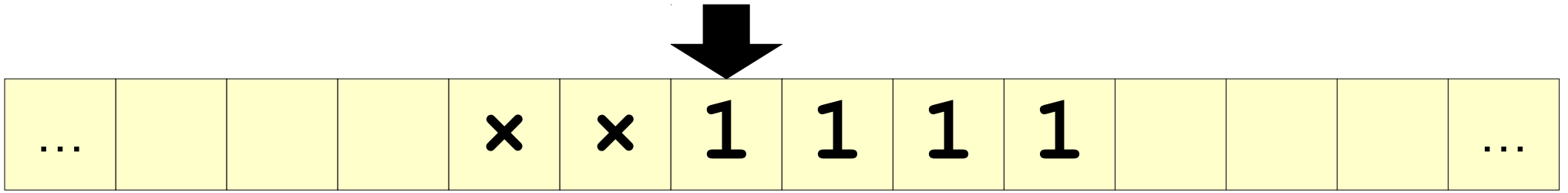
One Solution



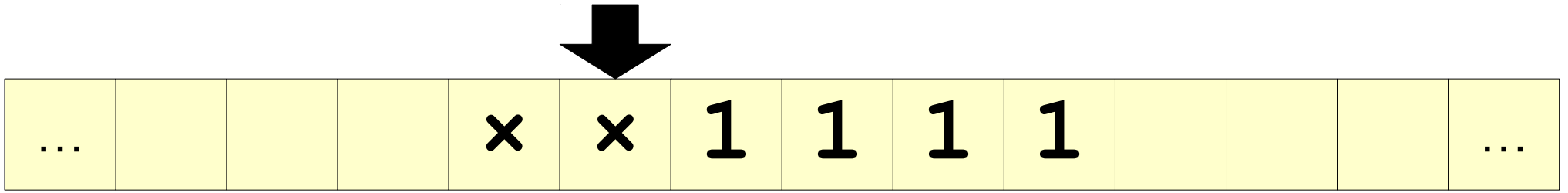
One Solution



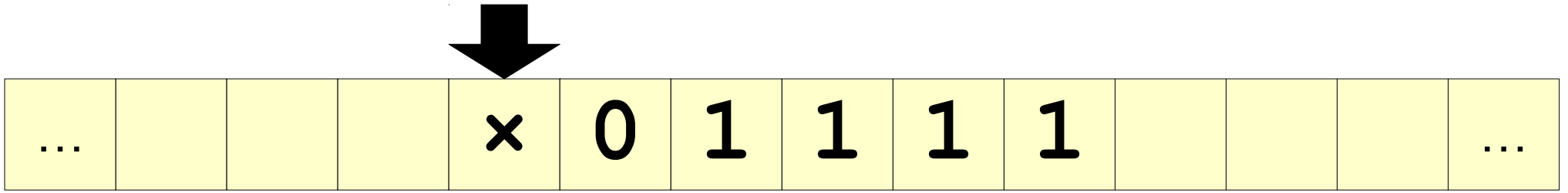
One Solution



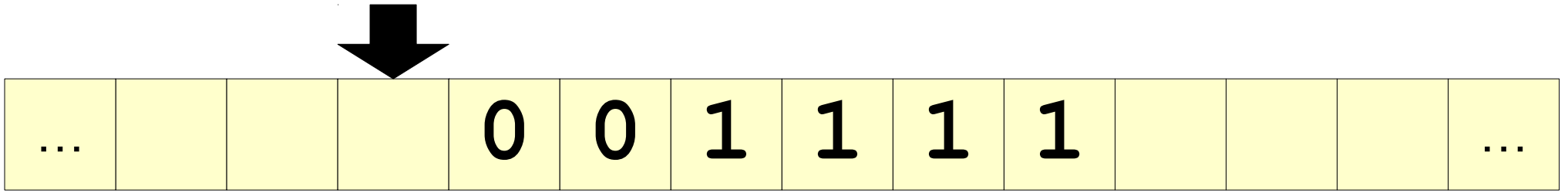
One Solution



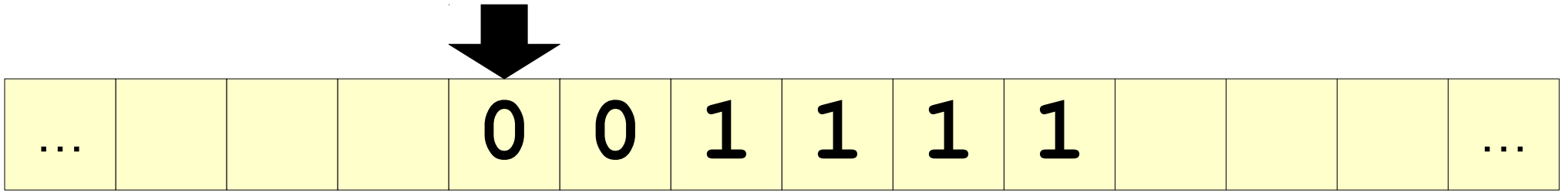
One Solution

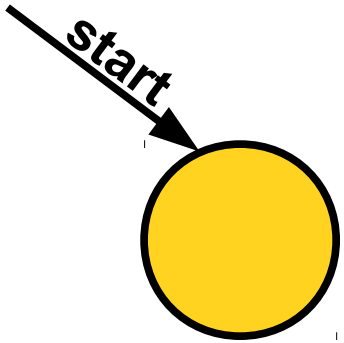


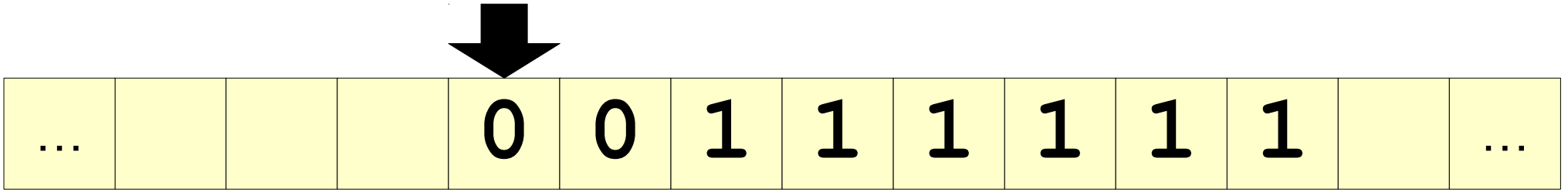
One Solution

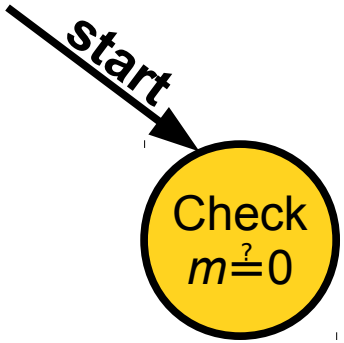


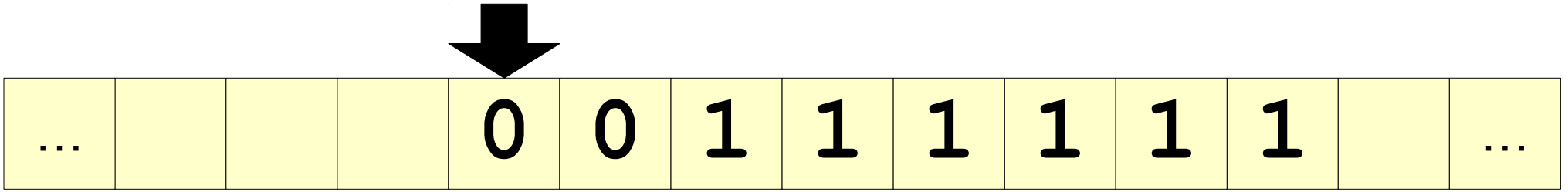
One Solution

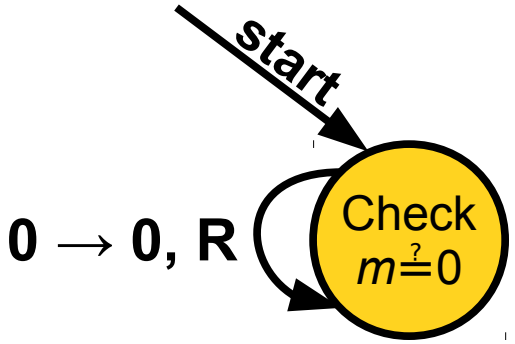


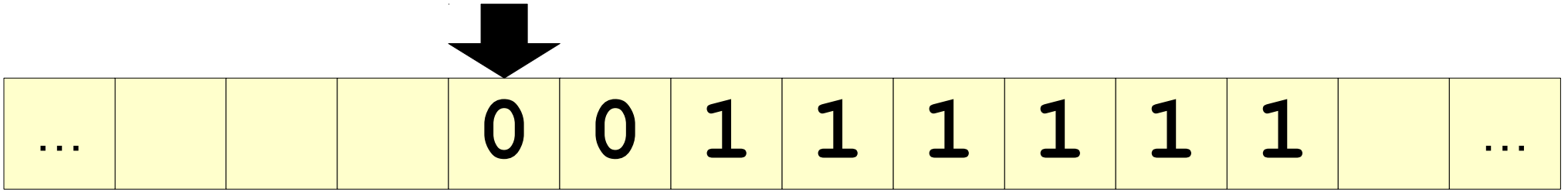

start

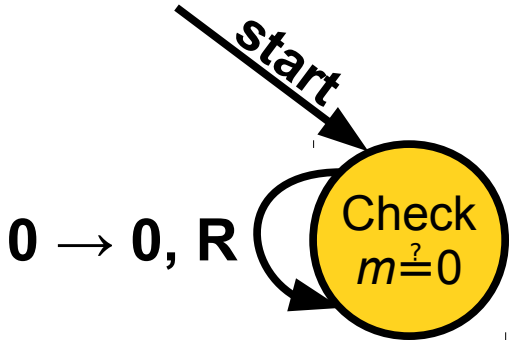

...
0
0
1
1
1
1
1
1
...

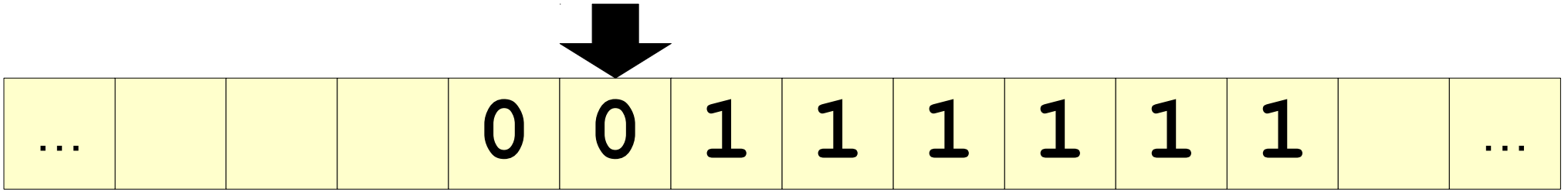

start
Check
$m \stackrel{?}{=} 0$

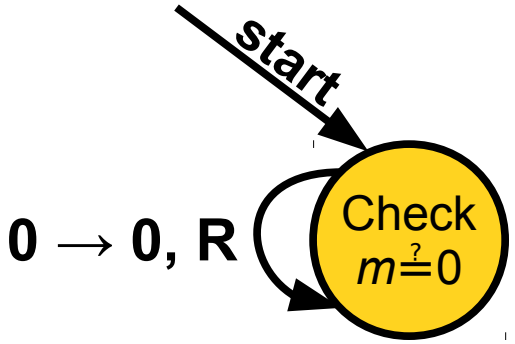

...
0
0
1
1
1
1
1
1
...

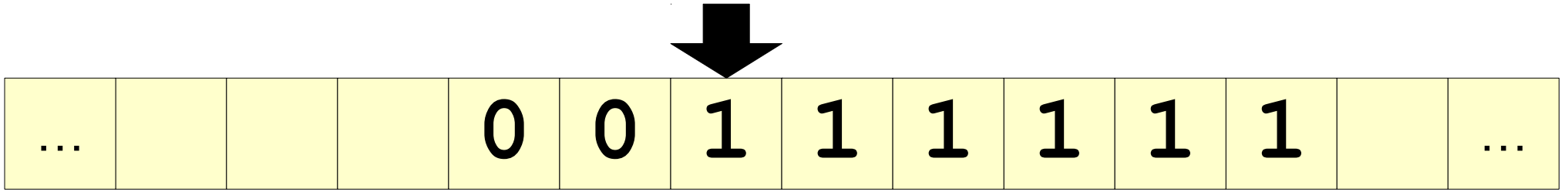

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R

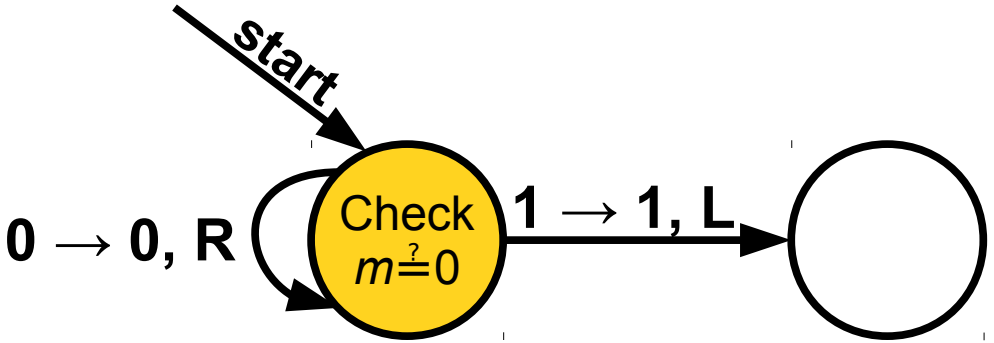

...
0
0
1
1
1
1
1
1
...

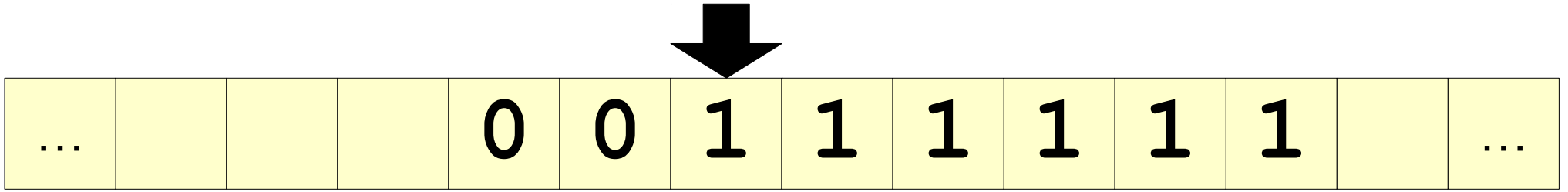

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R

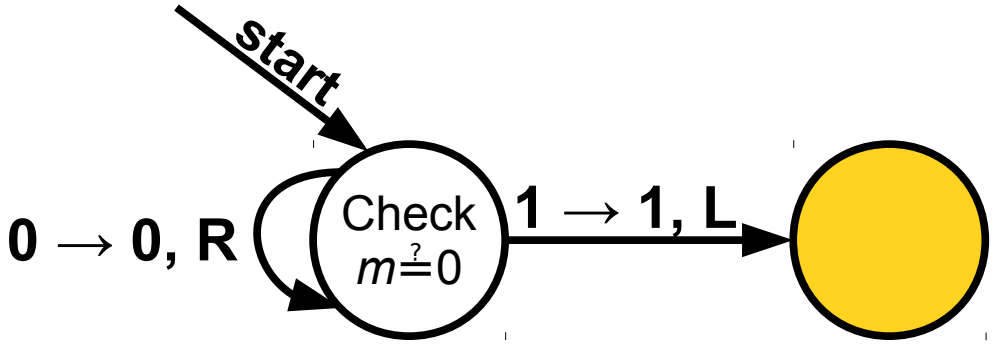

…
0
0
1
1
1
1
1
1
…

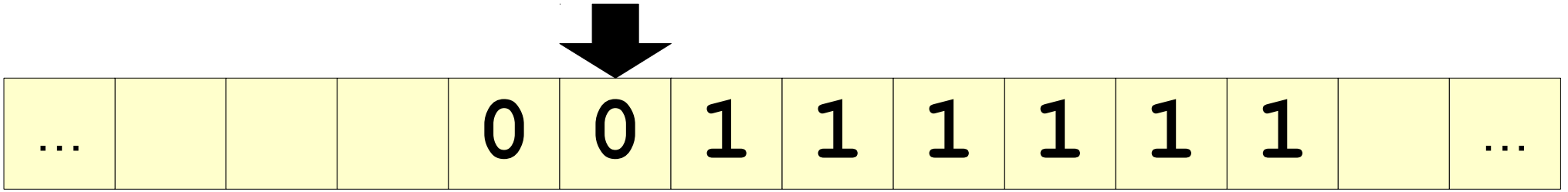

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R

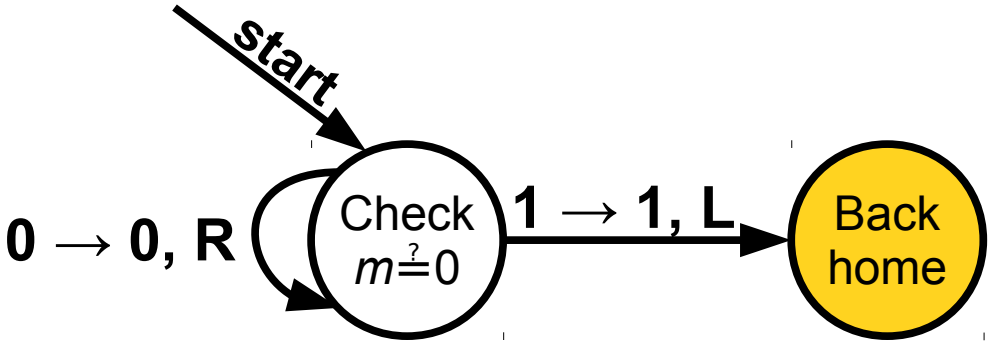

…
0
0
1
1
1
1
1
1
…

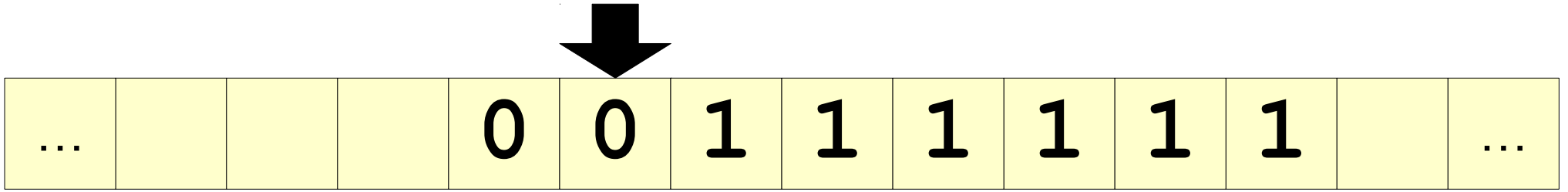

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L

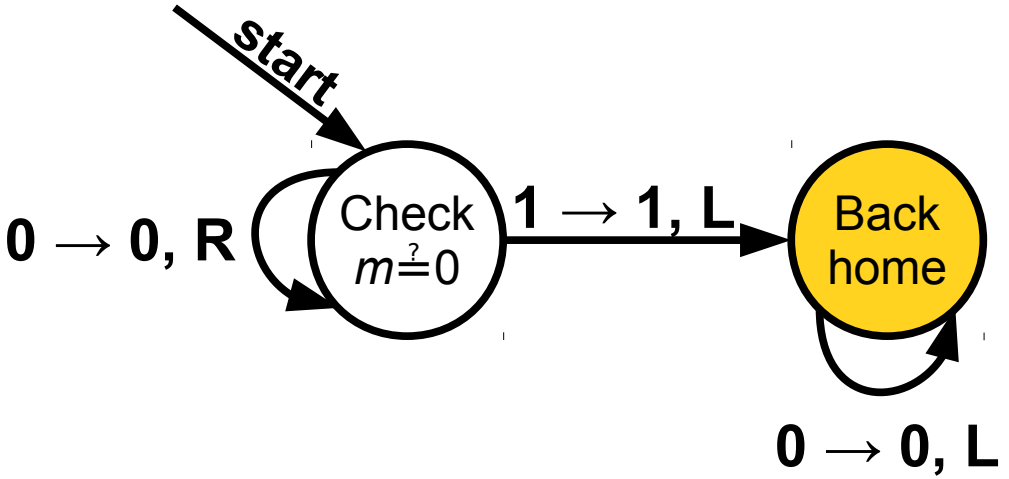

...
0
0
1
1
1
1
1
1
...

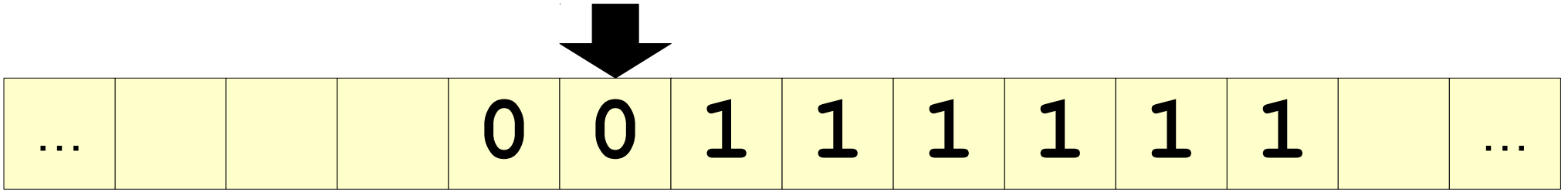

start
Check
$m \stackrel{?}{=} 0$
$0 \rightarrow 0, R$
$1 \rightarrow 1, L$

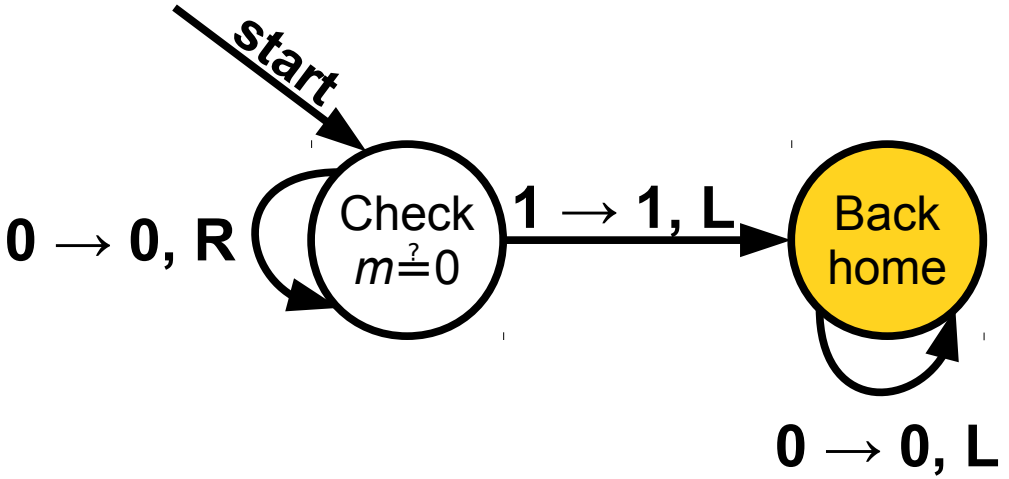

...
0
0
1
1
1
1
1
1
...

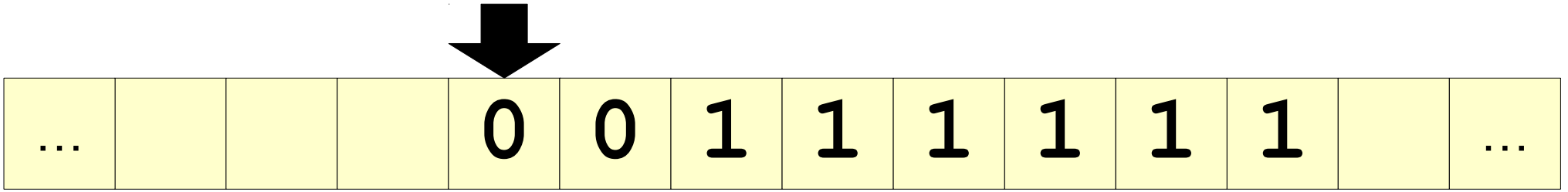

start
Check
$m \stackrel{?}{=} 0$
Back
home
0 → 0, R
1 → 1, L

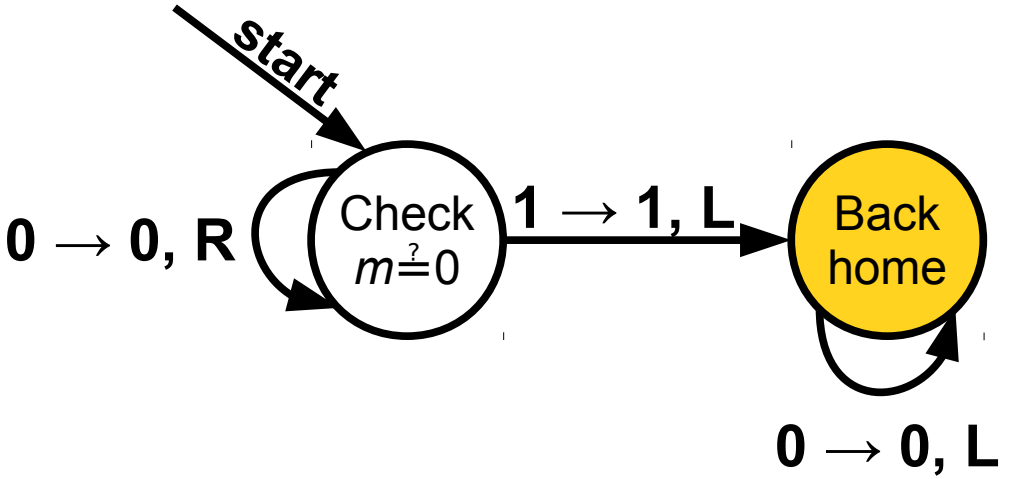

...
0
0
1
1
1
1
1
1
...

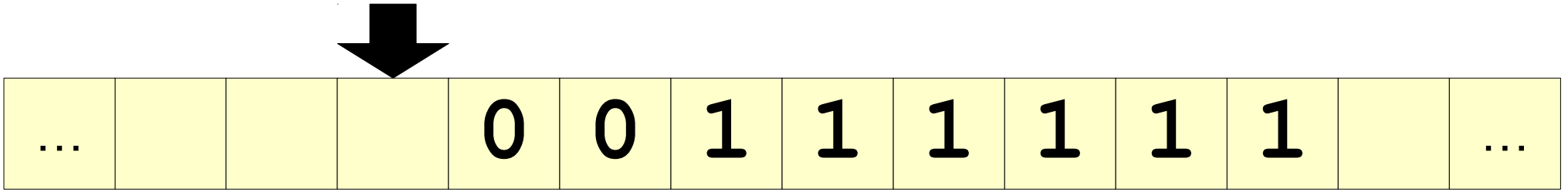

start
Check
$m \stackrel{?}{=} 0$
Back
home
0 → 0, R
1 → 1, L
0 → 0, L

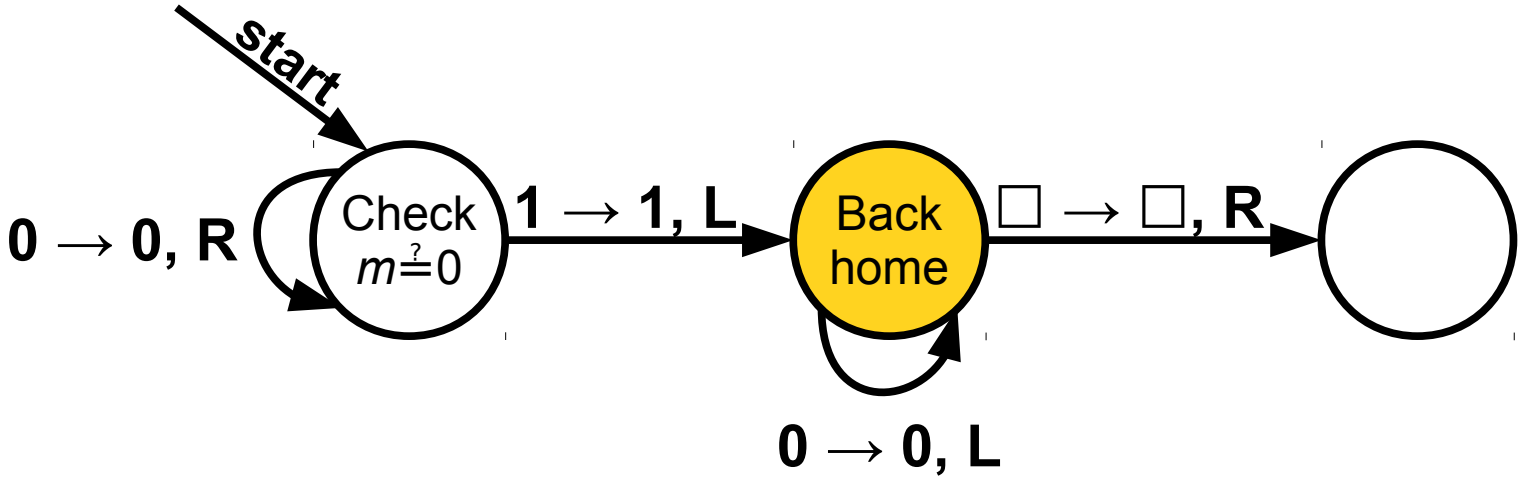

…
0
0
1
1
1
1
1
1
…

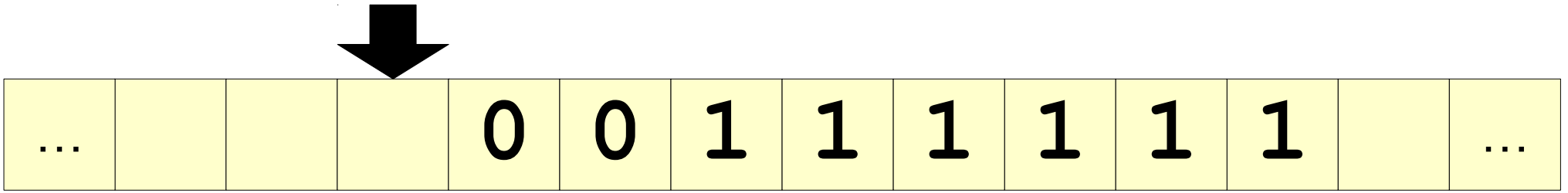

start
Check
$m \stackrel{?}{=} 0$
Back
home
0 → 0, R
1 → 1, L
0 → 0, L

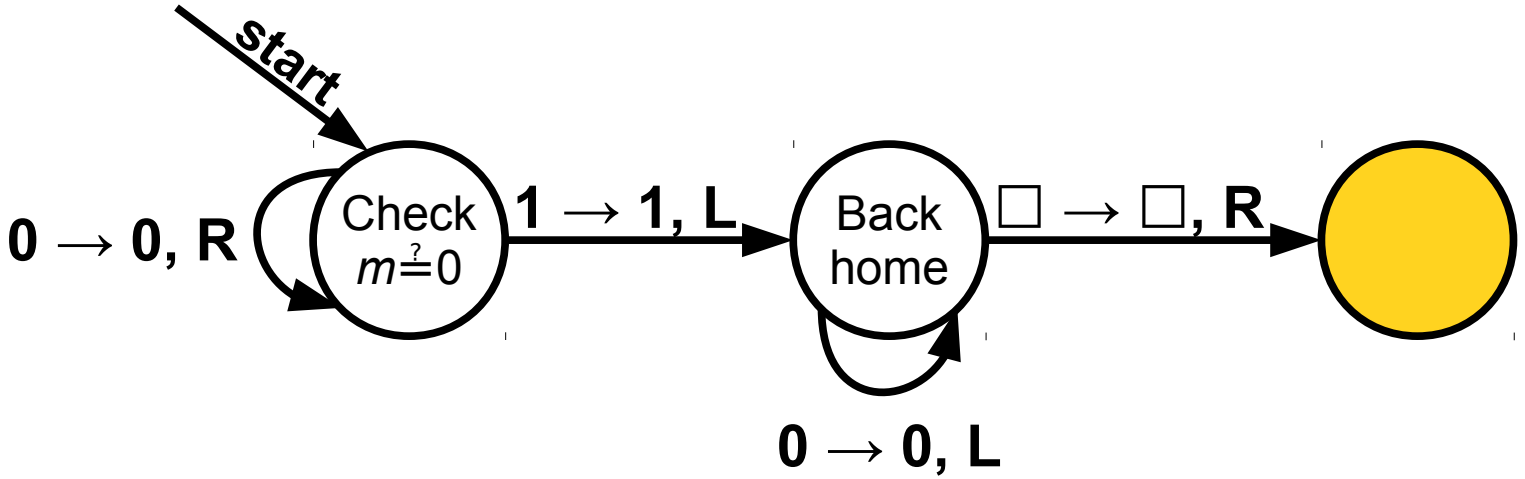

…
0
0
1
1
1
1
1
1
…

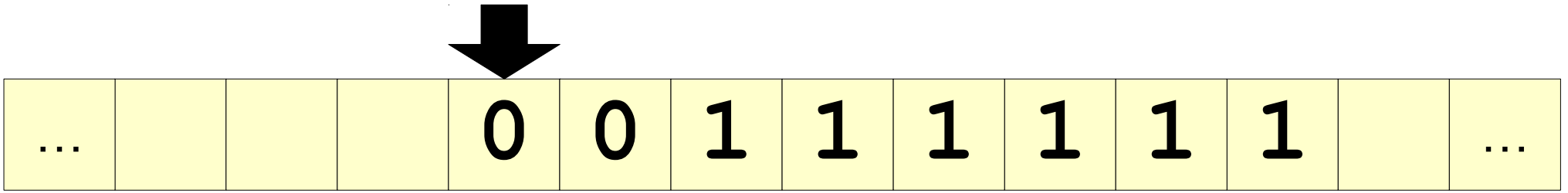

start
Check
$m \stackrel{?}{=} 0$
Back
home
0 → 0, R
1 → 1, L
0 → 0, L

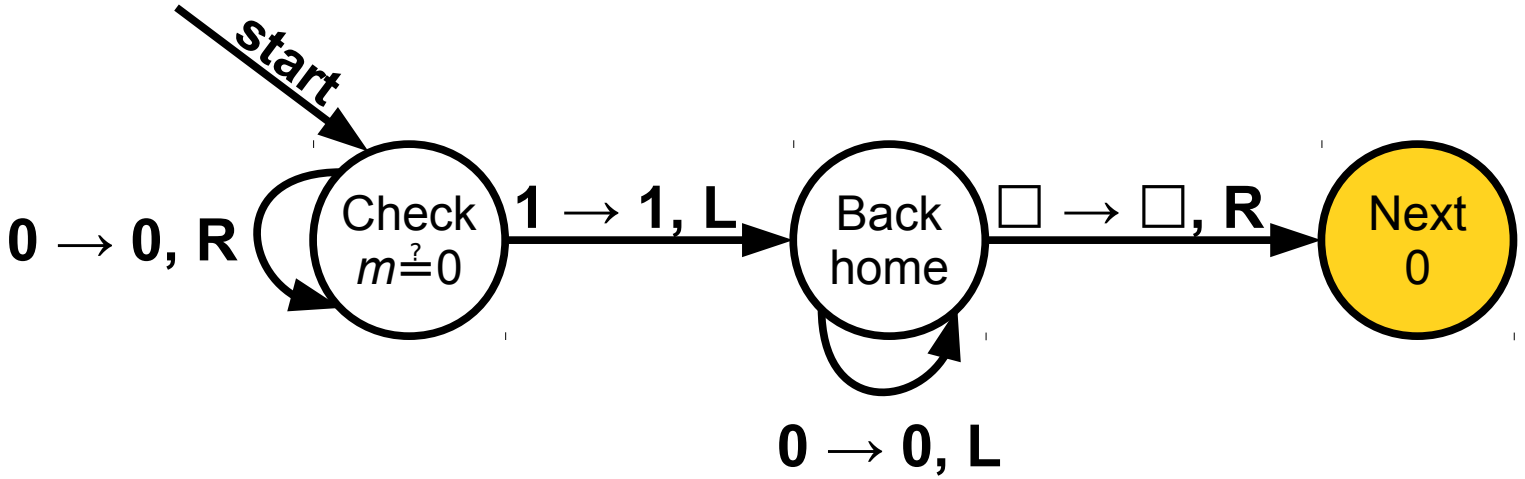

…
0
0
1
1
1
1
1
1
…

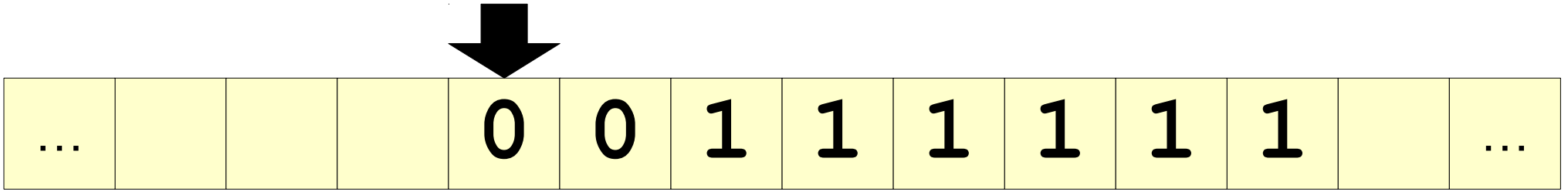

start
Check
$m \stackrel{?}{=} 0$
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L

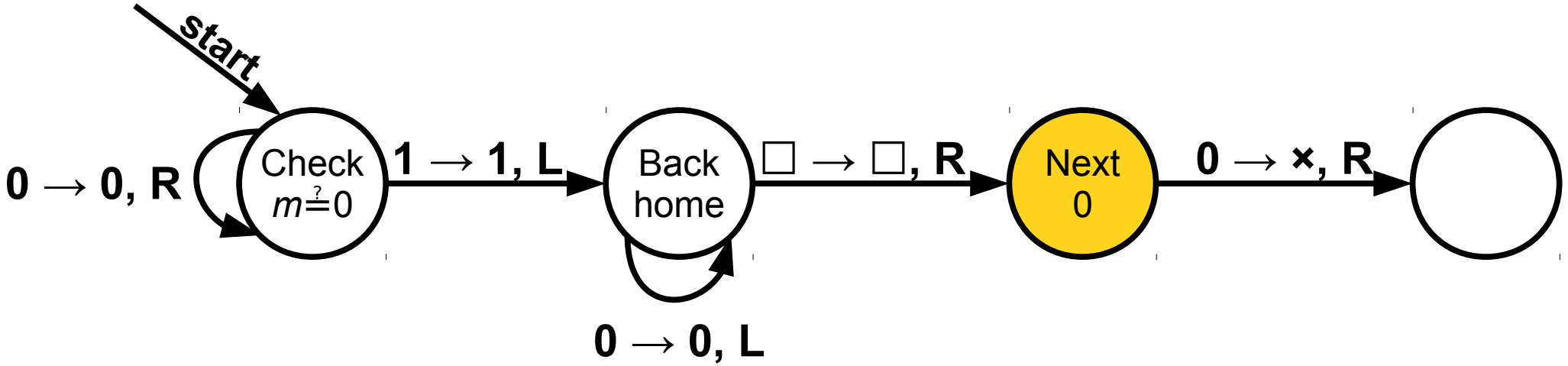

…
0
0
1
1
1
1
1
1
…

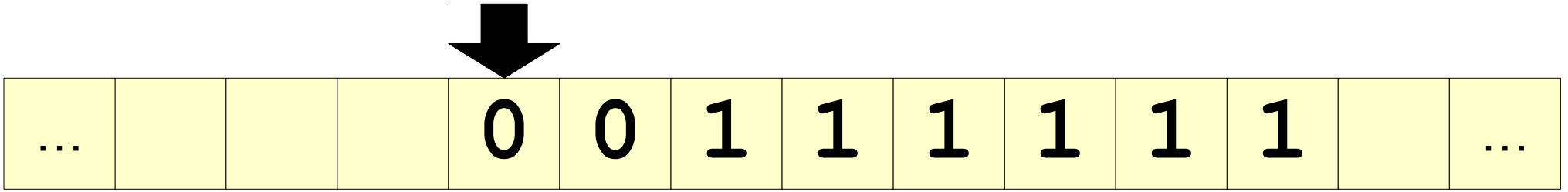

start
Check
$m \stackrel{?}{=} 0$
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L

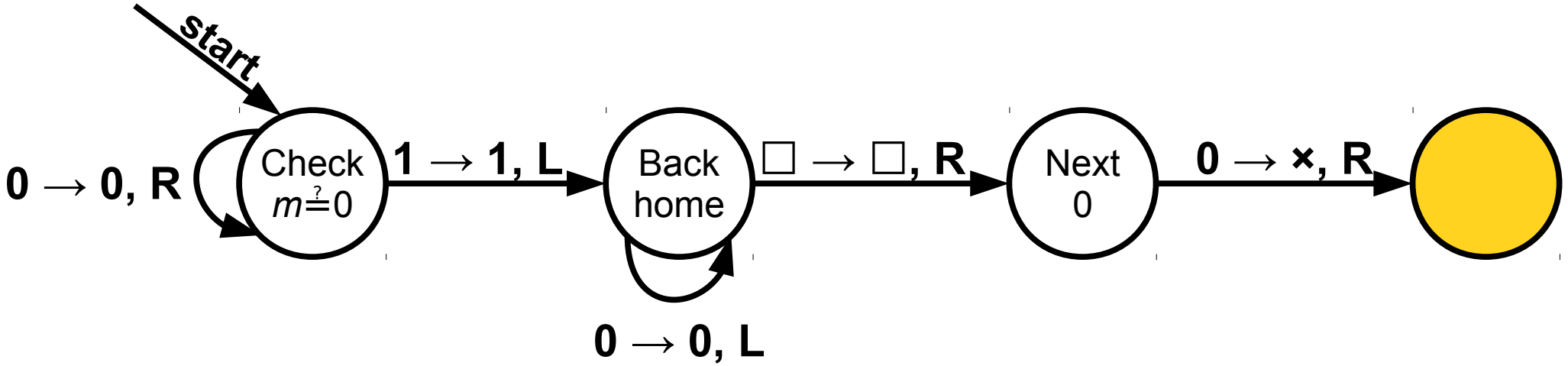

…
0
0
1
1
1
1
1
1
…

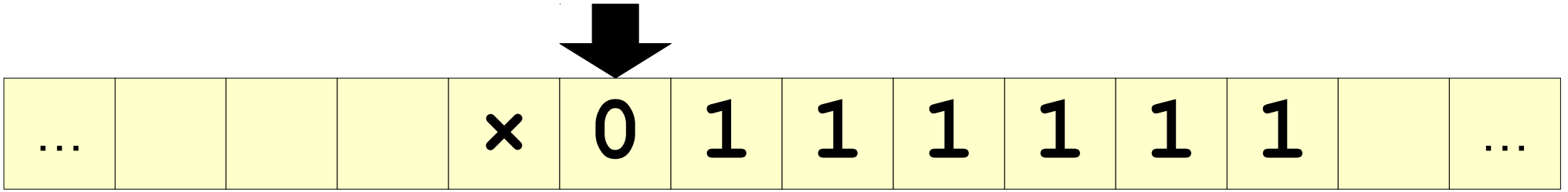

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L

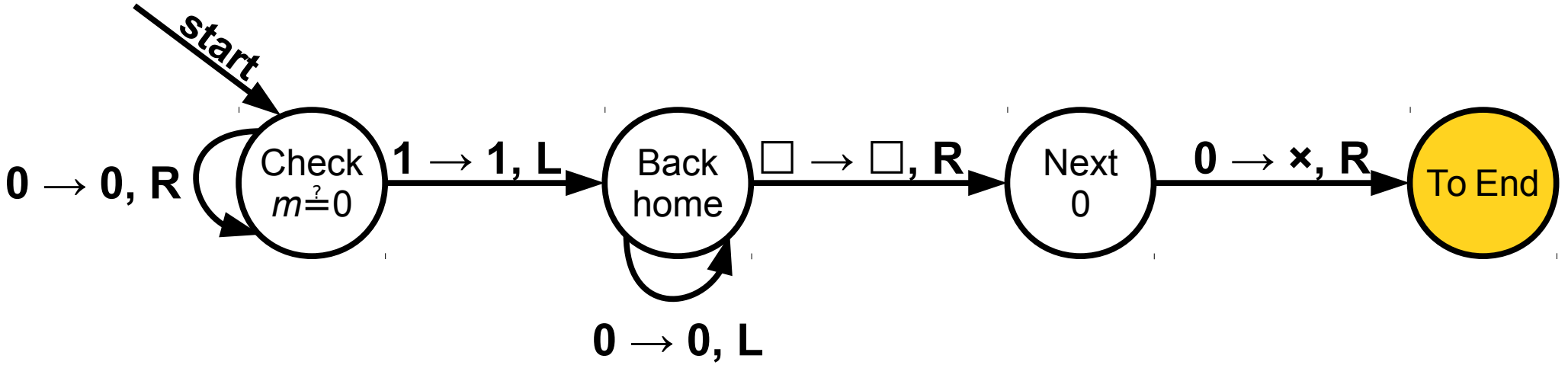

…
0
0
1
1
1
1
1
1
…

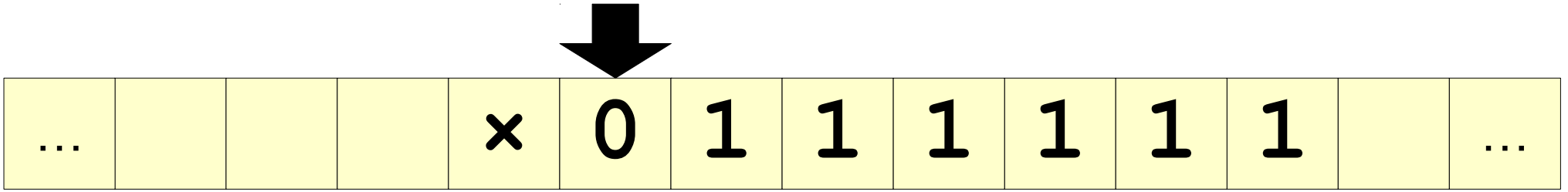

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
0 → 0, L

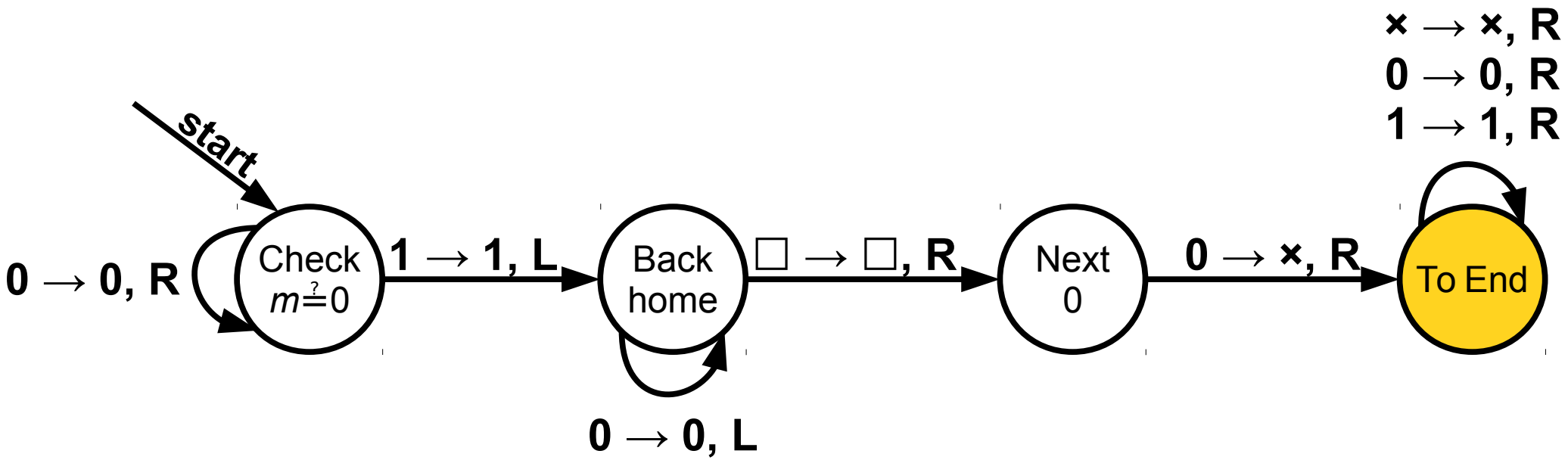

…
0
0
1
1
1
1
1
1
…

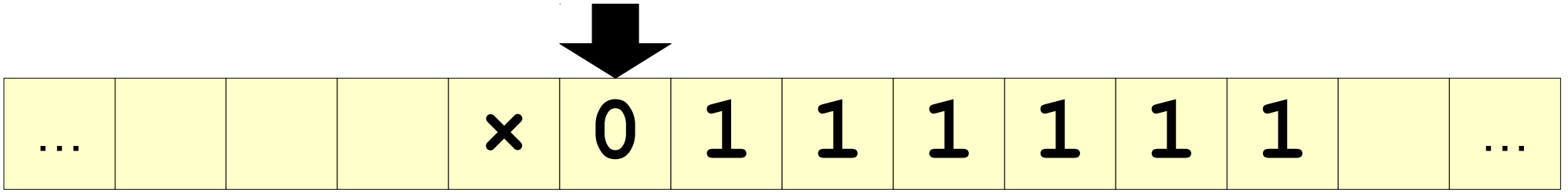

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
0 → 0, L

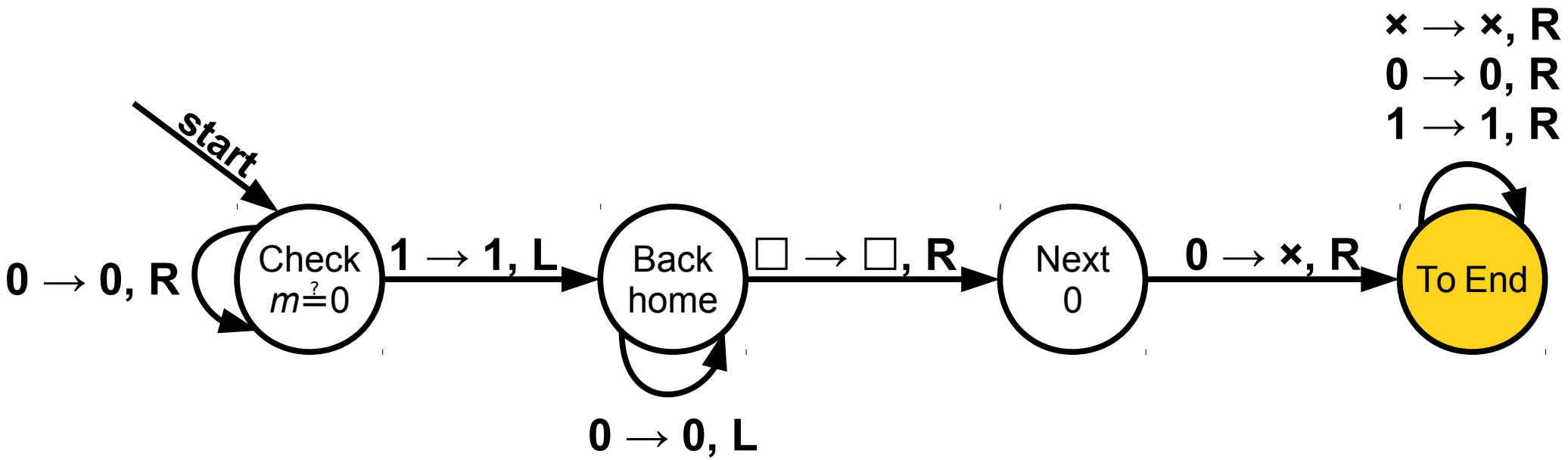

…
×
0
1
1
1
1
1
1
…

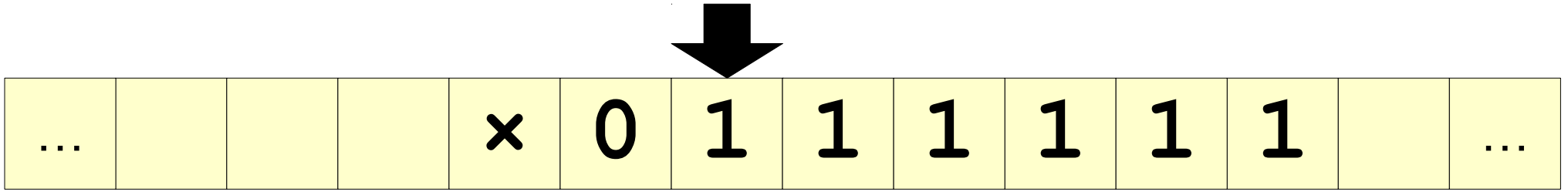

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
0 → 0, L

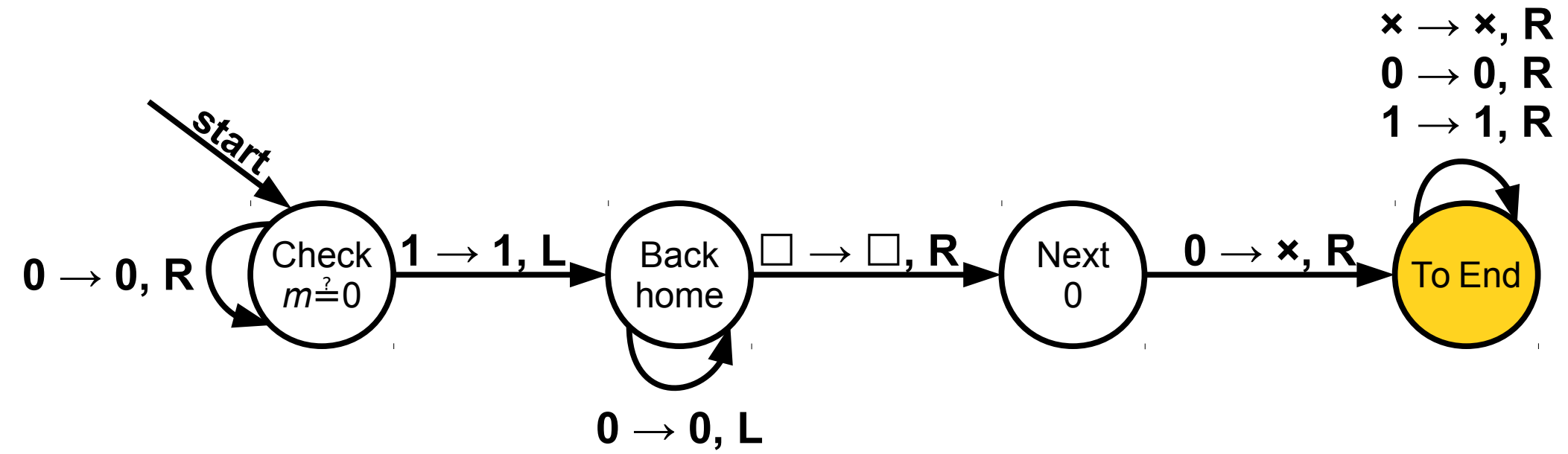

…
×
0
1
1
1
1
1
1
…

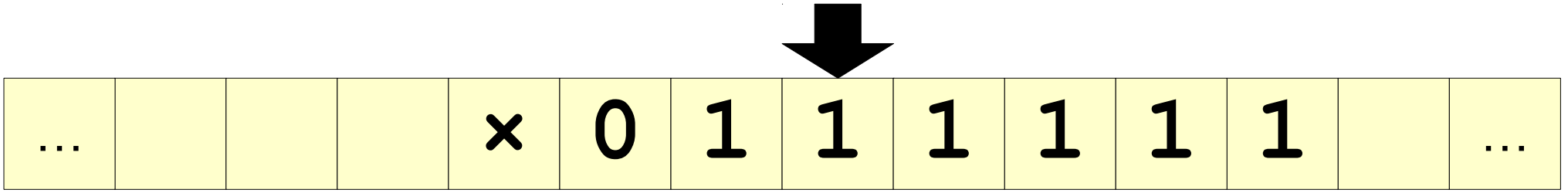

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

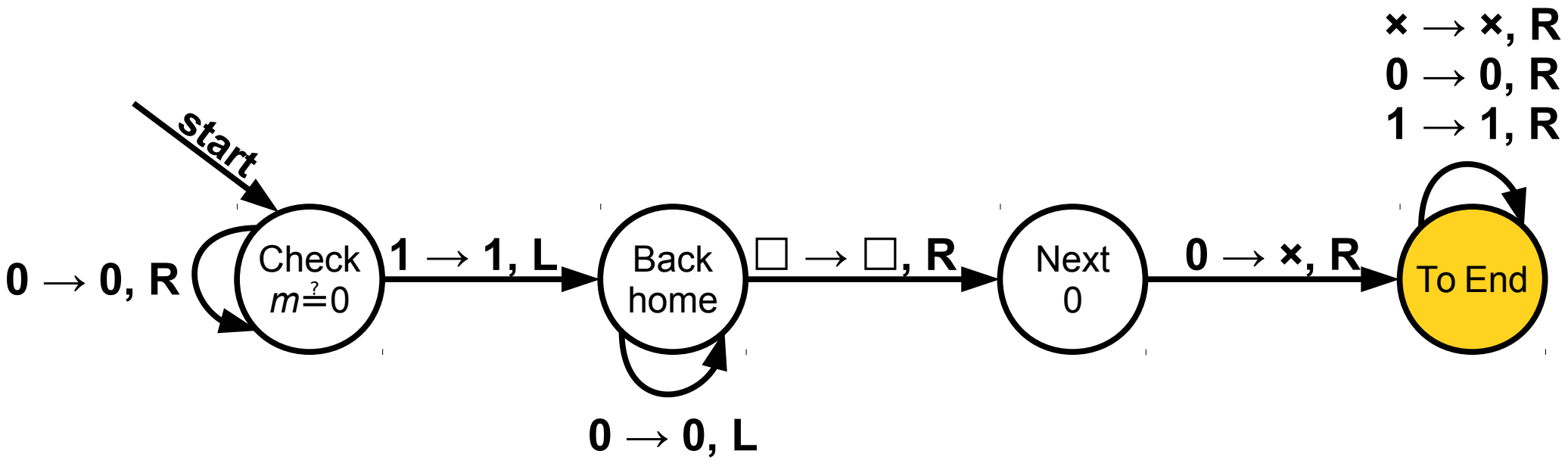

…
×
0
1
1
1
1
1
1
…

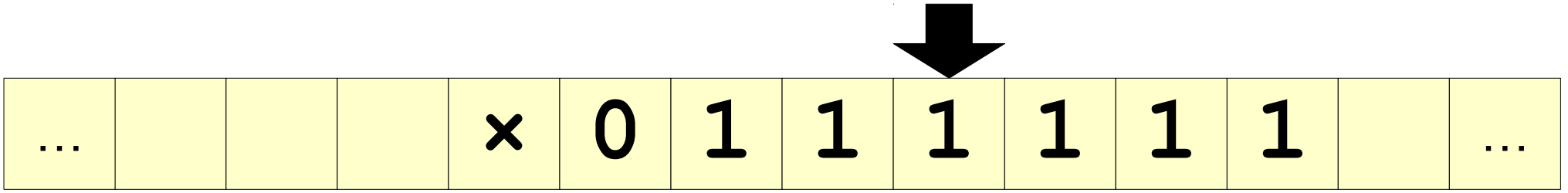

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

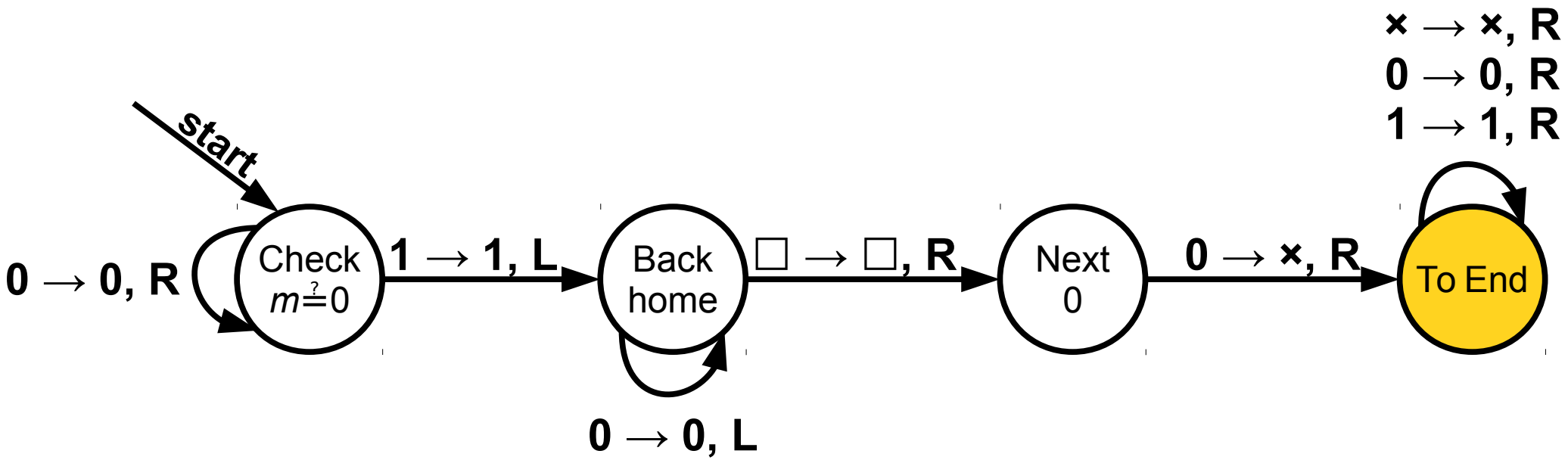

…
×
0
1
1
1
1
1
1
…

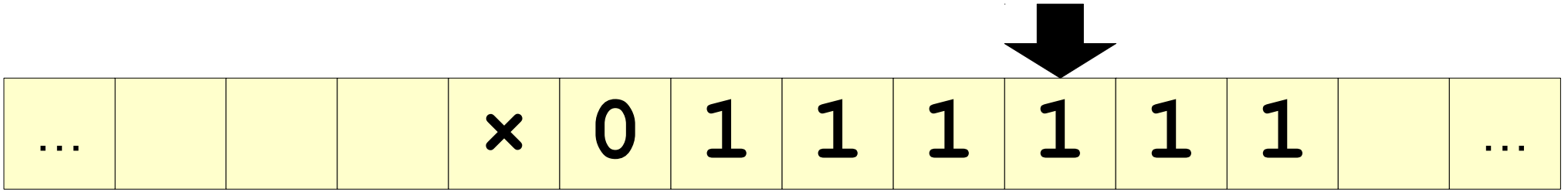

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

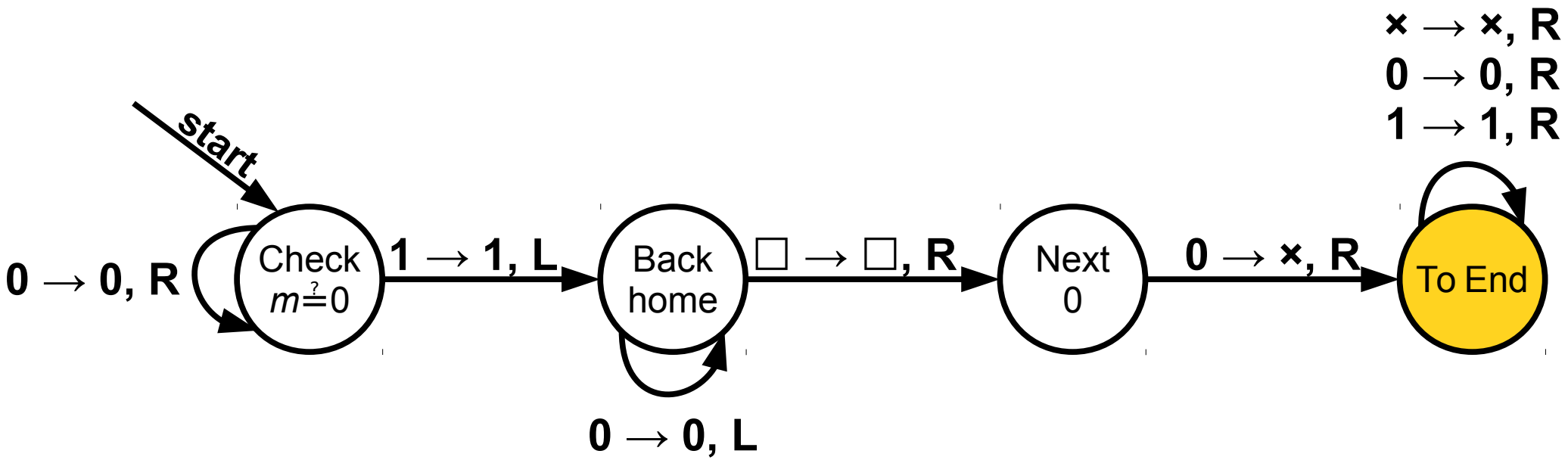

…
×
0
1
1
1
1
1
1
…

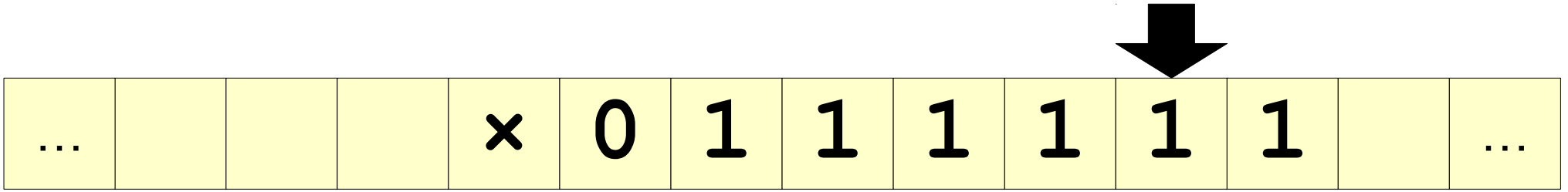

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

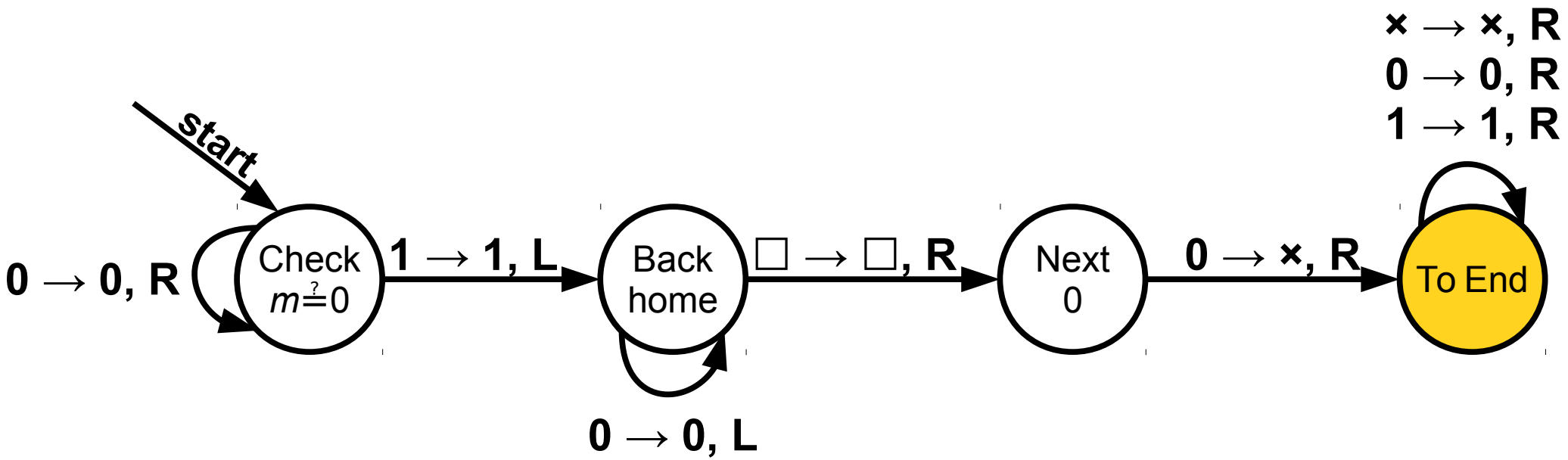

…
×
0
1
1
1
1
1
1
…

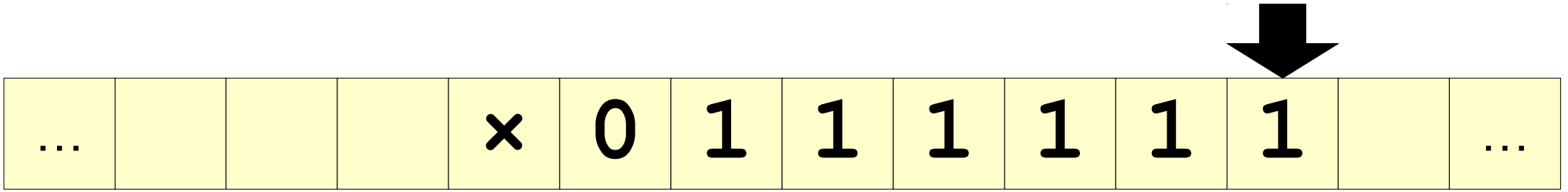

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

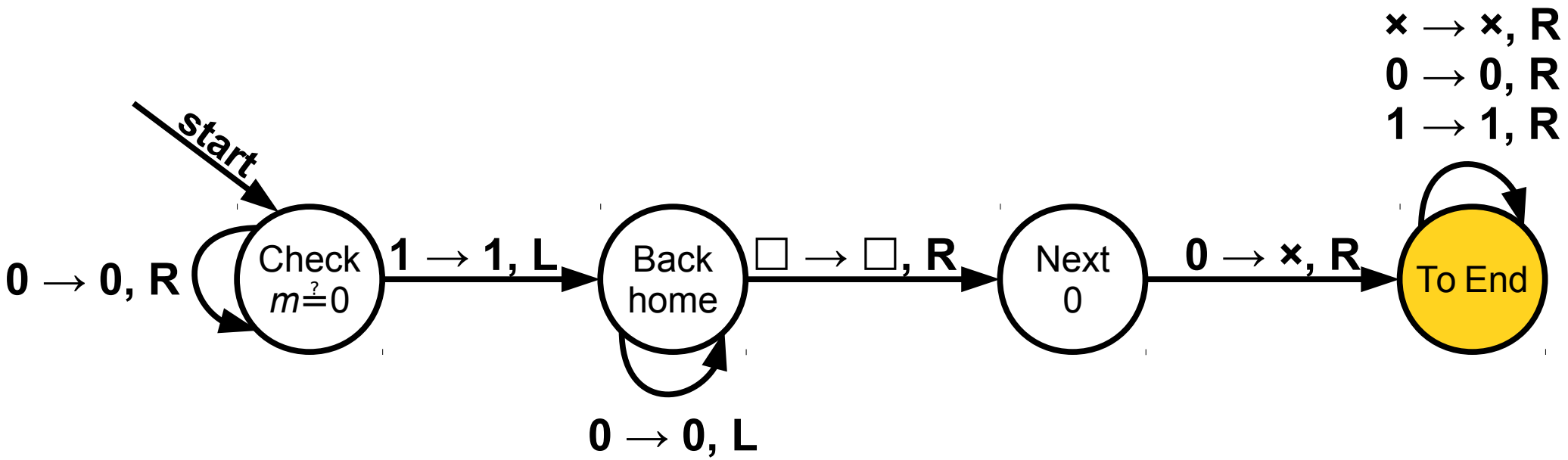

…
×
0
1
1
1
1
1
1
…

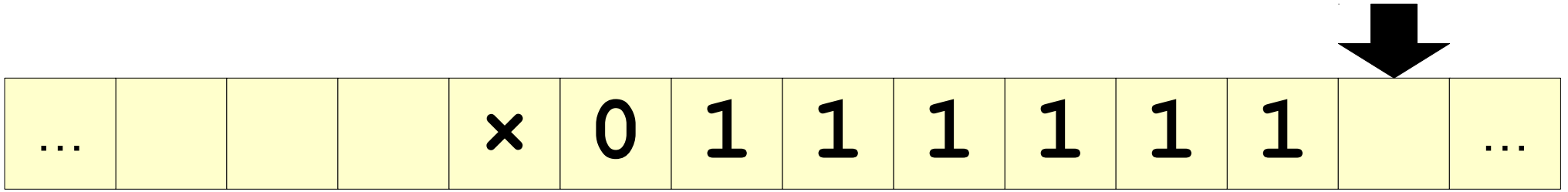

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

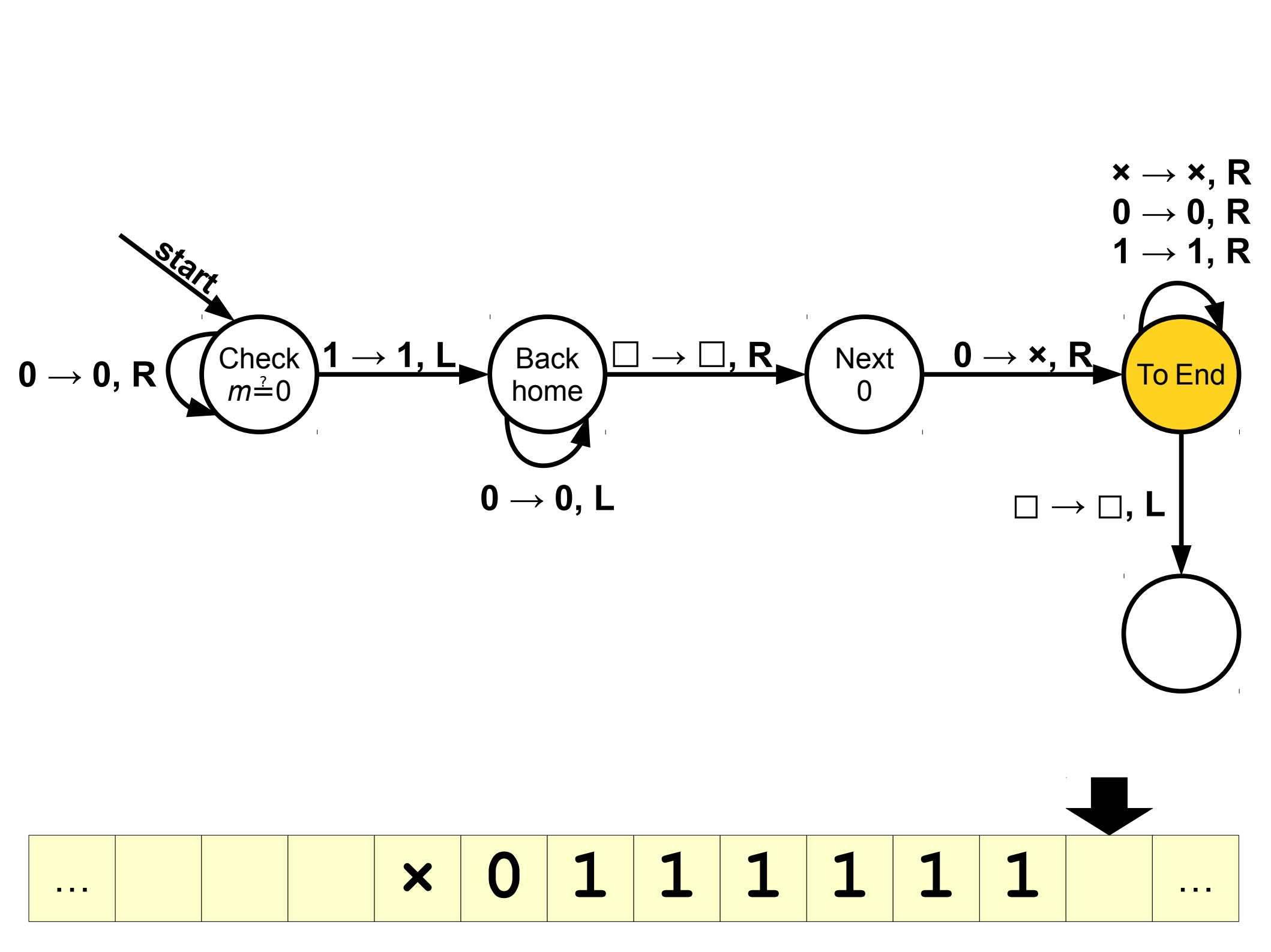

…
×
0
1
1
1
1
1
1
…

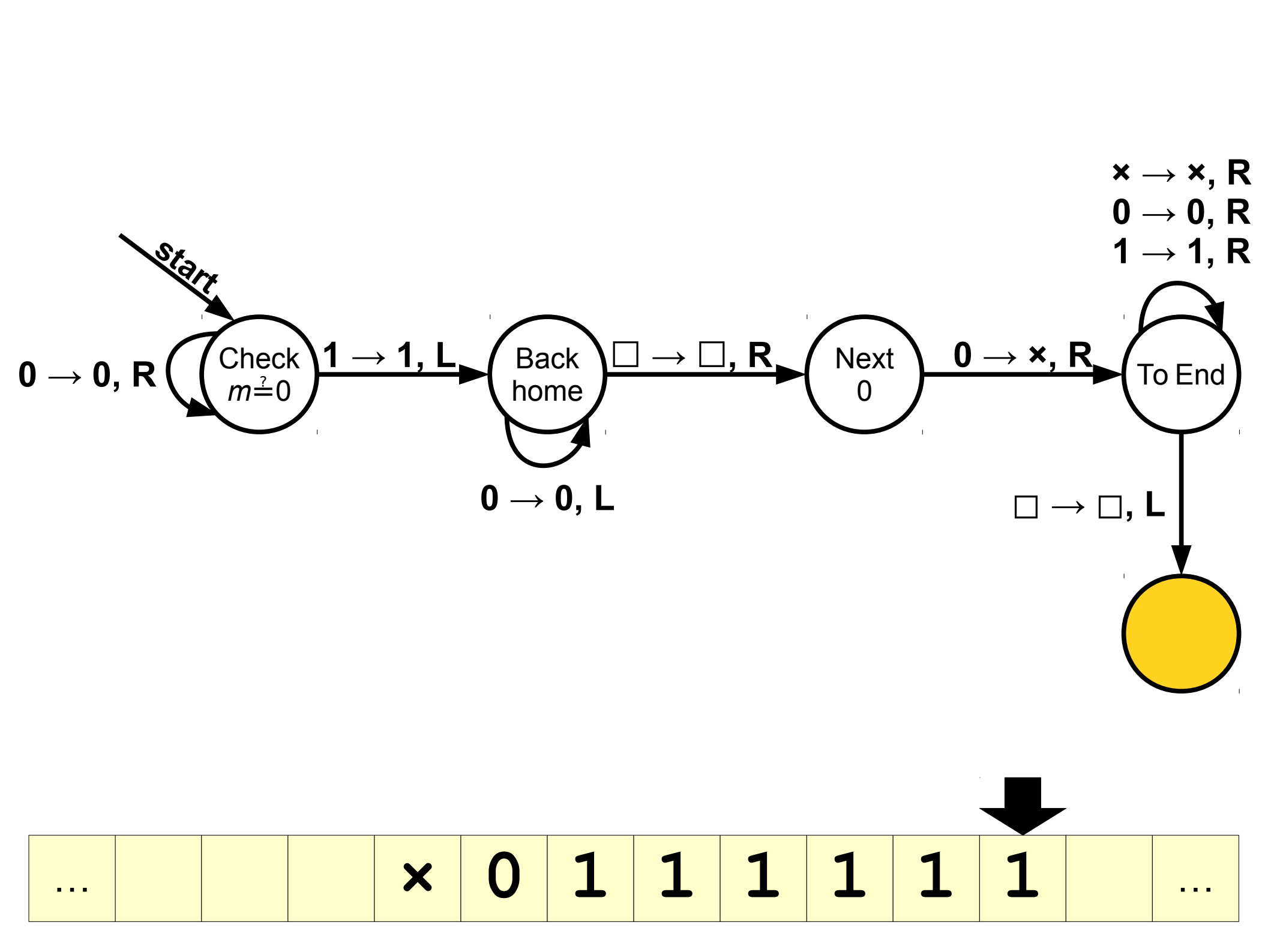

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

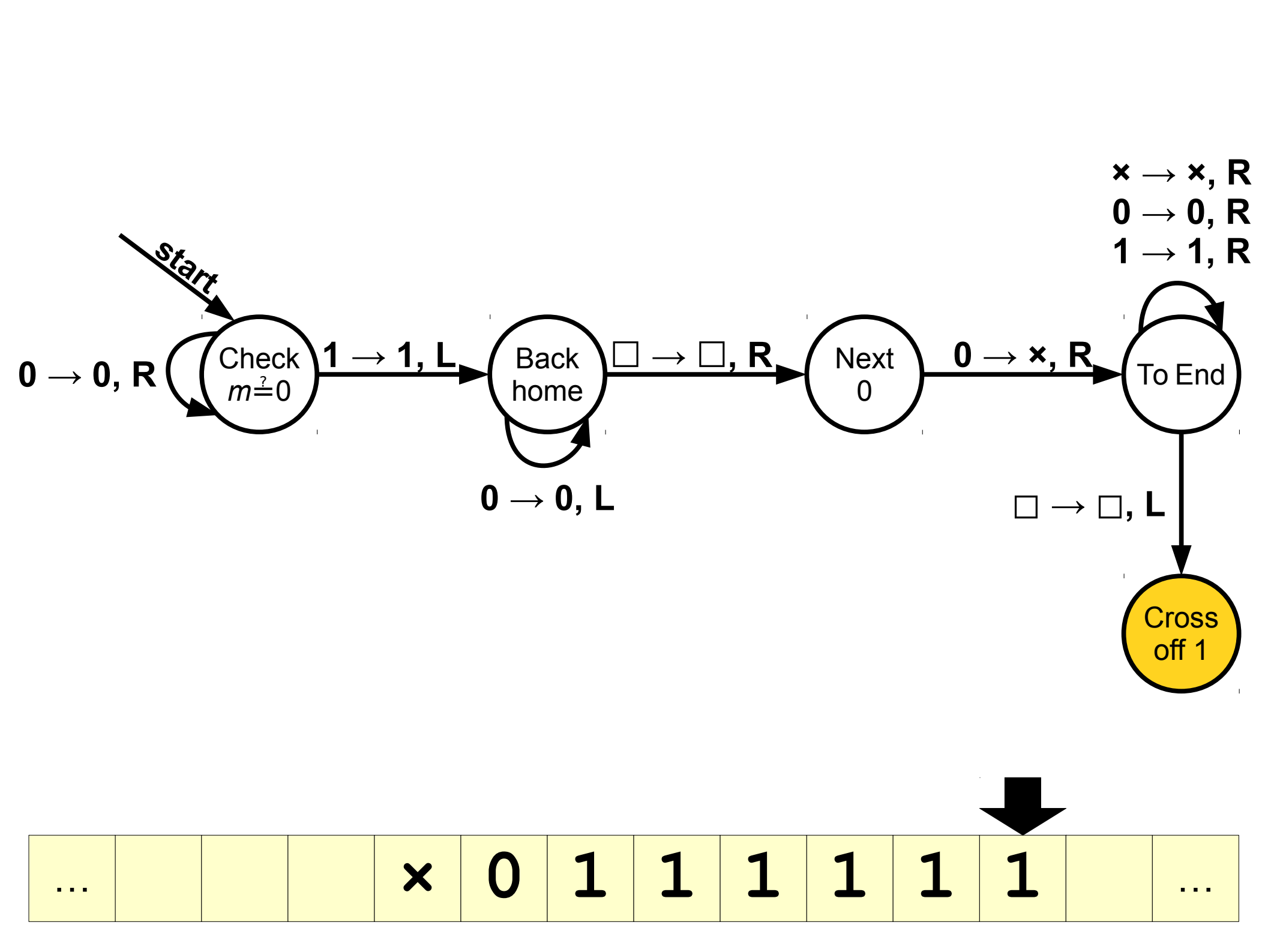

…
×
0
1
1
1
1
1
1
…

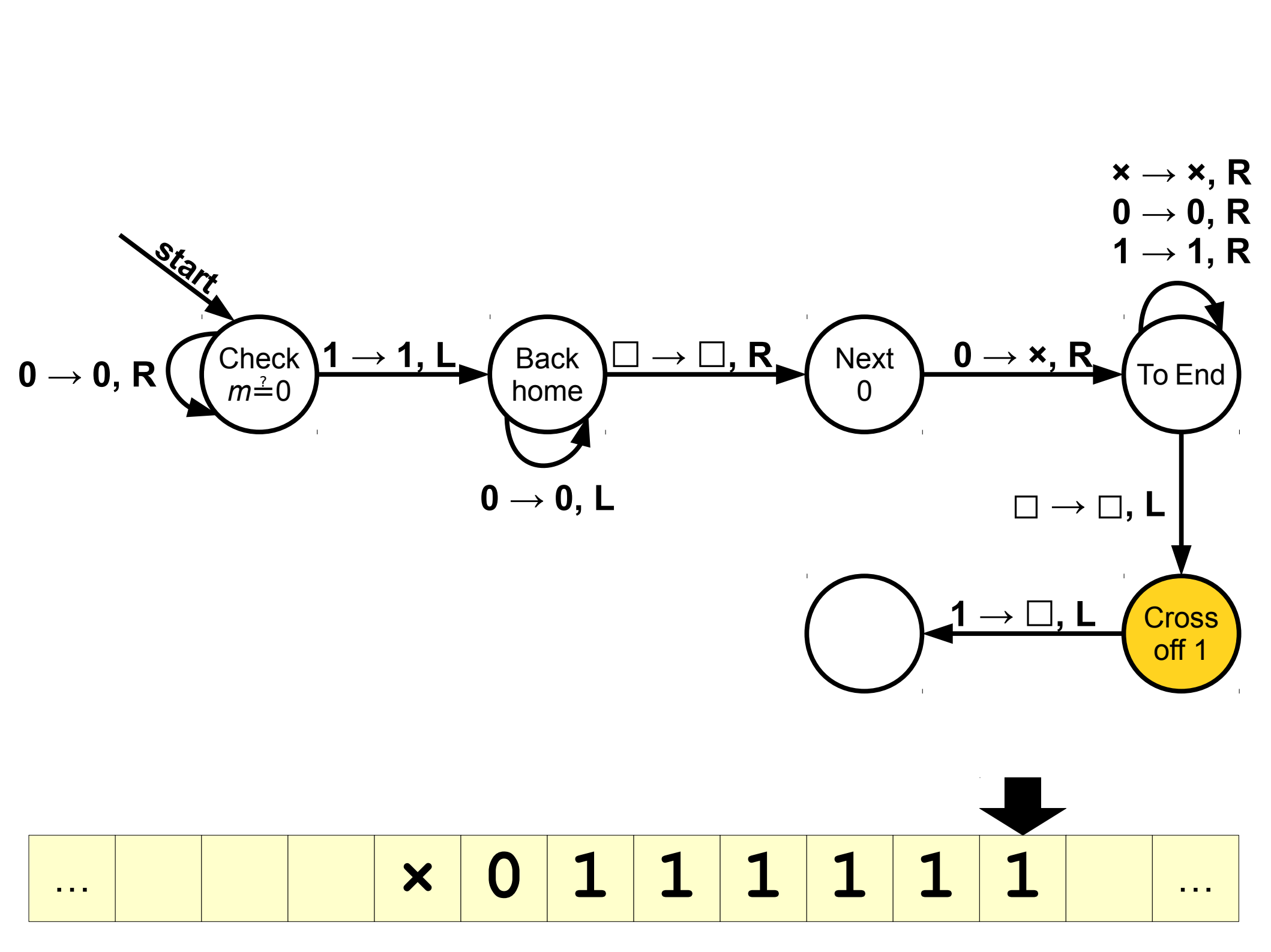

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L

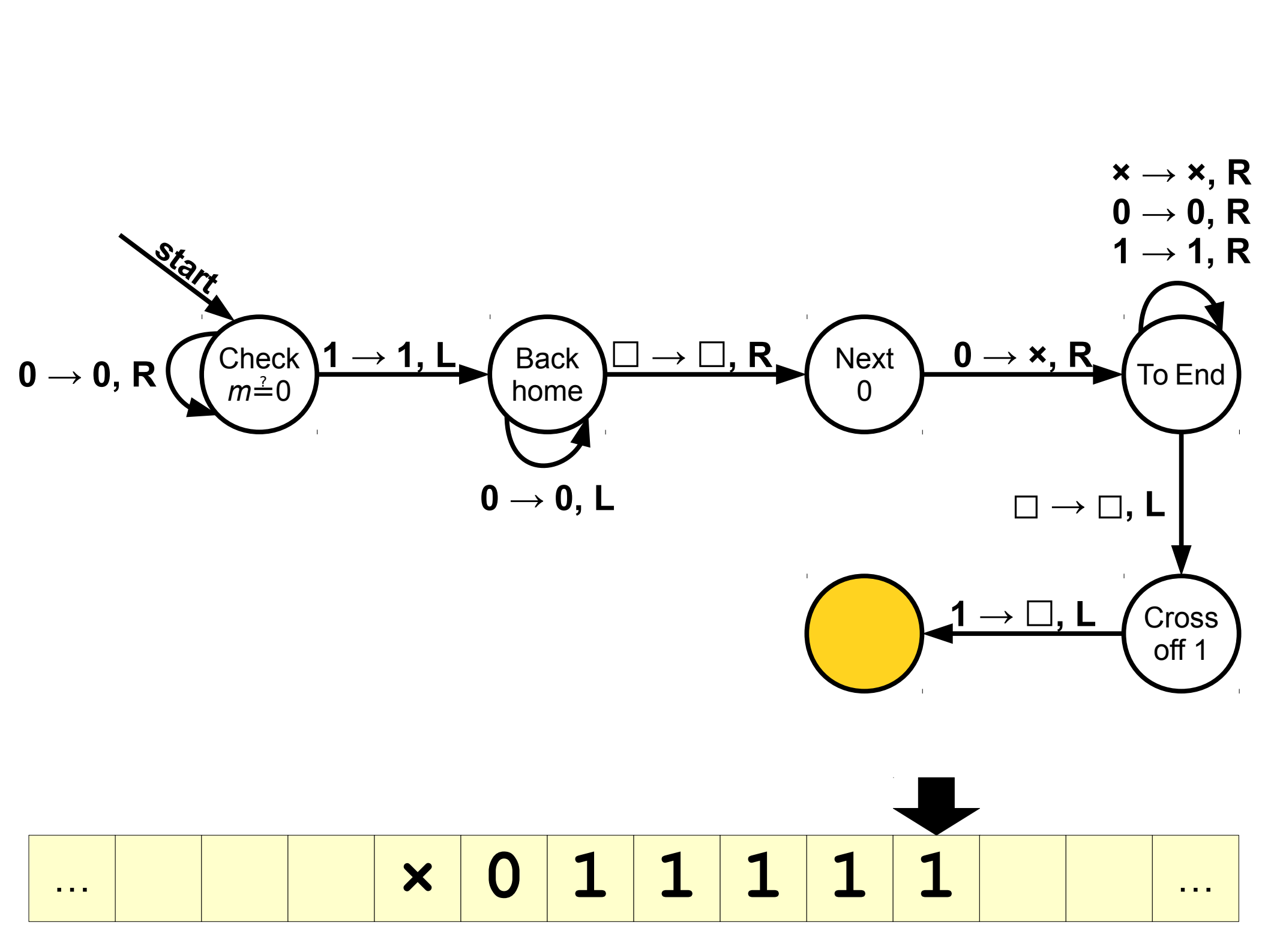

…
×
0
1
1
1
1
1
1
…

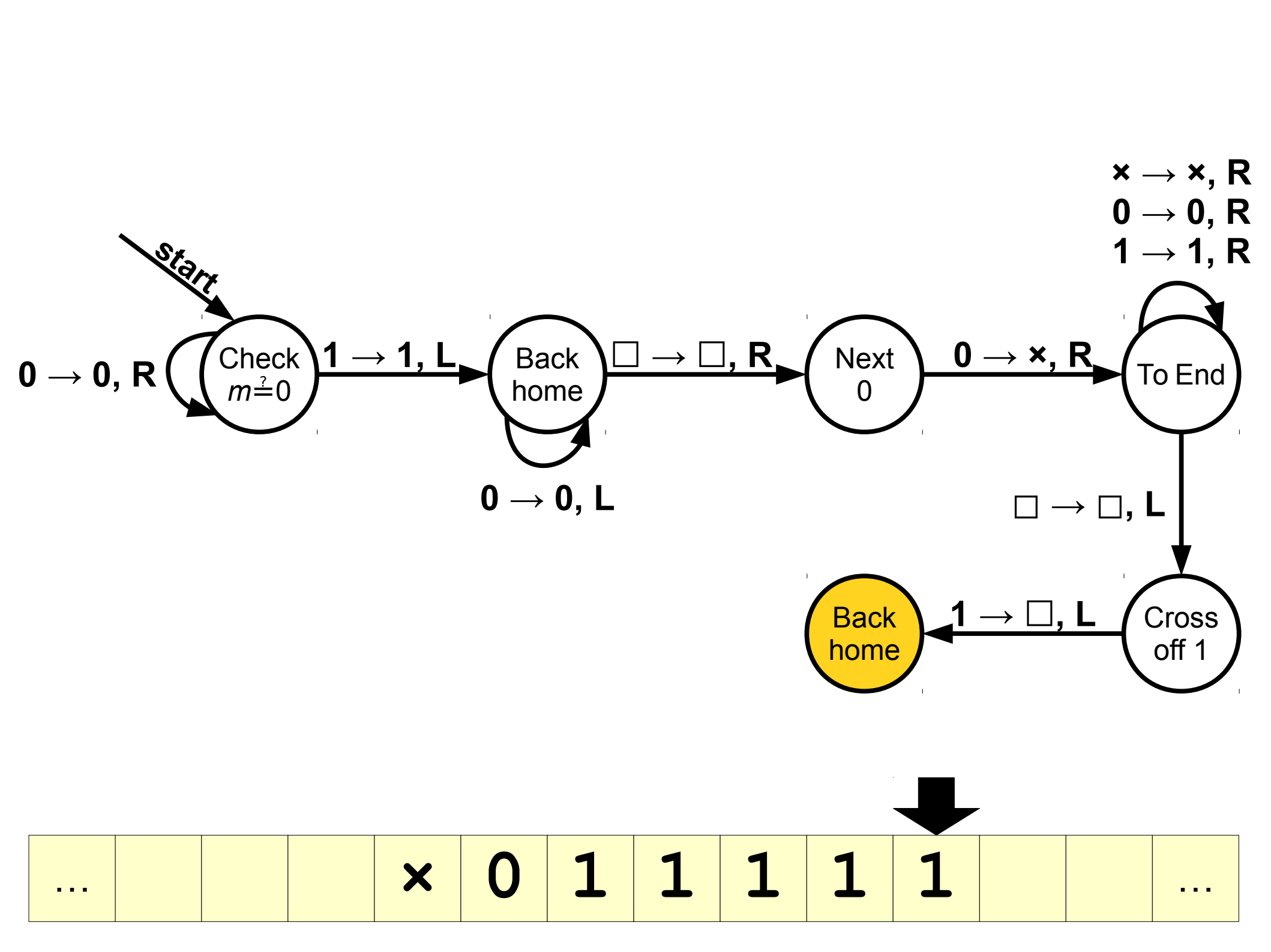

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
…
×
0
1
1
1
1
1
1
…

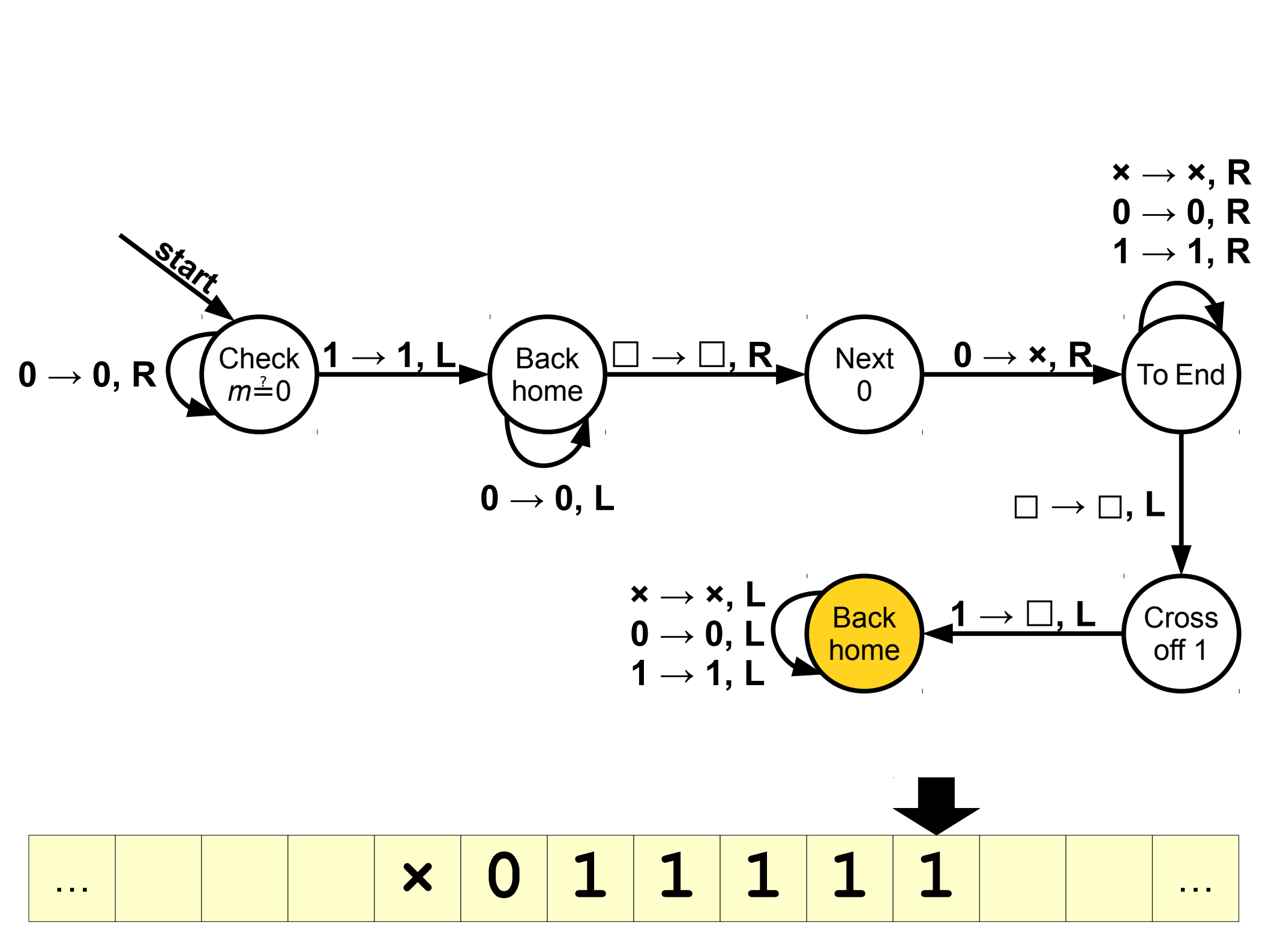

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
…
×
0
1
1
1
1
1
1
…

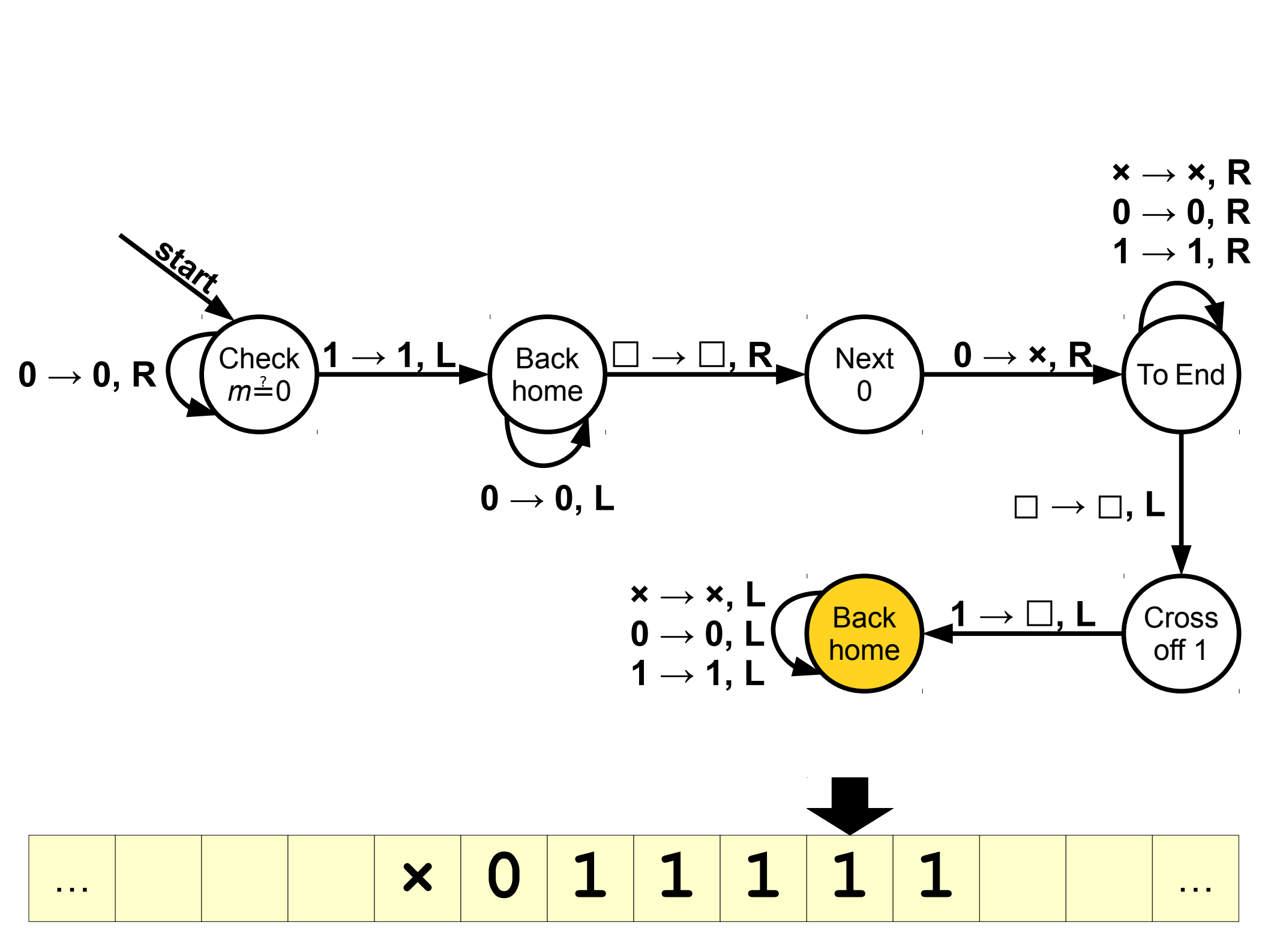

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
…
×
0
1
1
1
1
1
1
…

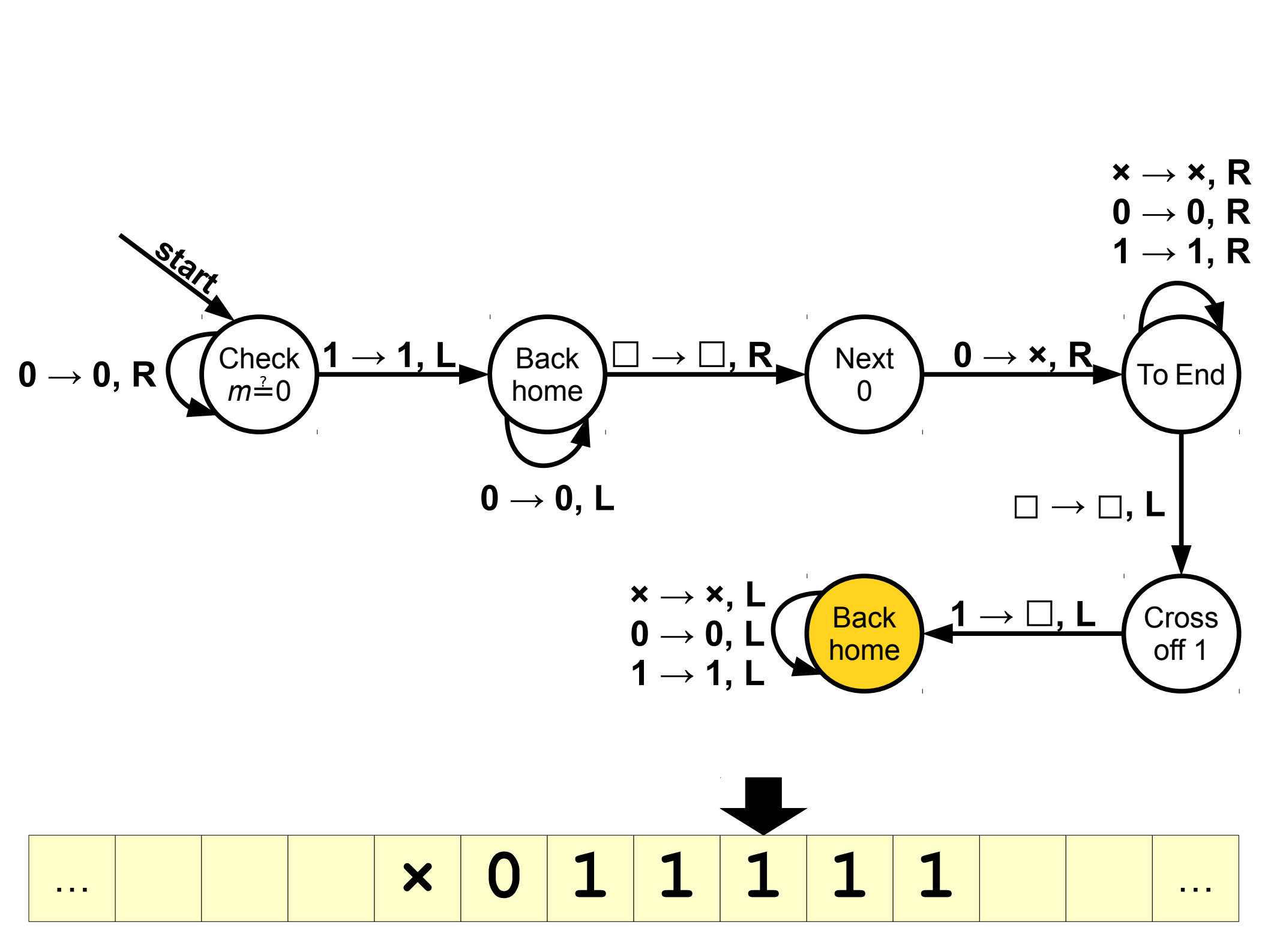

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
… × 0 1 1 1 1 1 1 …

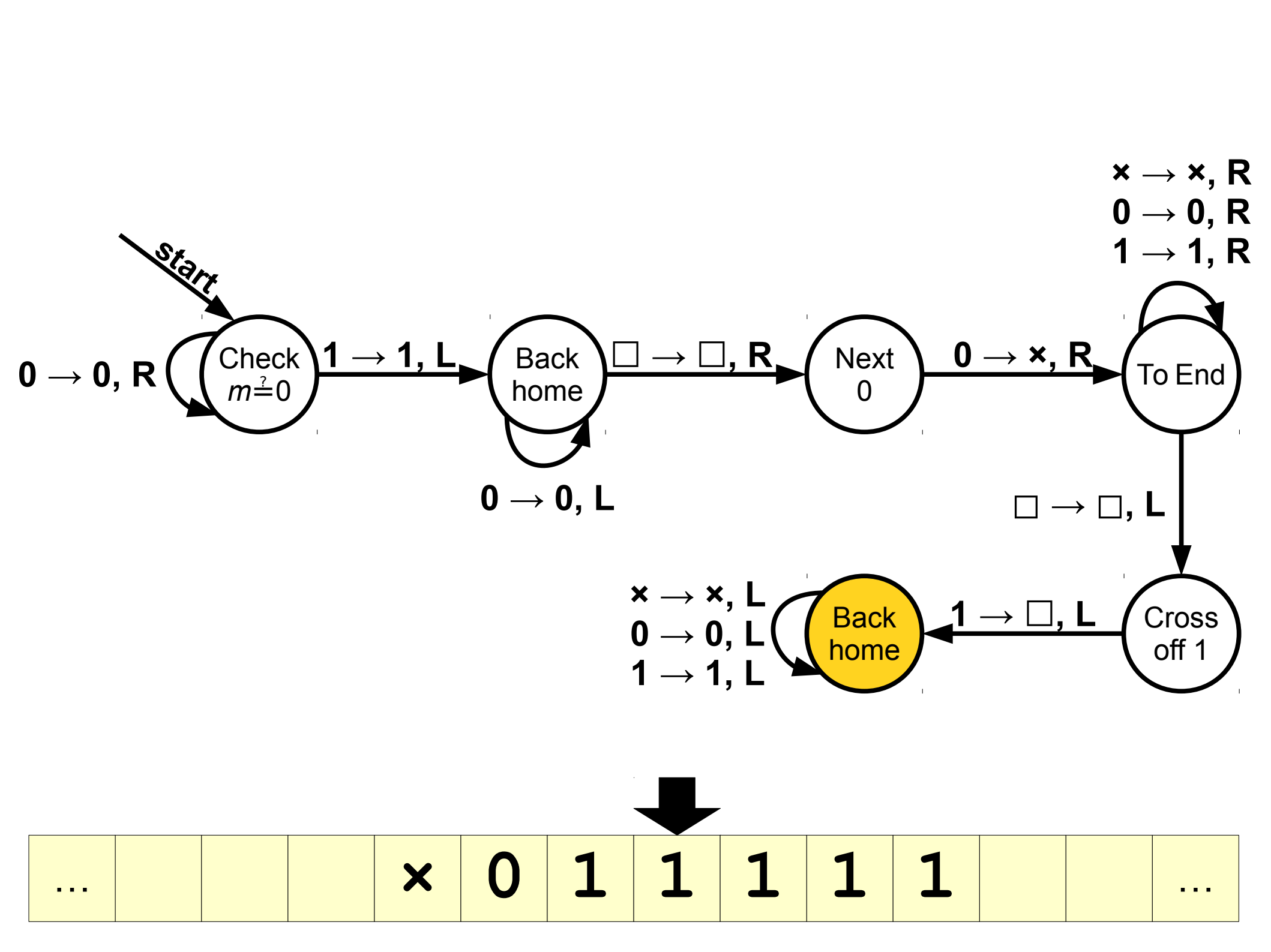

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
… × 0 1 1 1 1 1 …

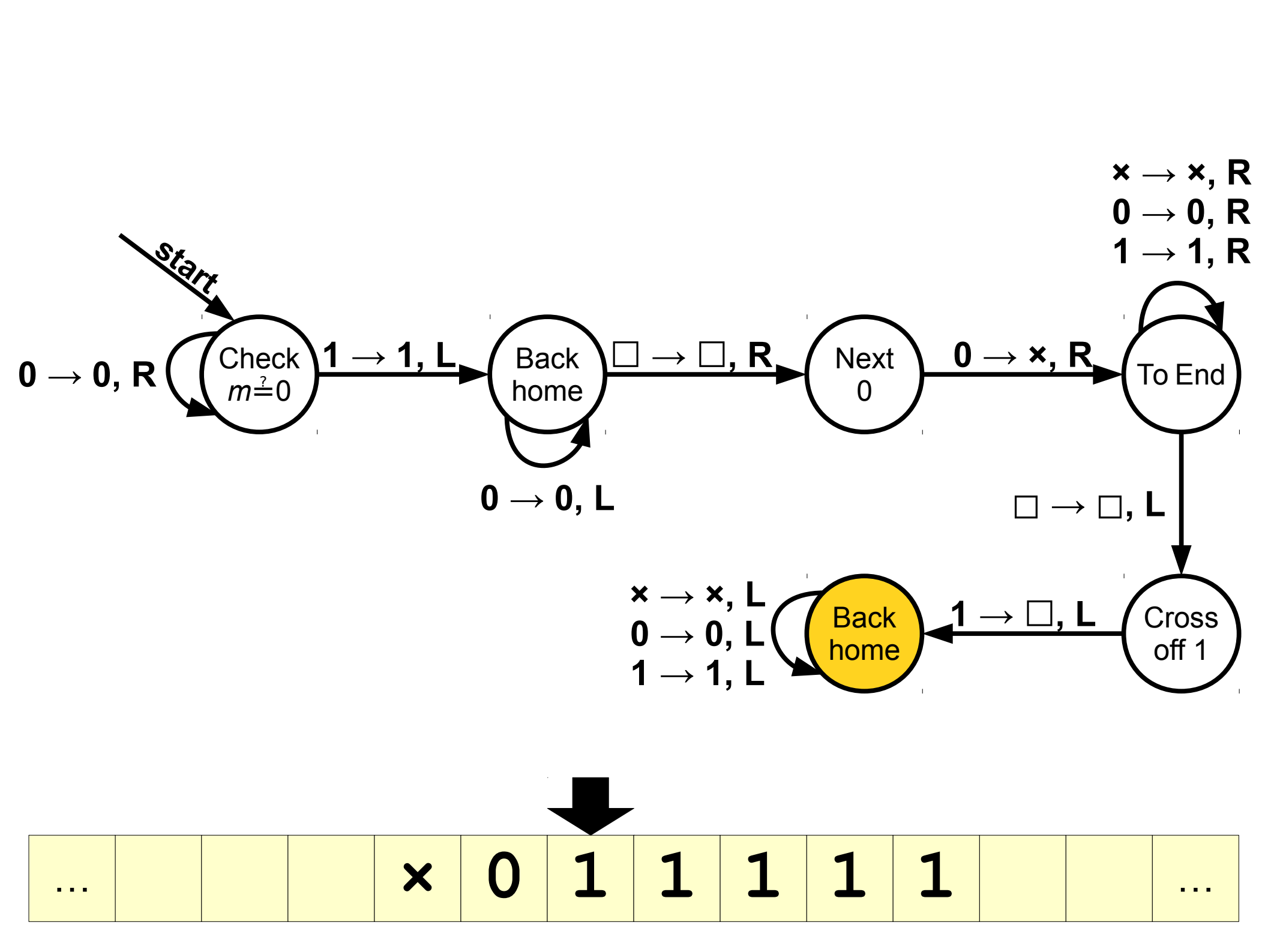

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
… × 0 1 1 1 1 1 …

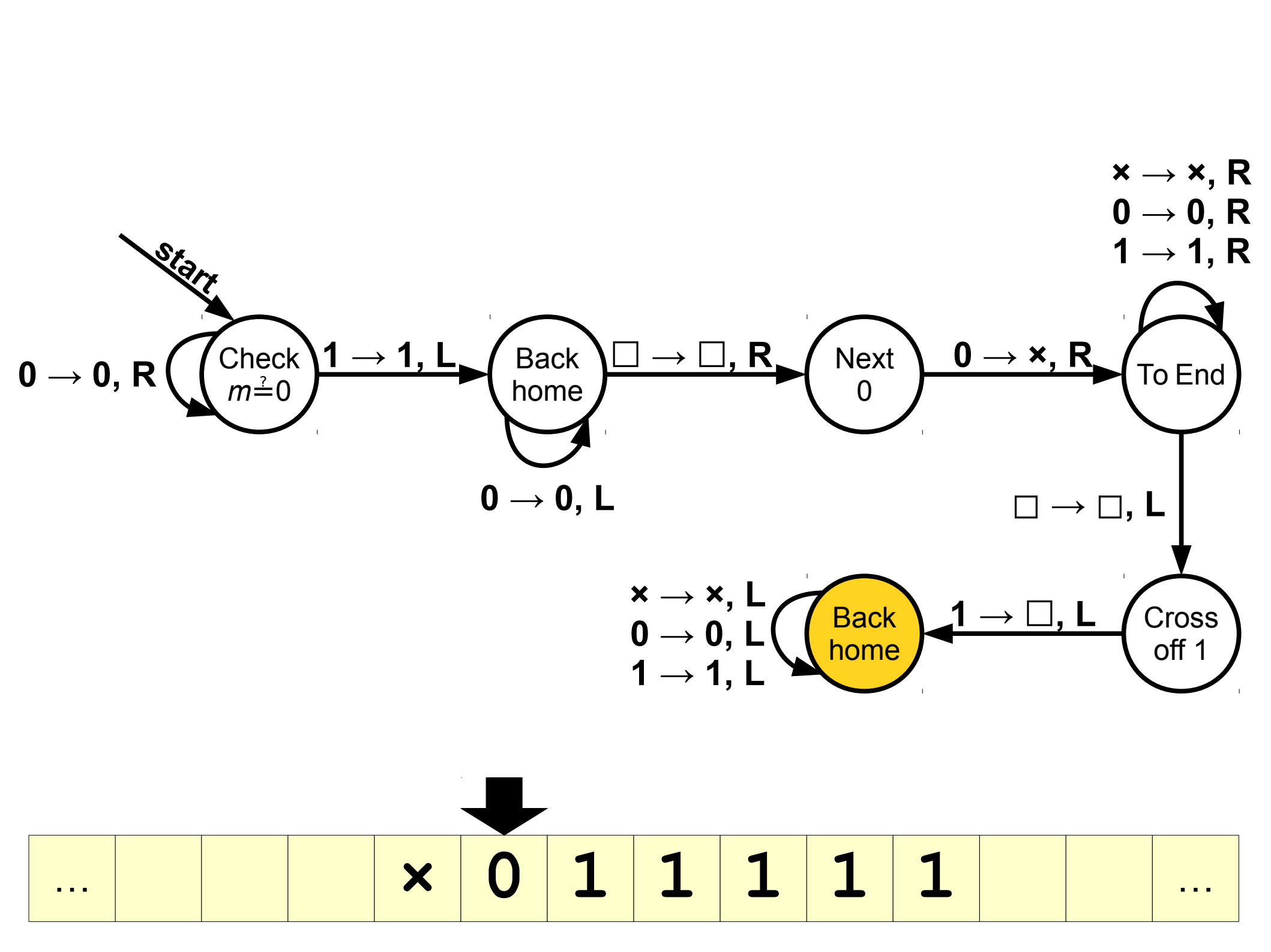

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
× → ×, L
0 → 0, L
1 → 1, L
…
×
0
1
1
1
1
1
…

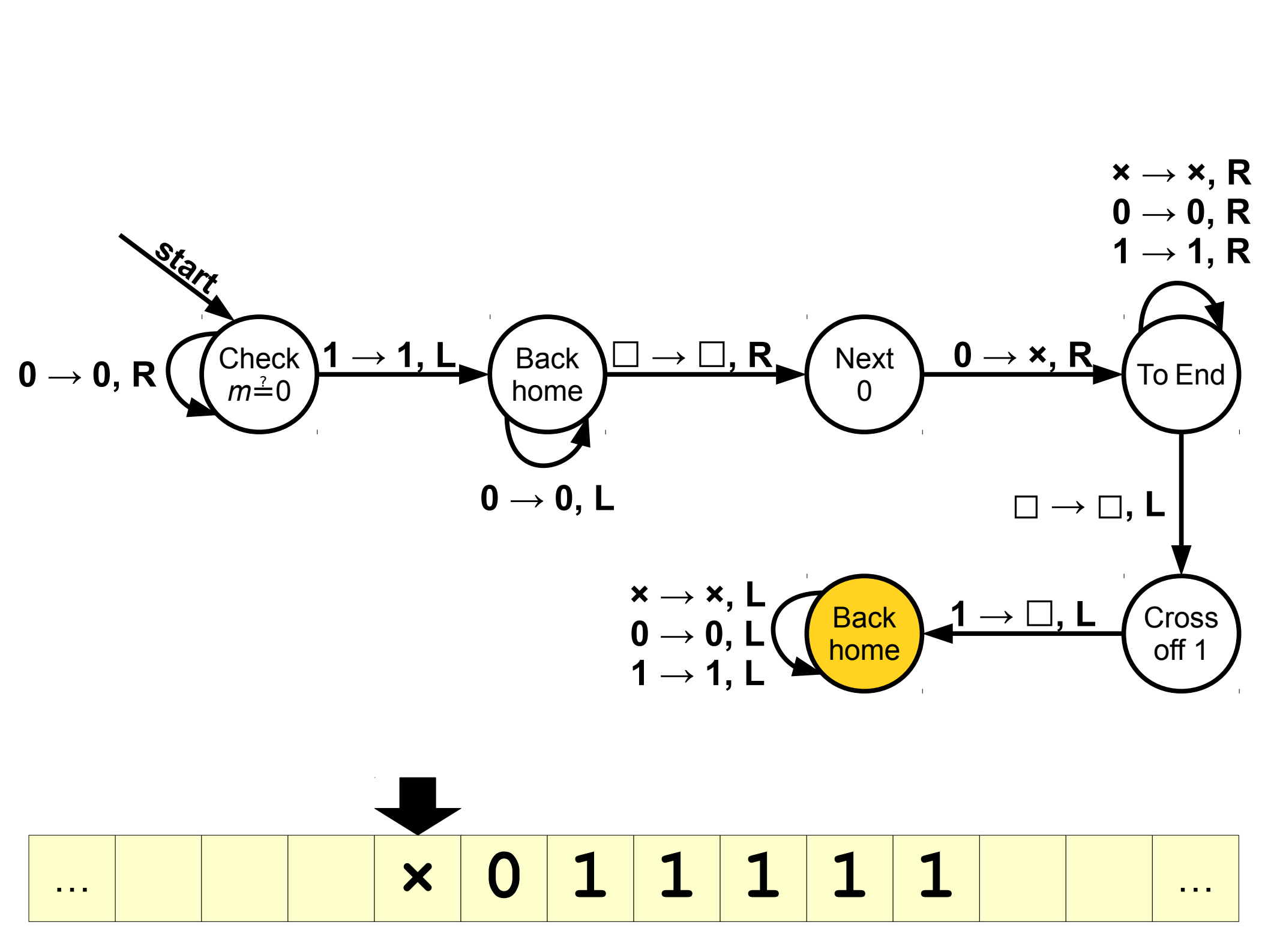

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
× → ×, L
0 → 0, L
1 → 1, L
… × 0 1 1 1 1 1 …

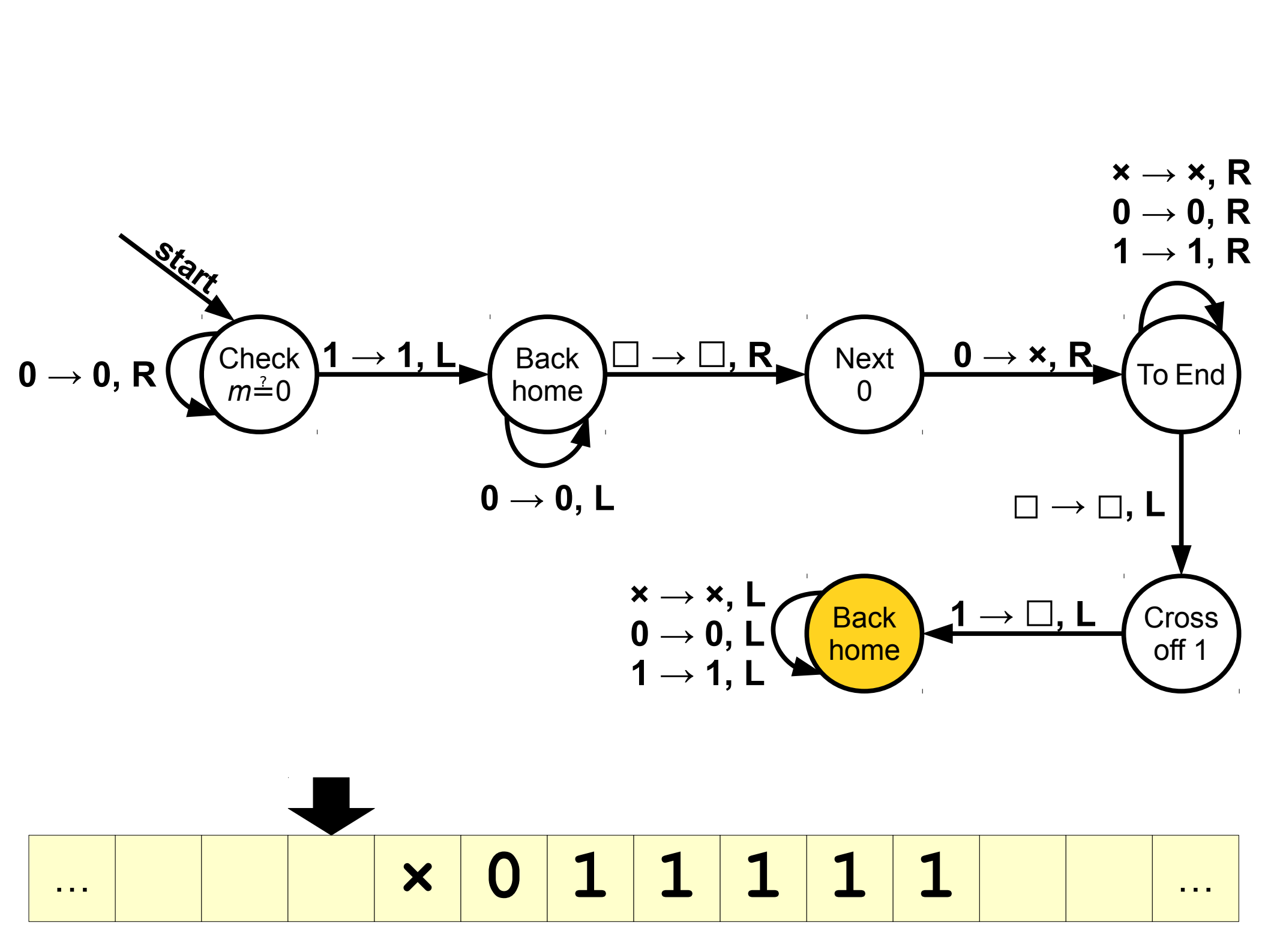

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
× → ×, L
0 → 0, L
1 → 1, L
… × 0 1 1 1 1 1 …

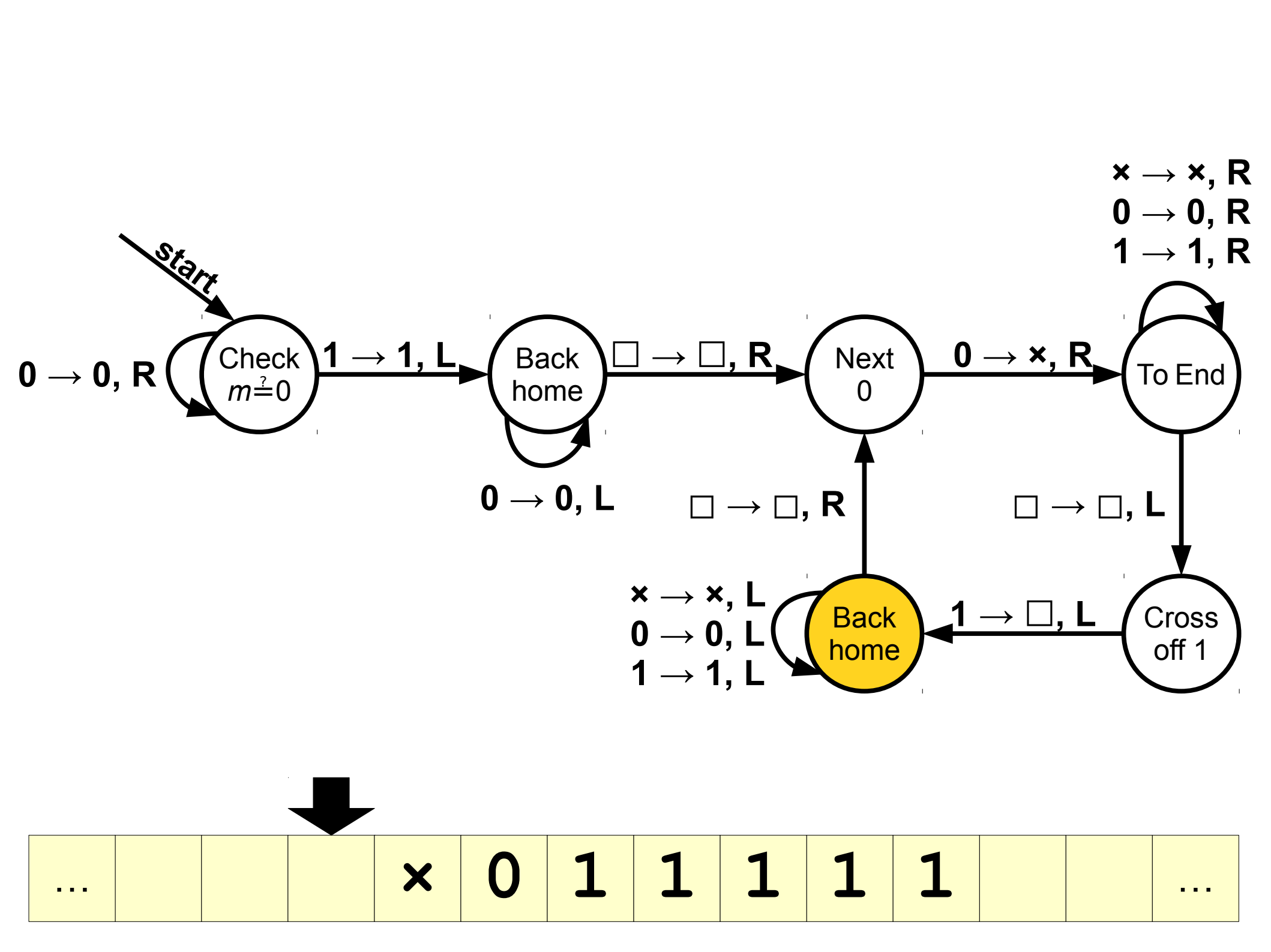

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
× → ×, L
0 → 0, L
1 → 1, L
… × 0 1 1 1 1 1 …

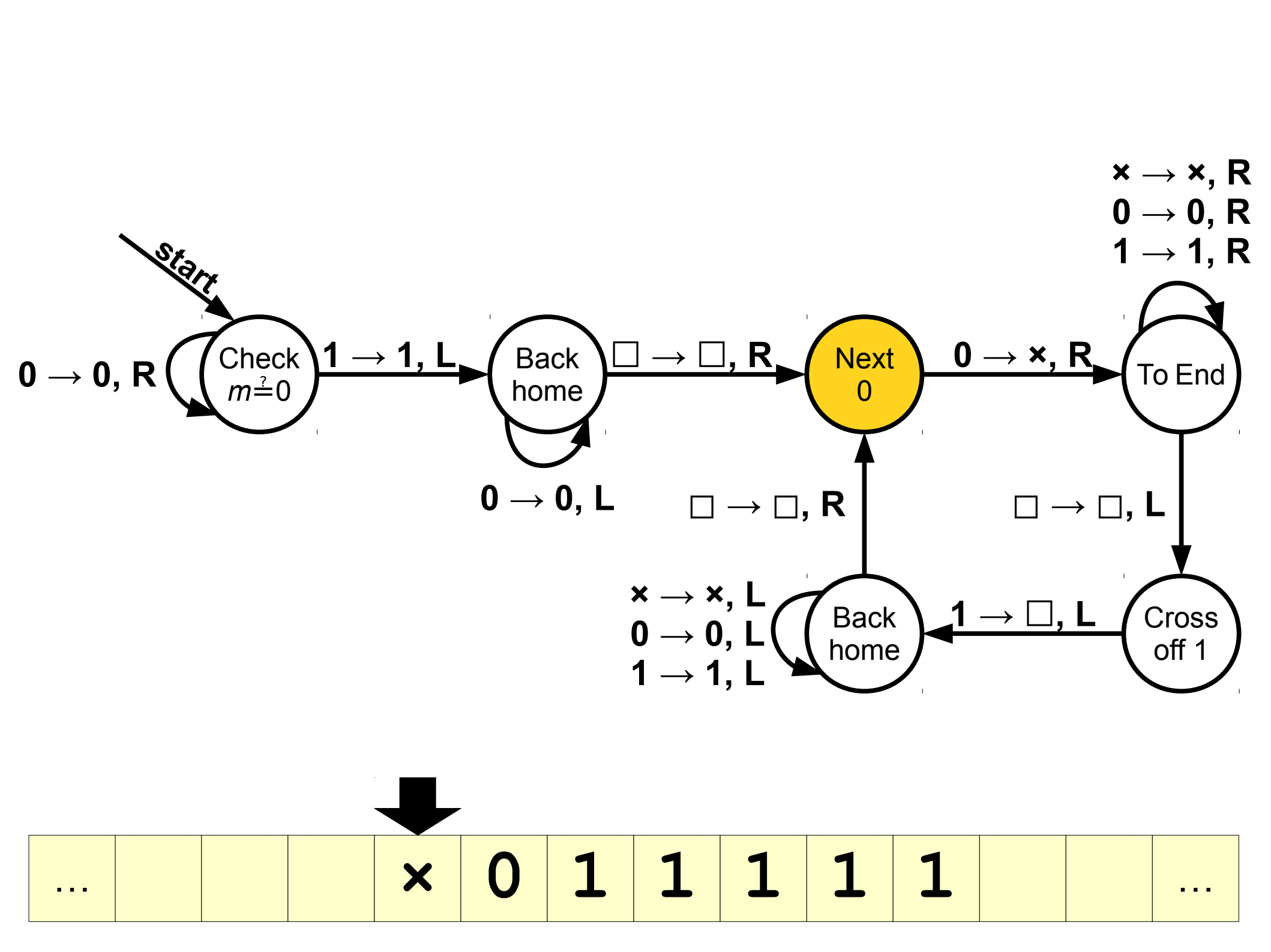

start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
0 → 0, L
□ → □, R
Next
0
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
… × 0 1 1 1 1 1 …

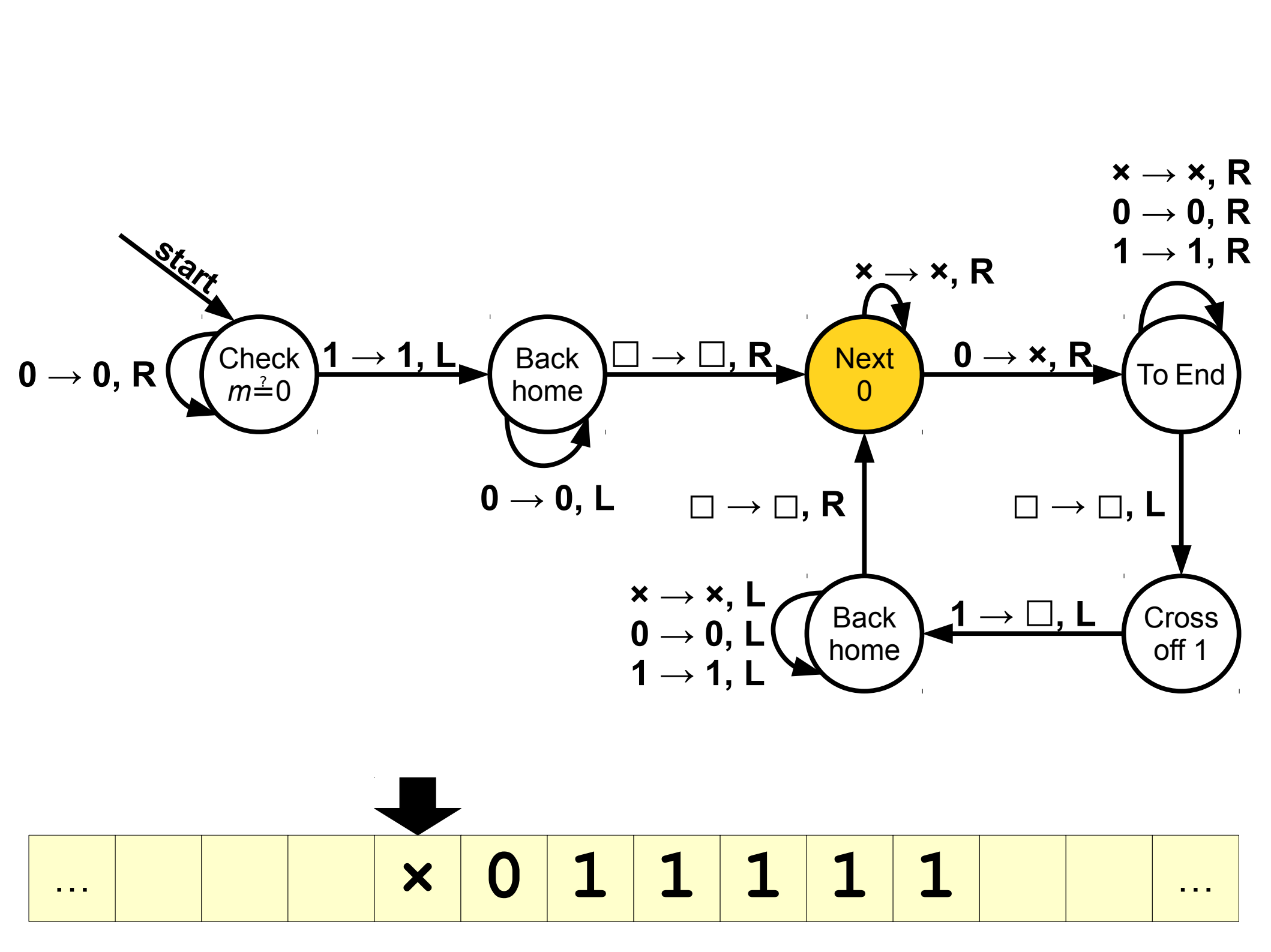

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Cross
off 1
Back
home
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
1 → □, L
× → ×, L
0 → 0, L
1 → 1, L
…
×
0
1
1
1
1
1
…

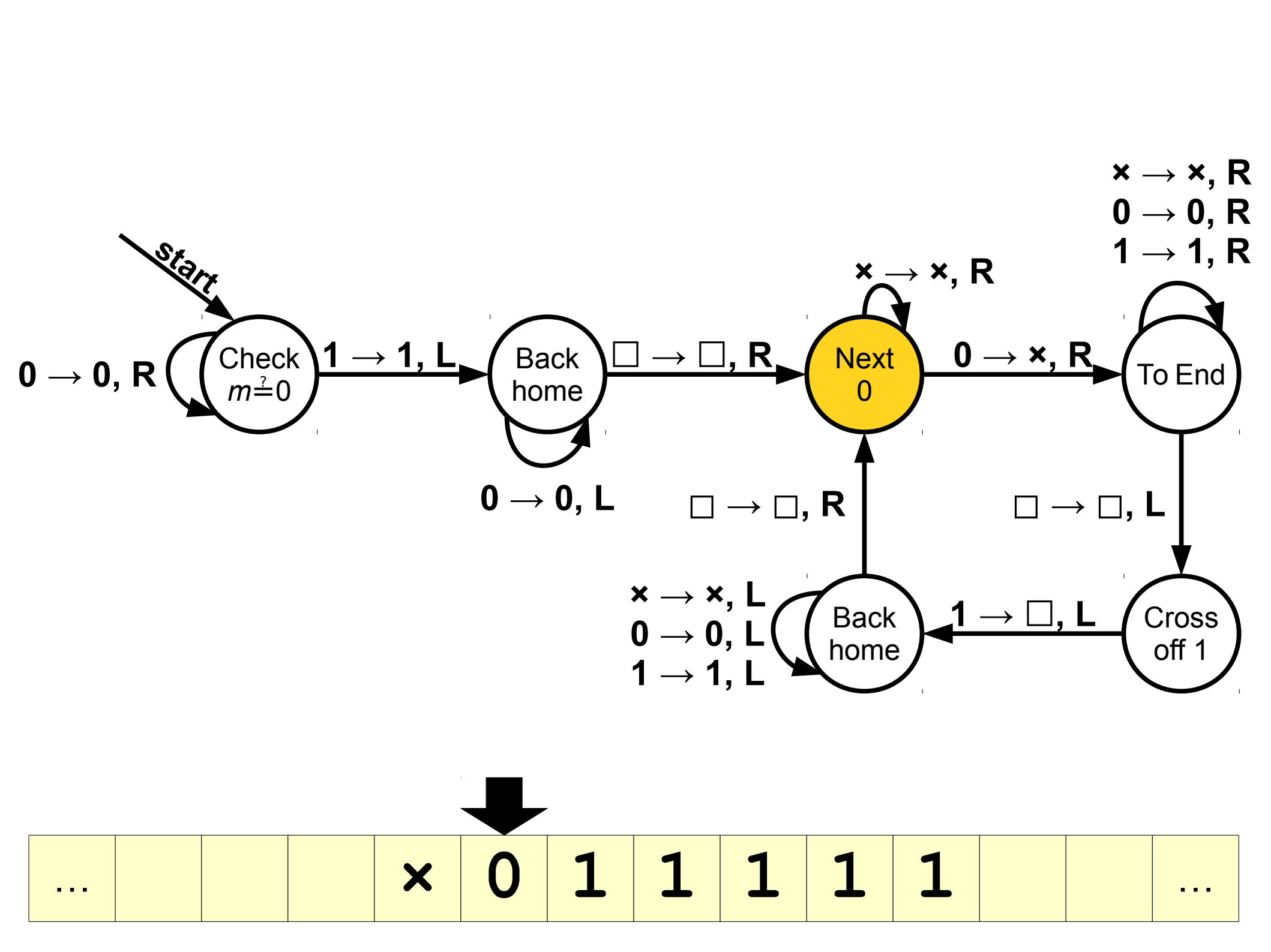

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × 0 1 1 1 1 1 …

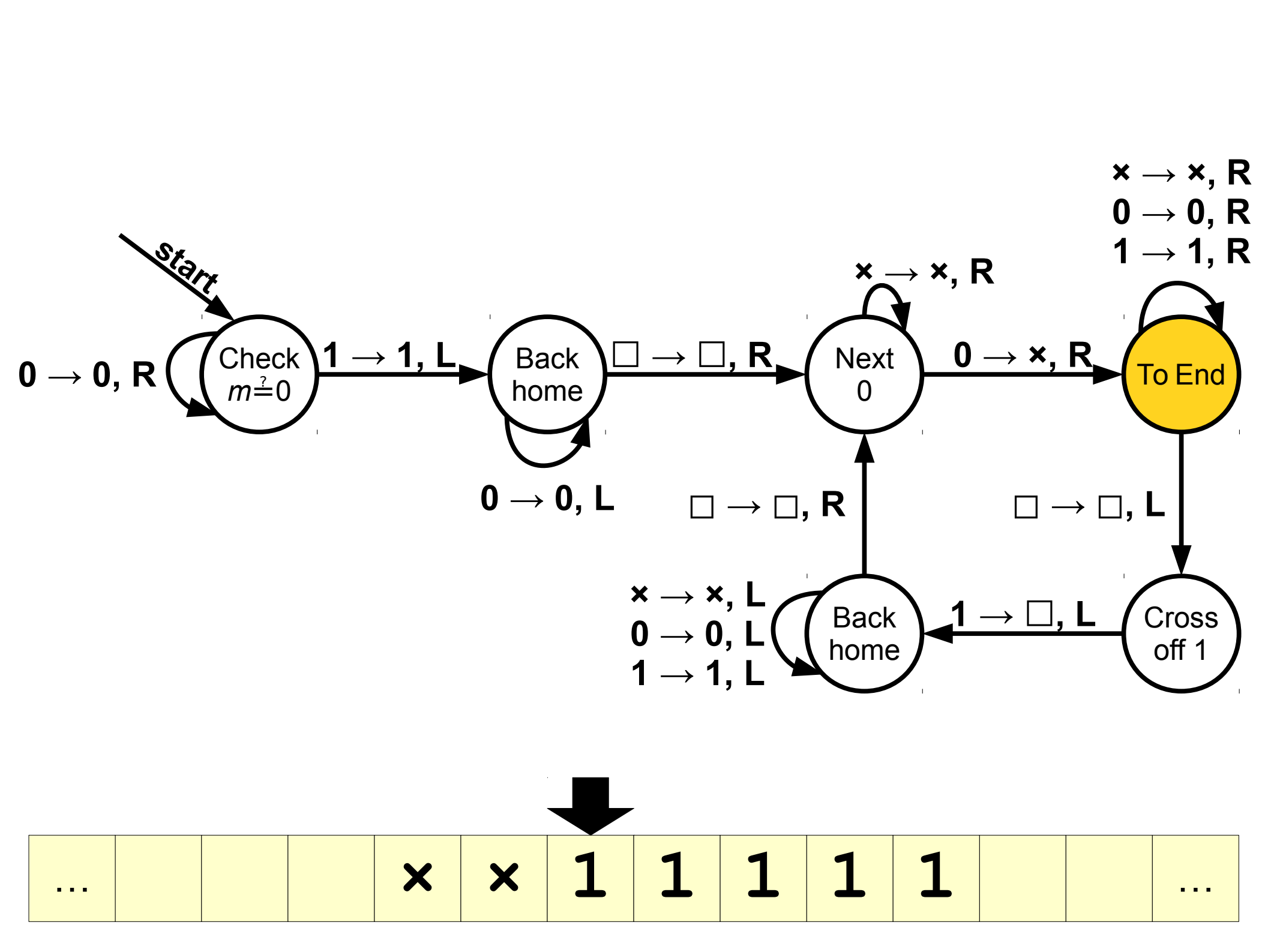

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
…
×
0
1
1
1
1
1
…

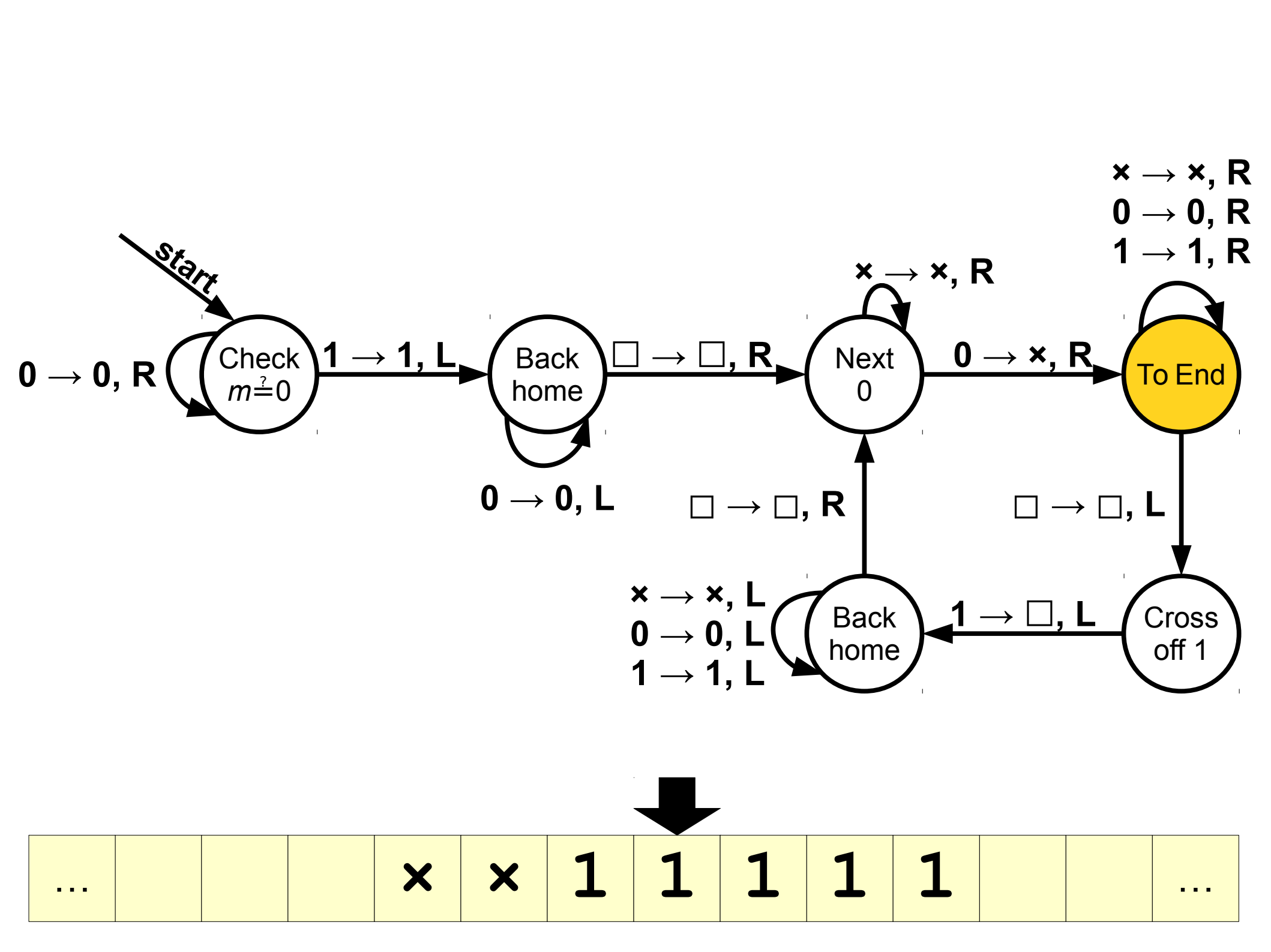

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × 0 1 1 1 1 1 …

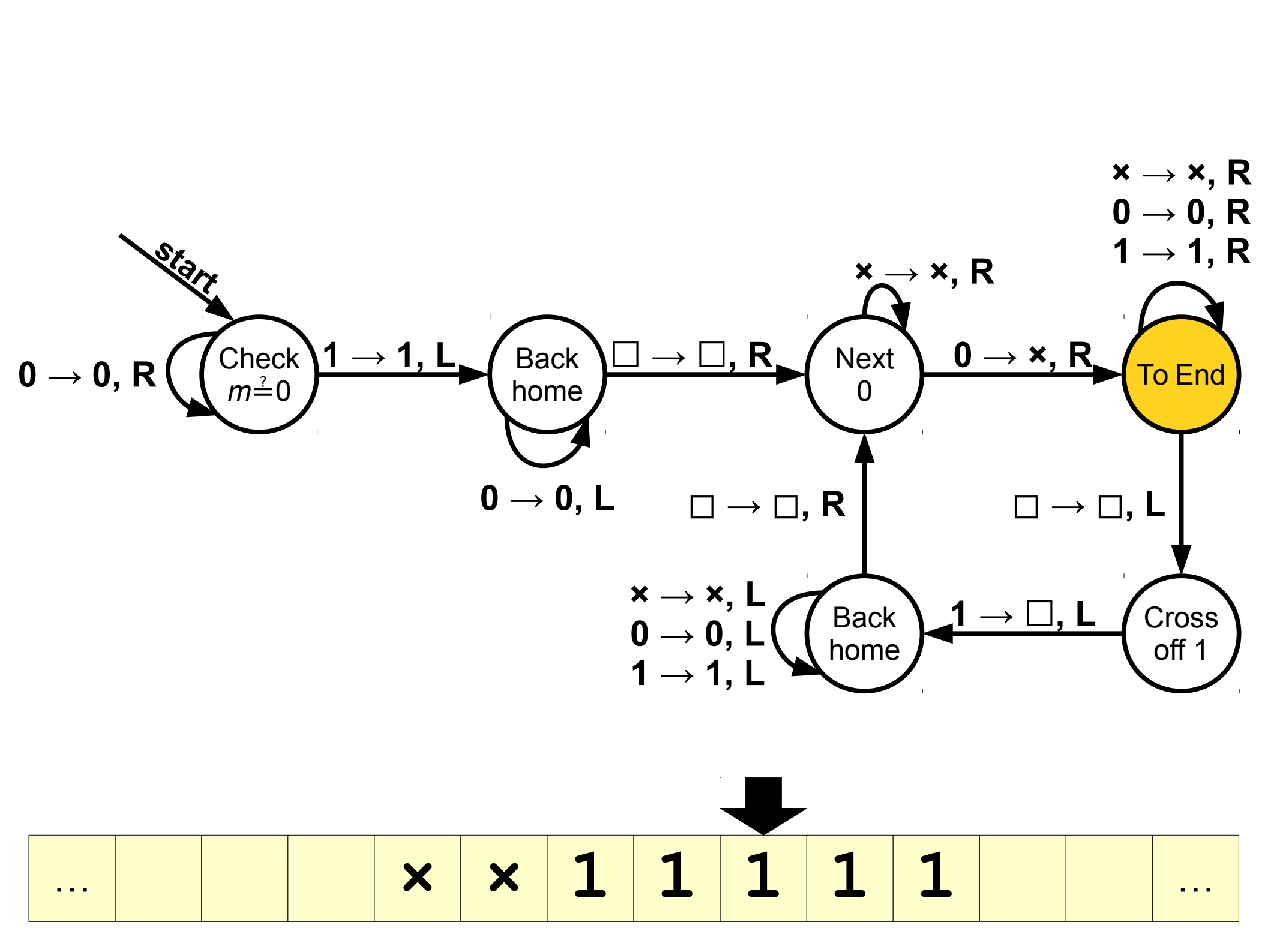

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × 0 1 1 1 1 1 …

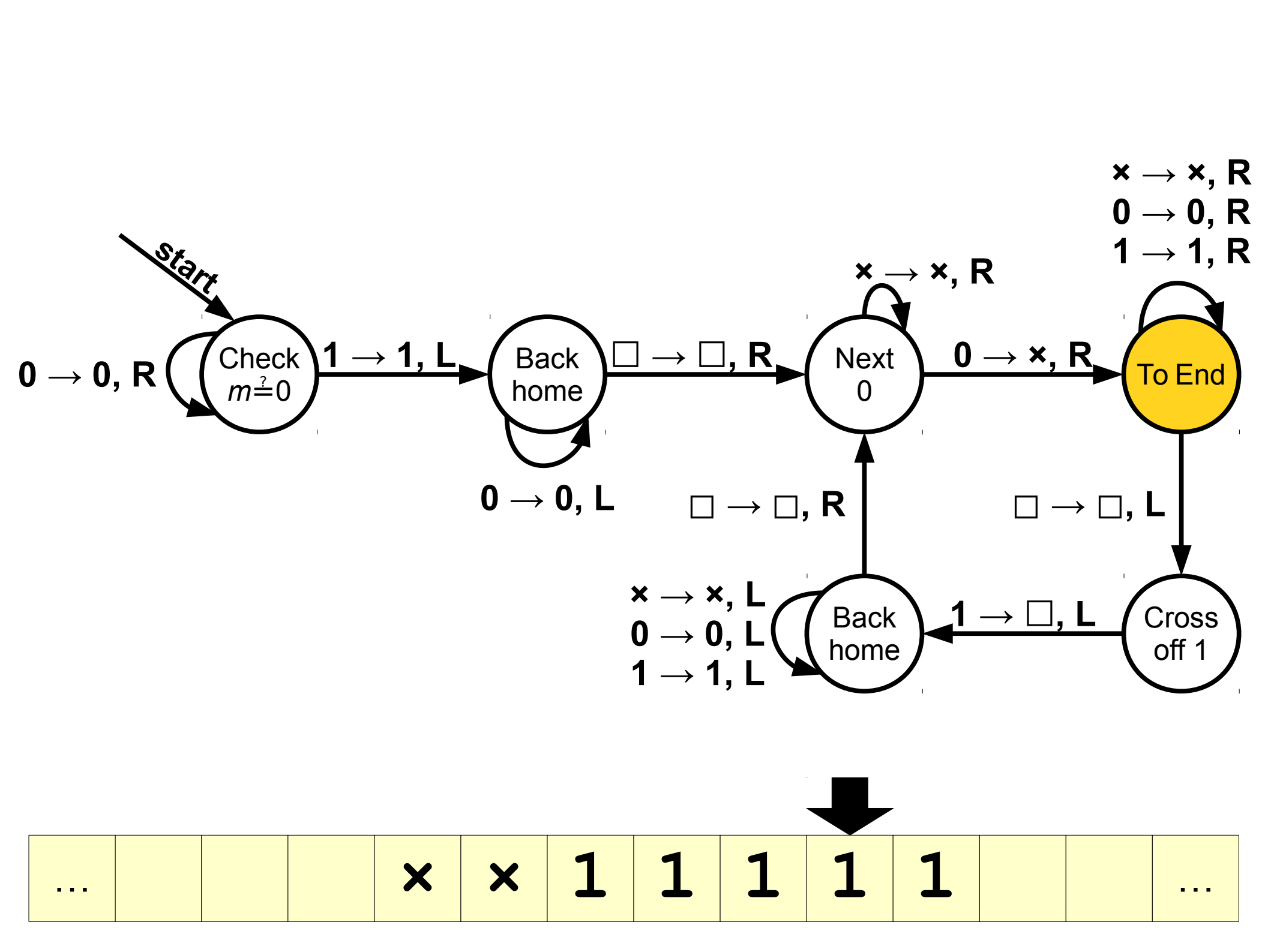

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × 0 1 1 1 1 1 …

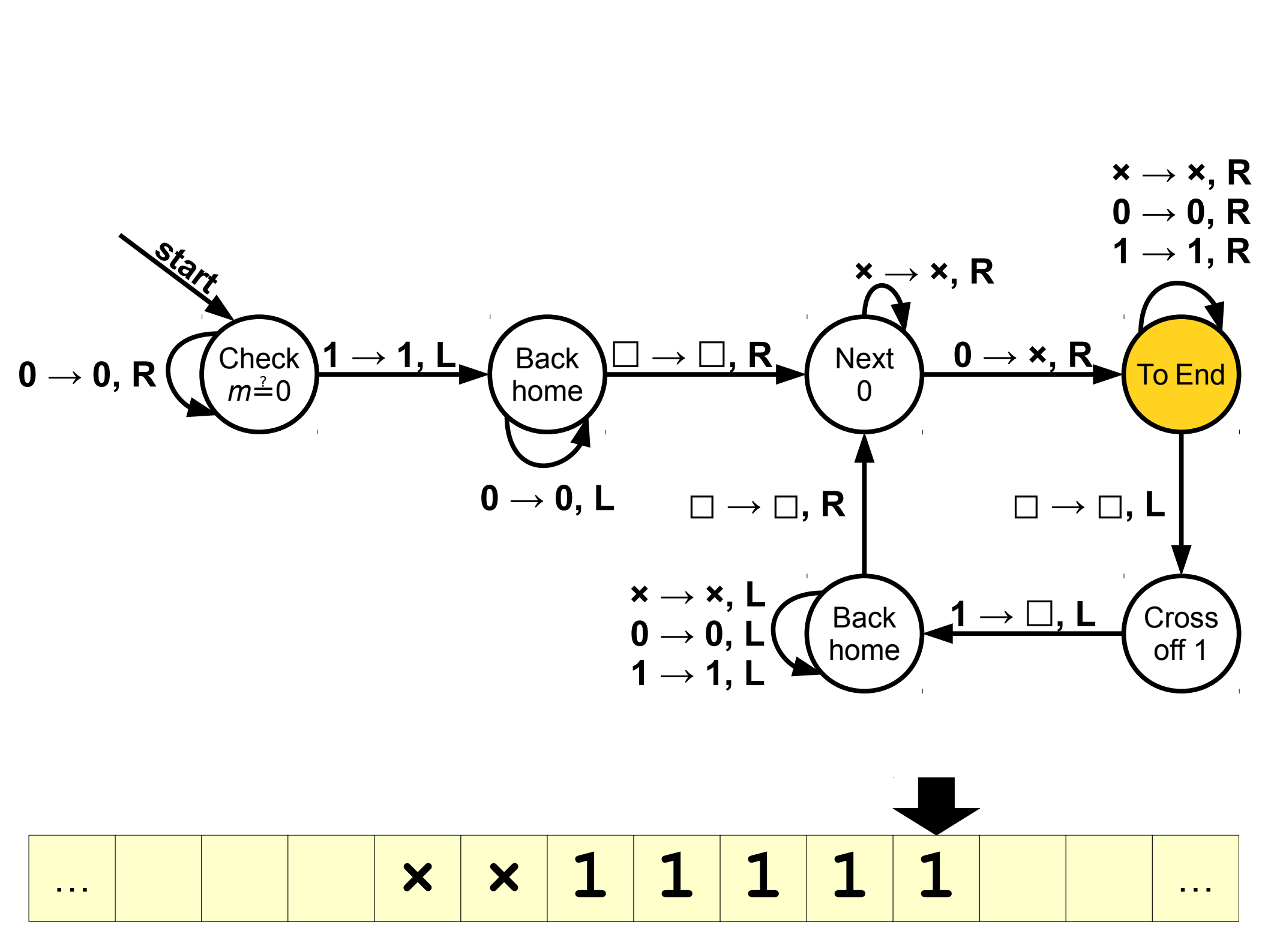

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × 0 1 1 1 1 1 …

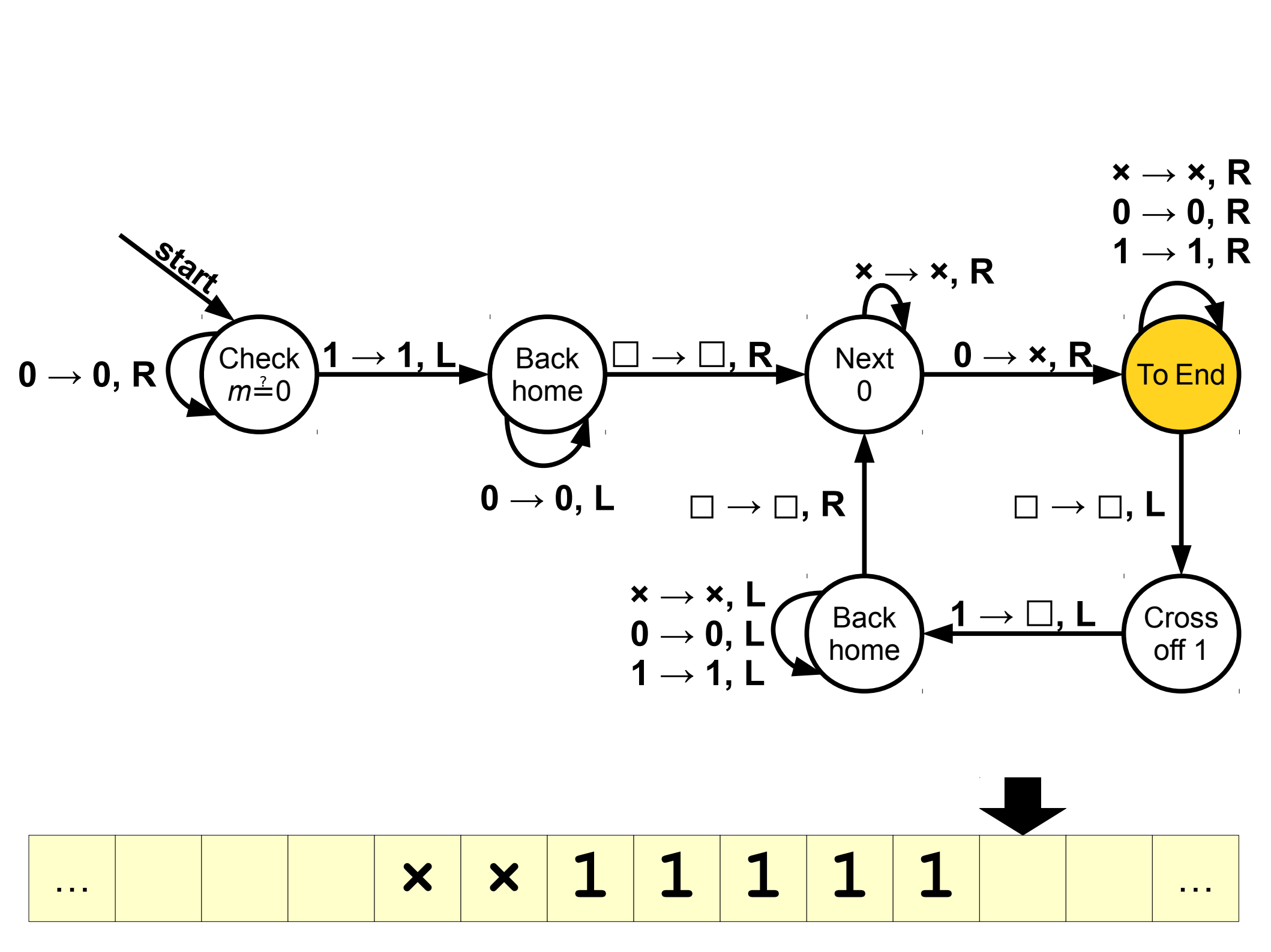

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × × 1 1 1 1 1 …

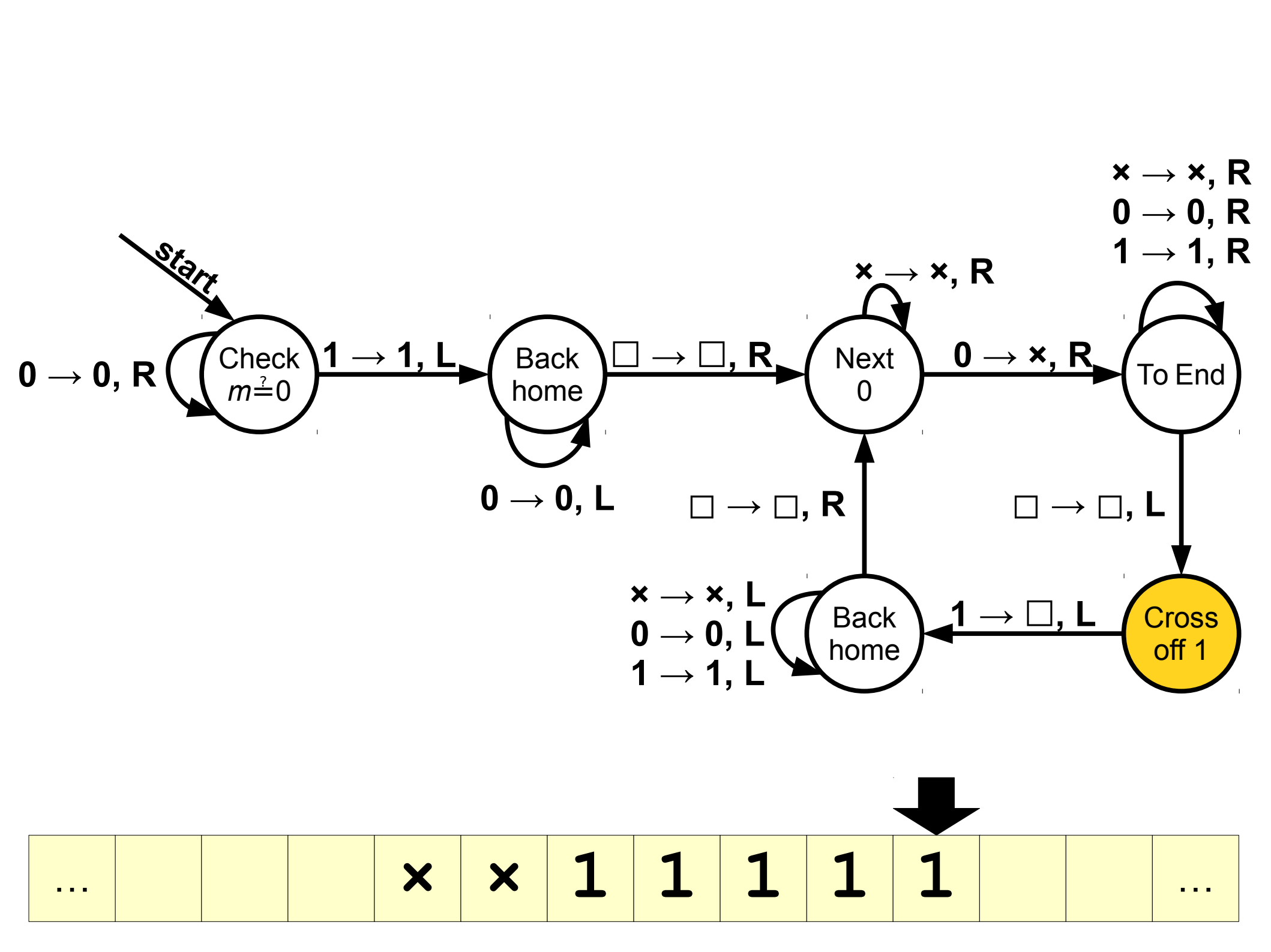

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 1 …

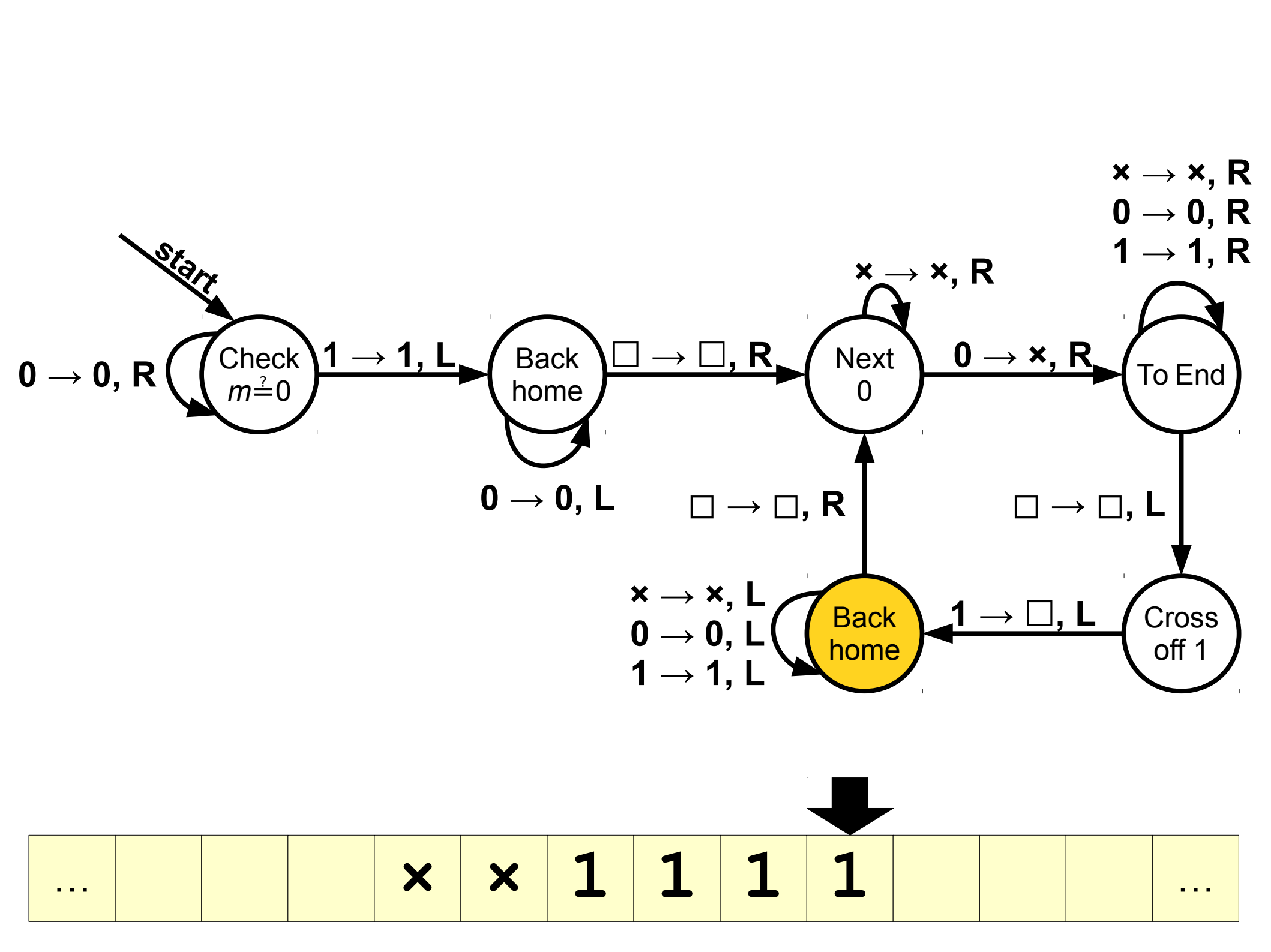

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 1 …

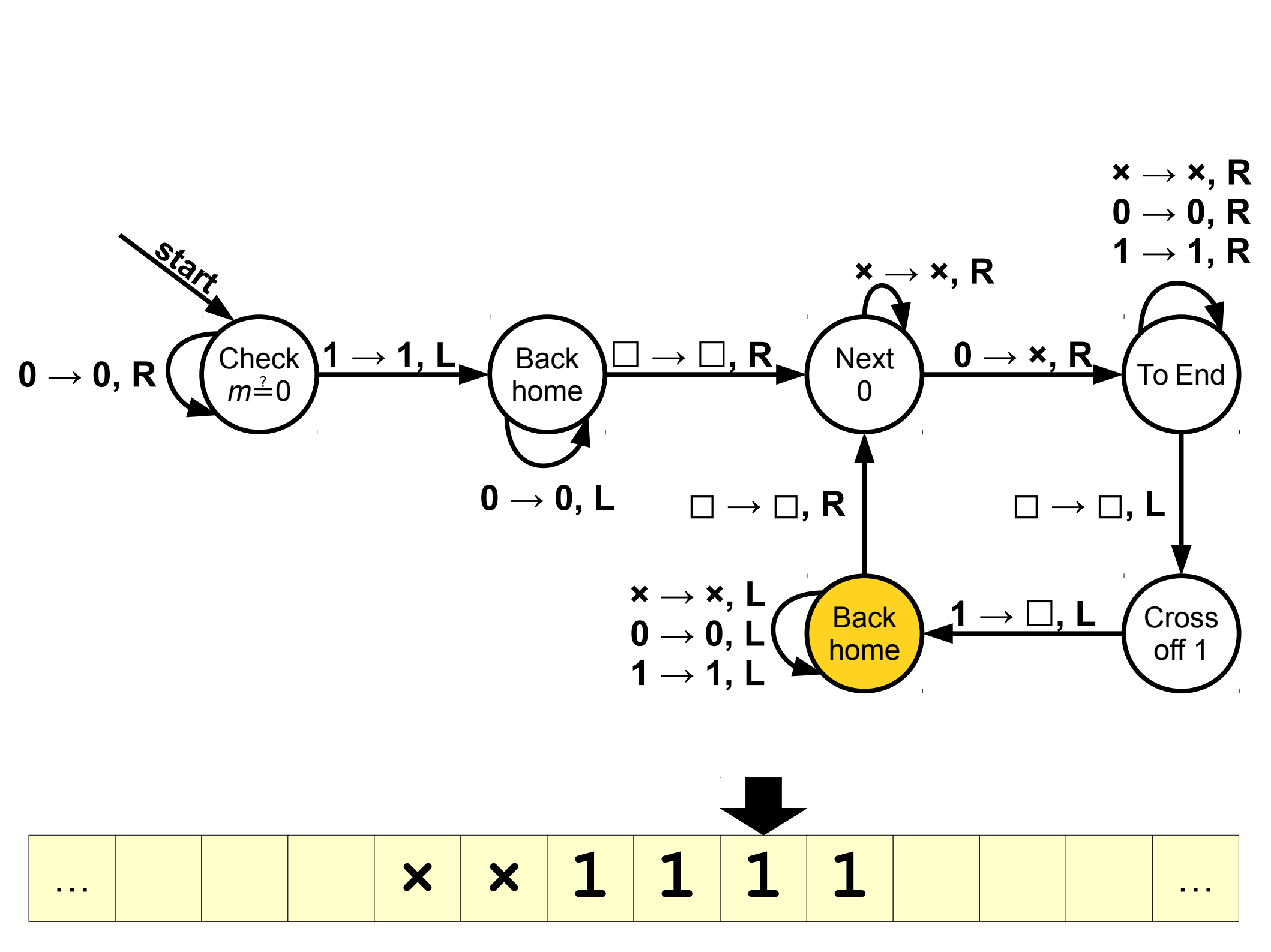

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 1 …

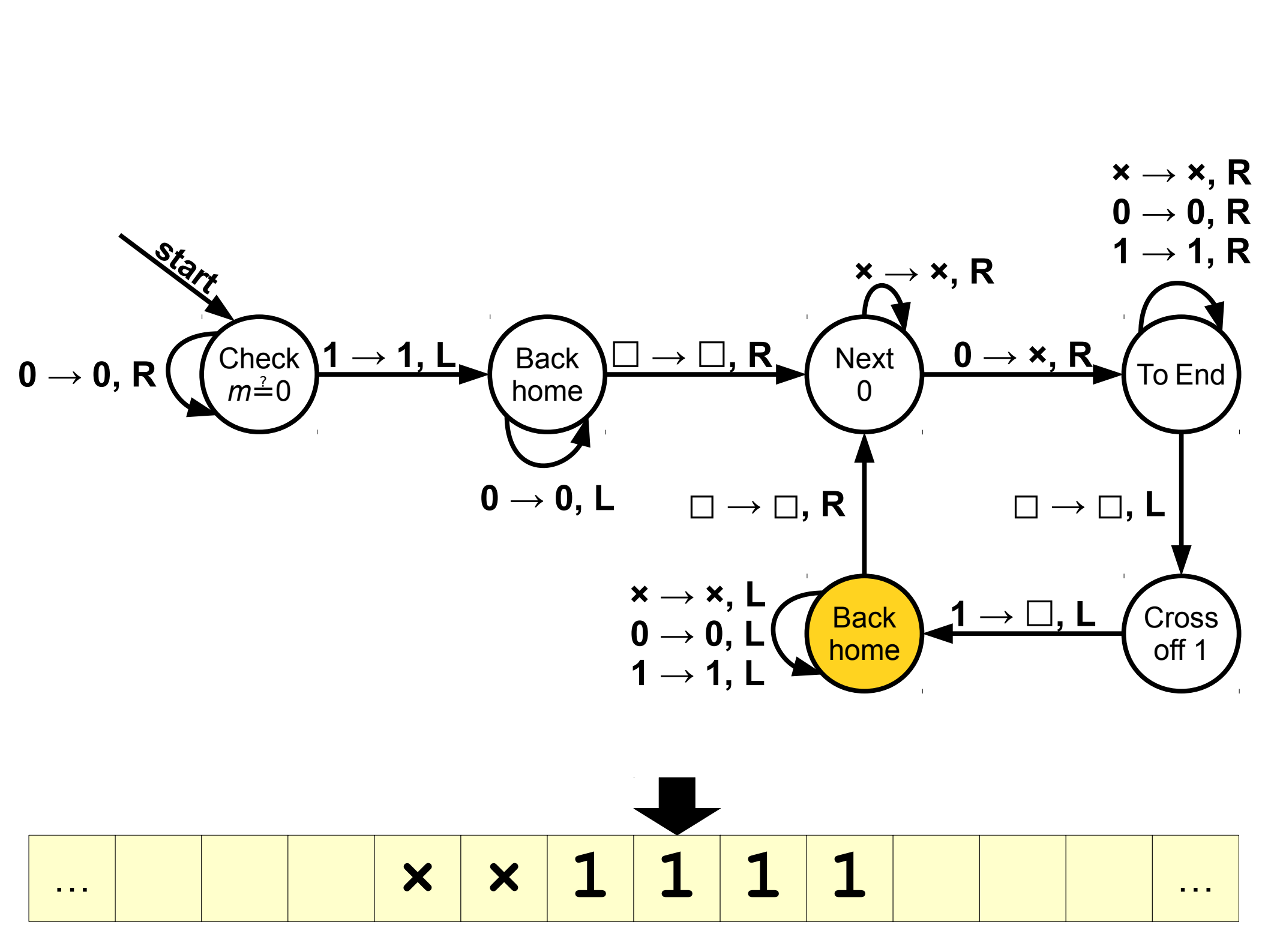

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 1 …

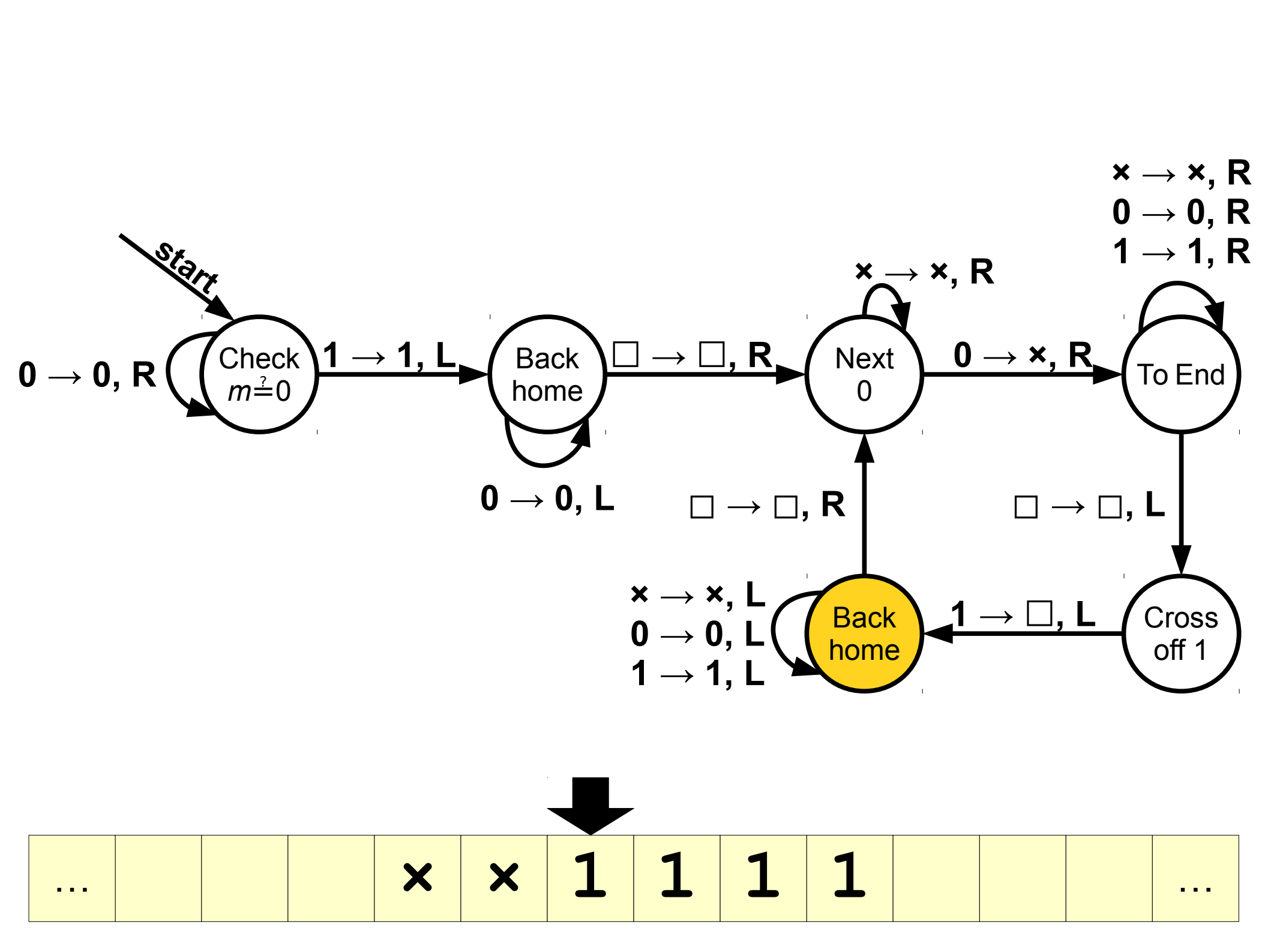

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 1 …

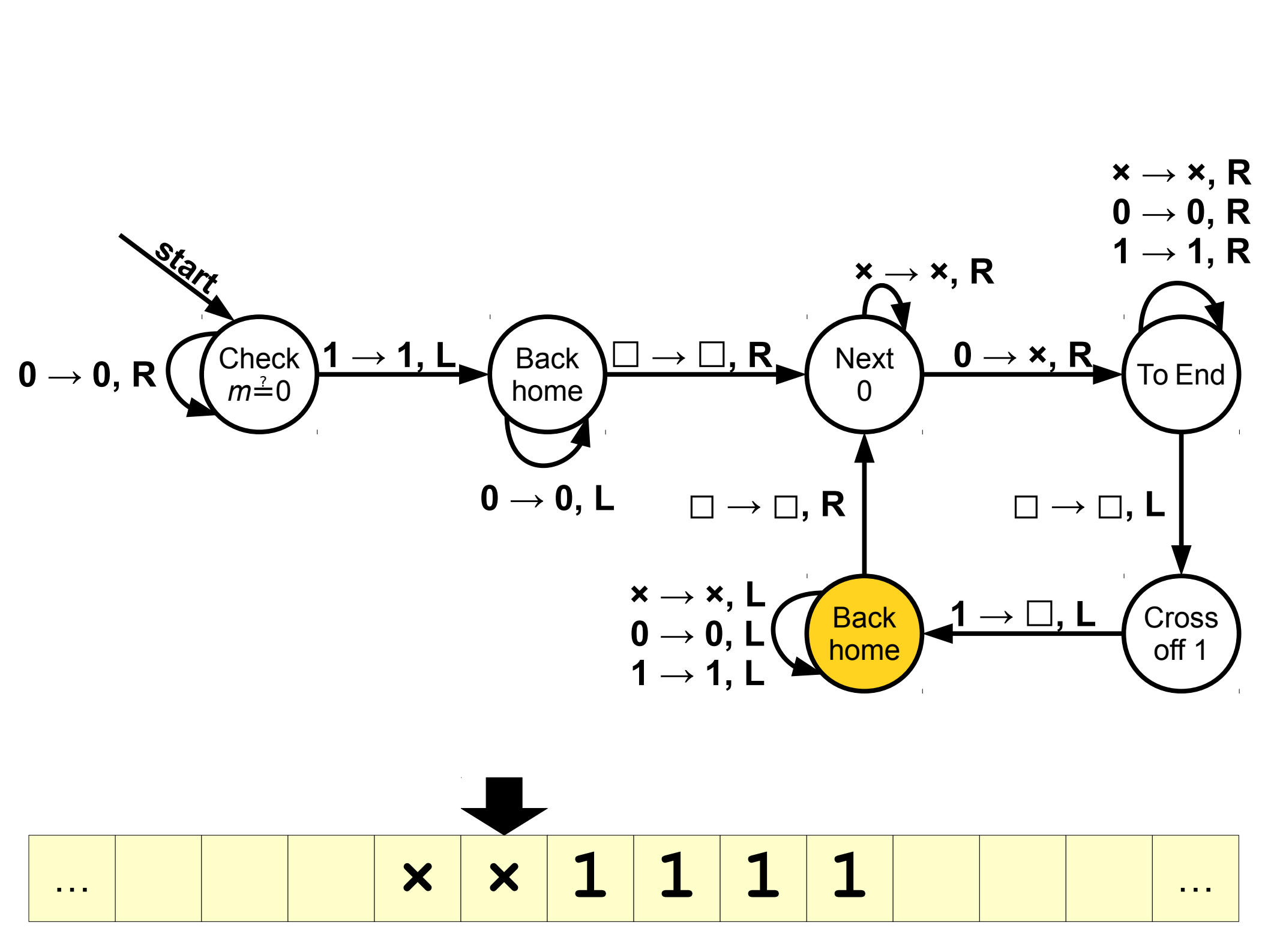

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × × 1 1 1 1 1 …

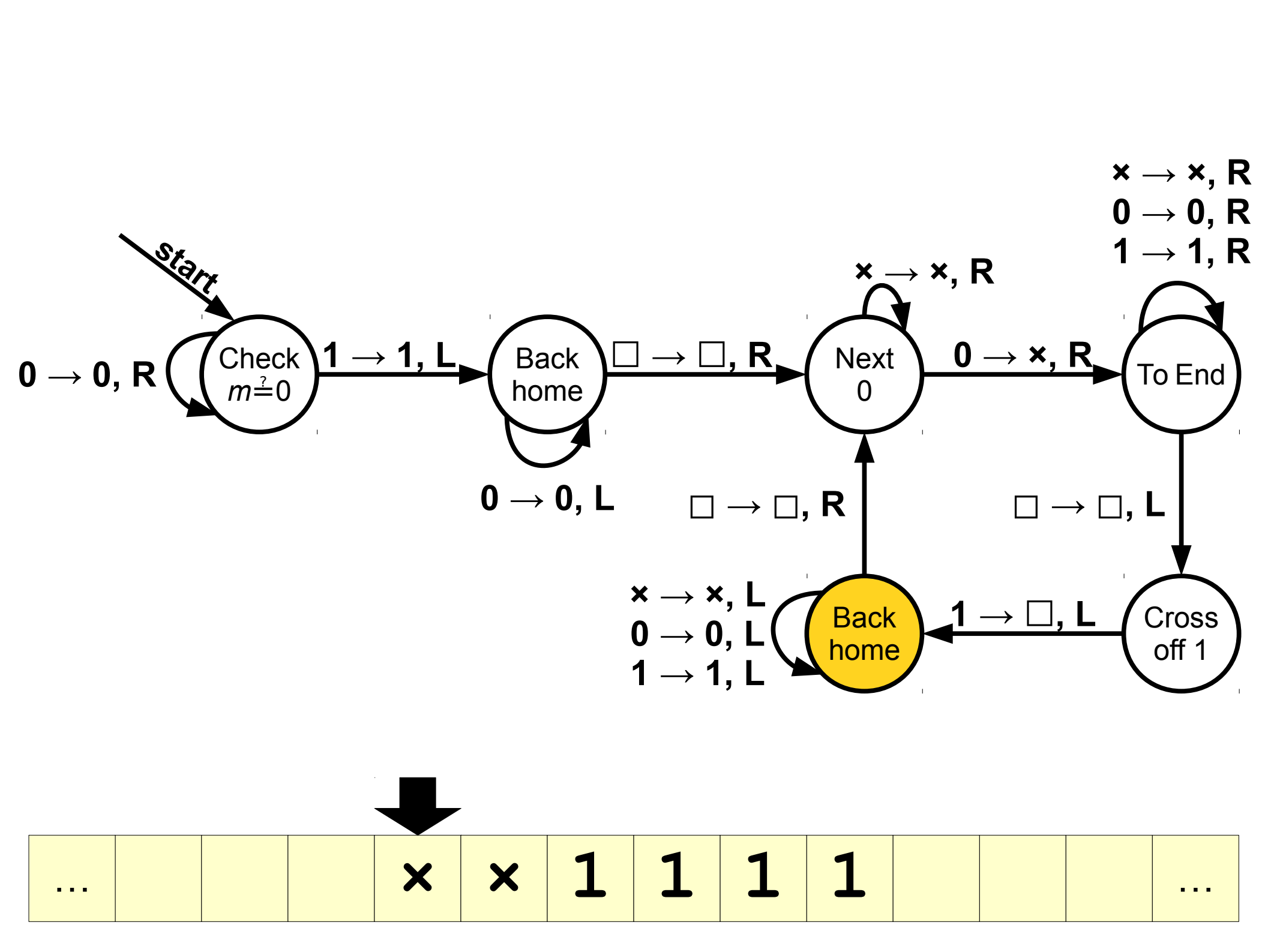

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
□ → □, R
× → ×, L
0 → 0, L
1 → 1, L
… × × 1 1 1 1 …

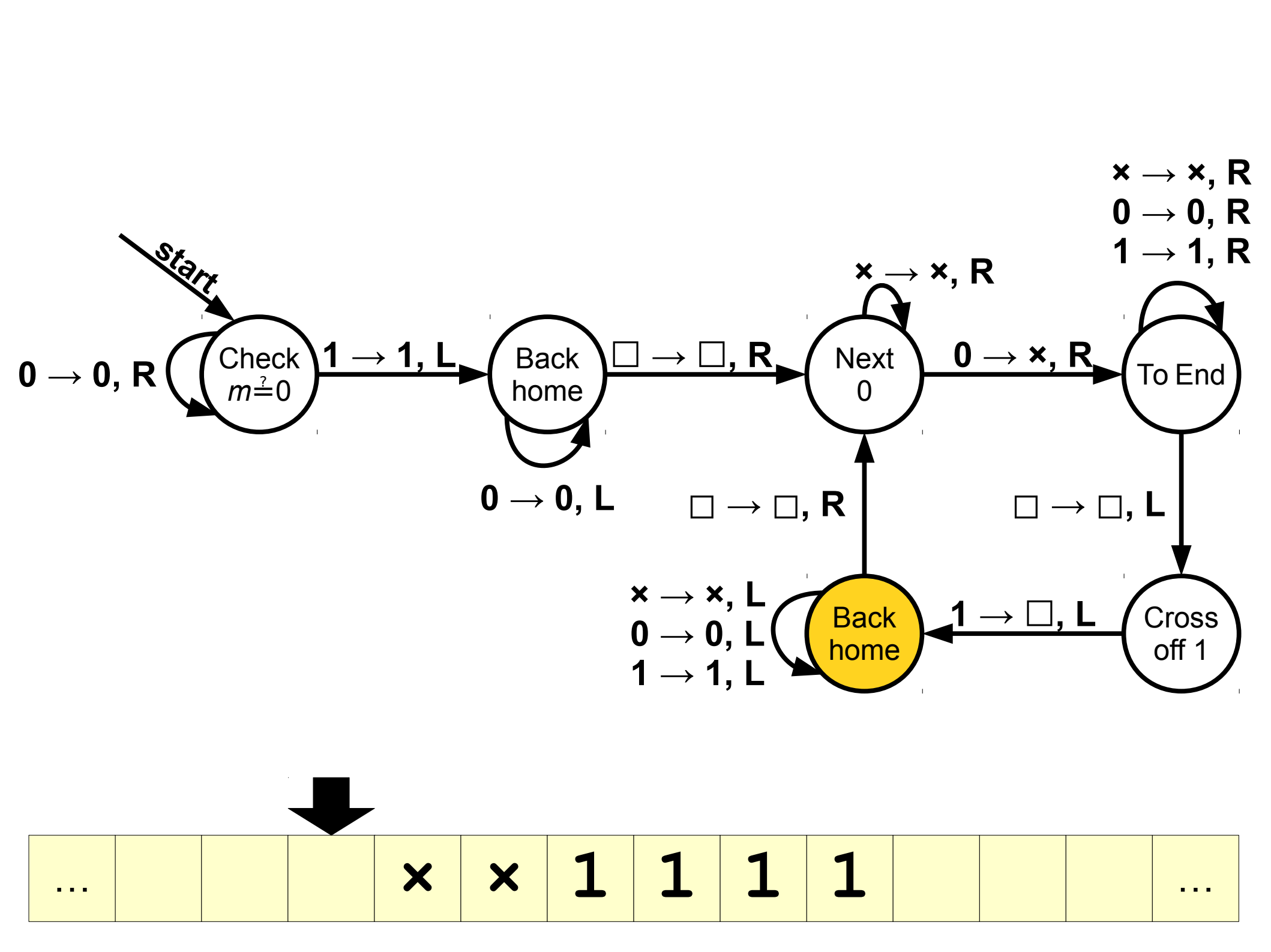

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
□ → □, R
× → ×, L
0 → 0, L
1 → 1, L
… × × 1 1 1 1 …

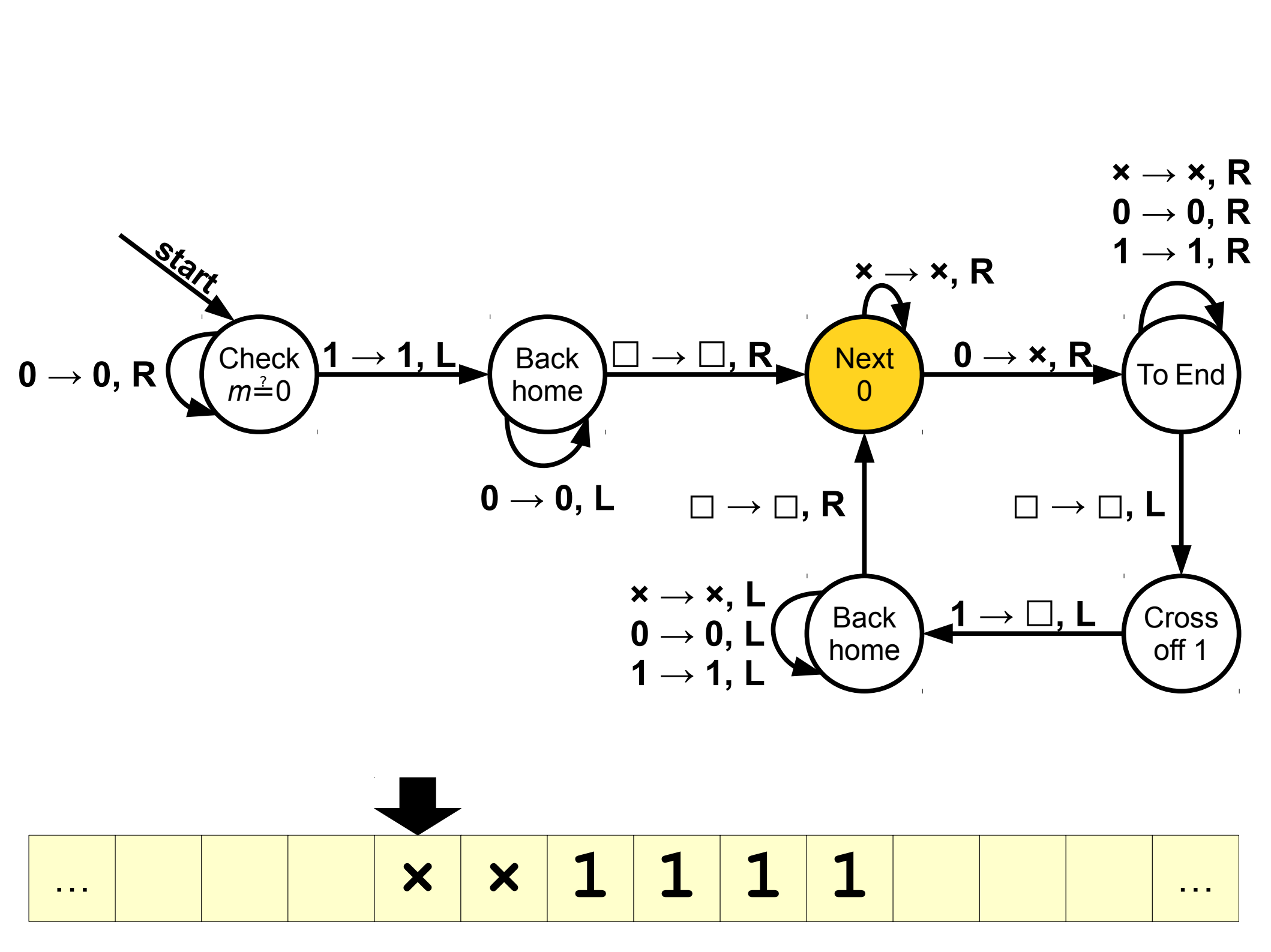

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
□ → □, R
× → ×, L
0 → 0, L
1 → 1, L
… × × 1 1 1 1 …

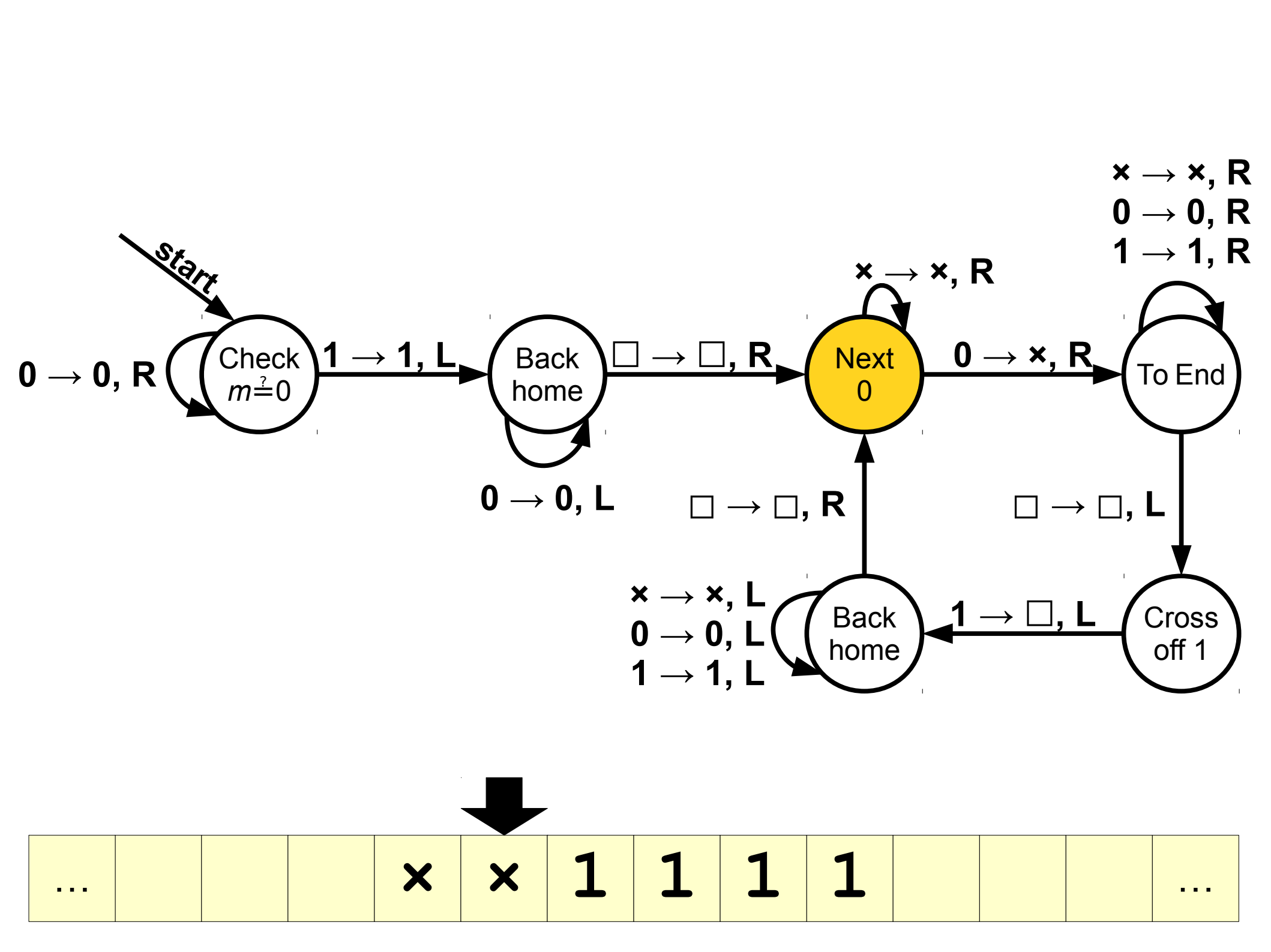

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × × 1 1 1 1 …

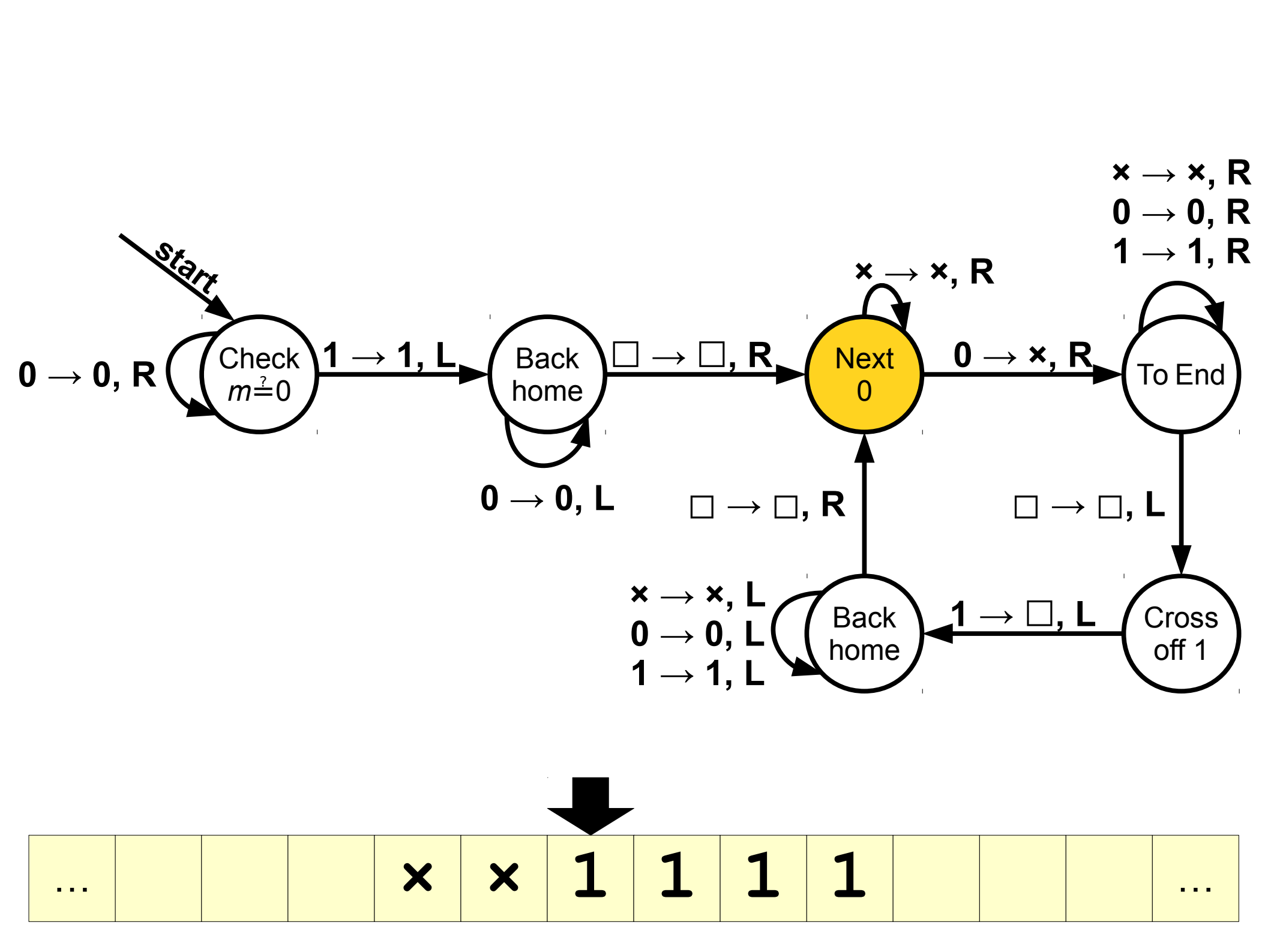

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
□ → □, R
× → ×, L
0 → 0, L
1 → 1, L
… × × 1 1 1 1 …

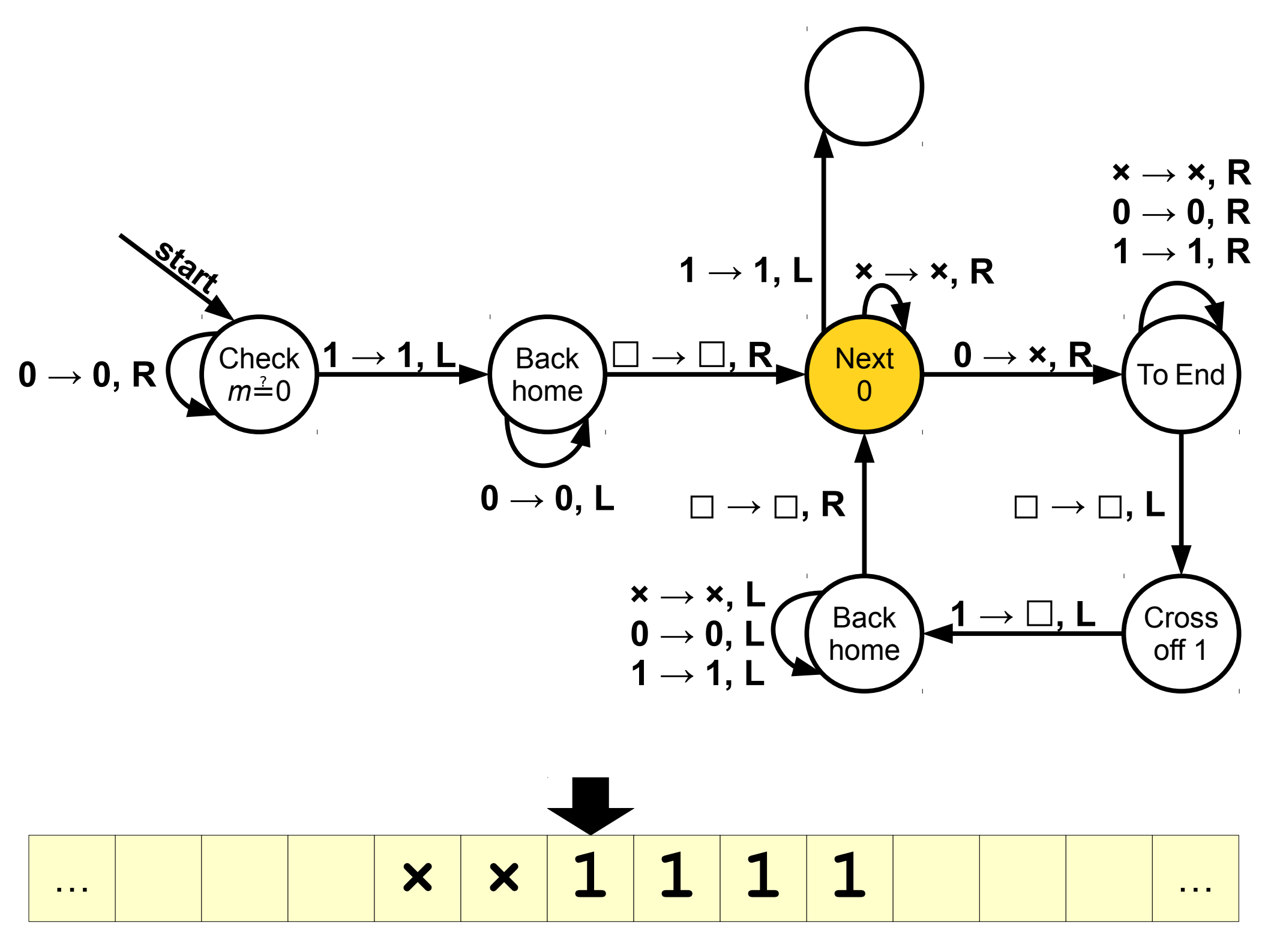

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
□ → □, R
× → ×, L
0 → 0, L
1 → 1, L
… × × 1 1 1 1 …

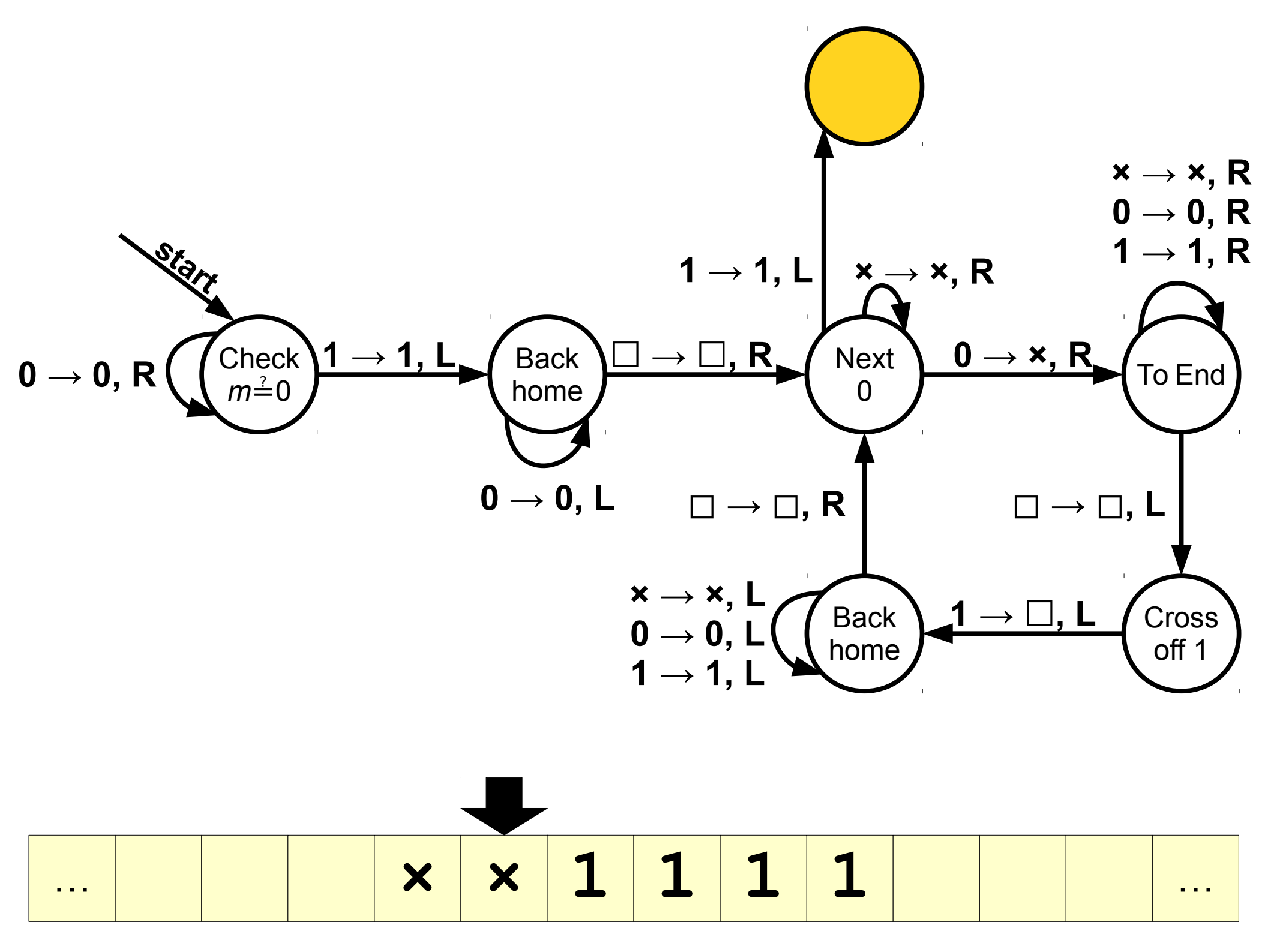

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
□ → □, R
× → ×, L
0 → 0, L
1 → 1, L
… × × 1 1 1 1 …

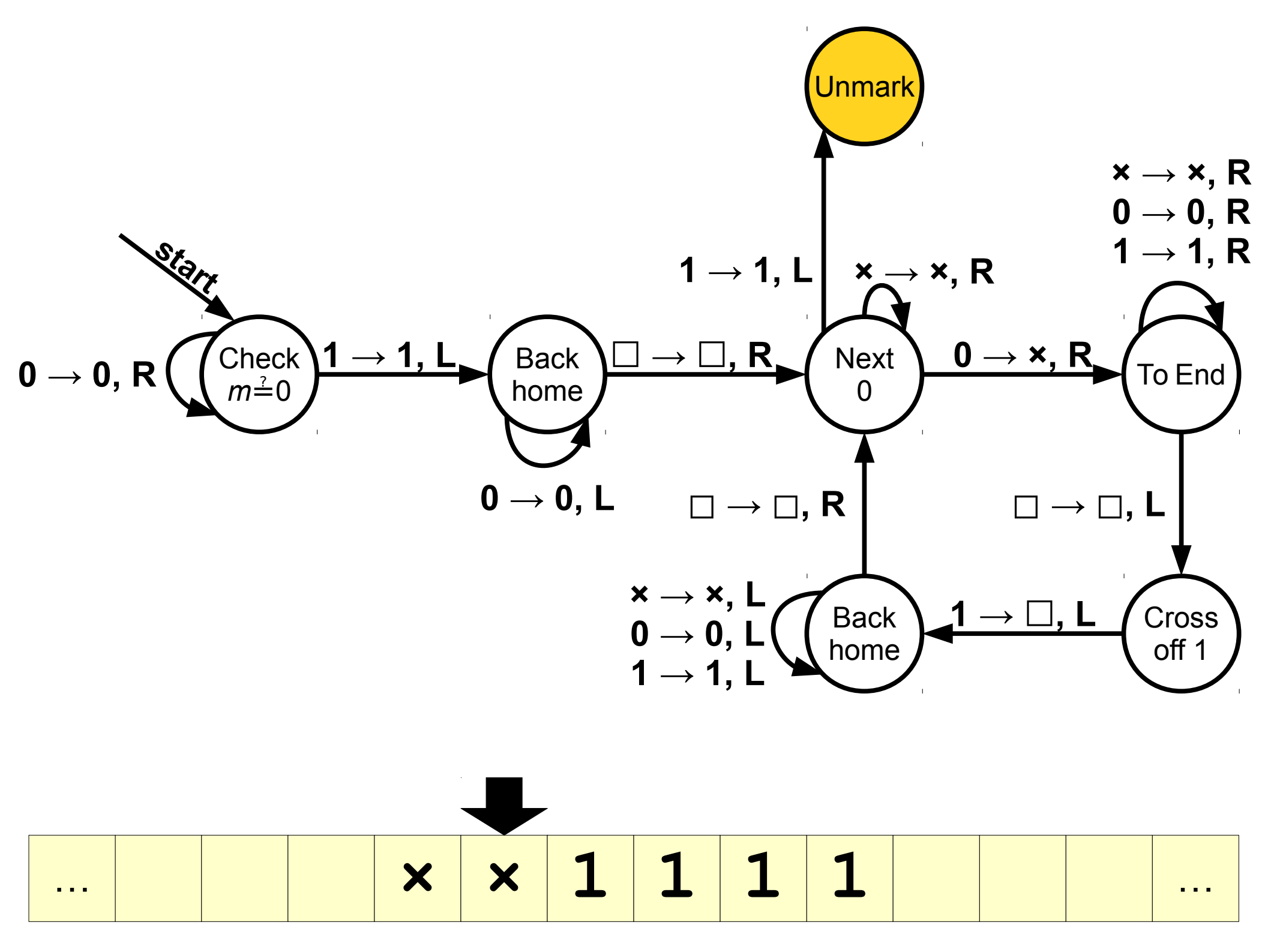

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 …

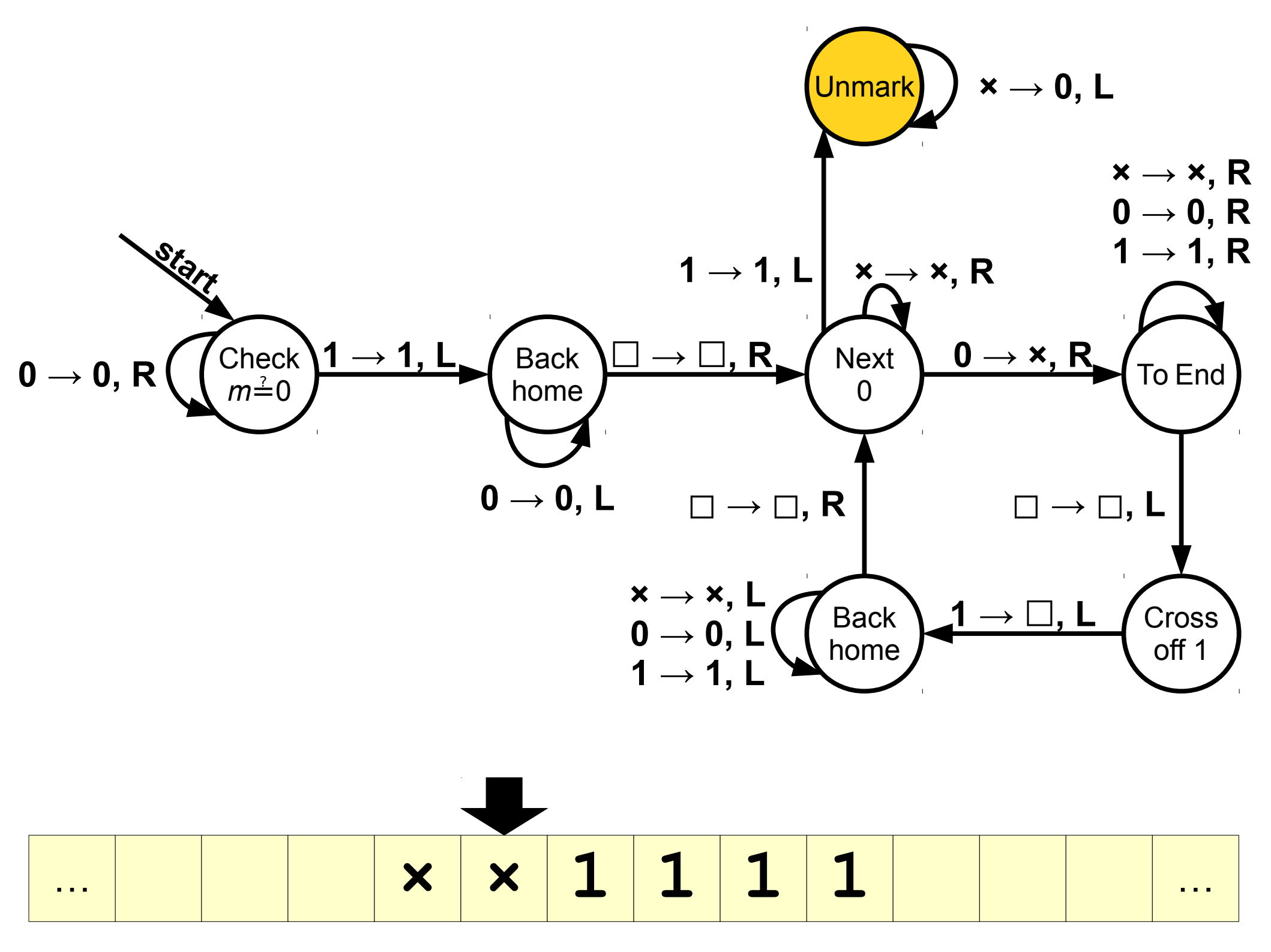

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × × 1 1 1 1 …

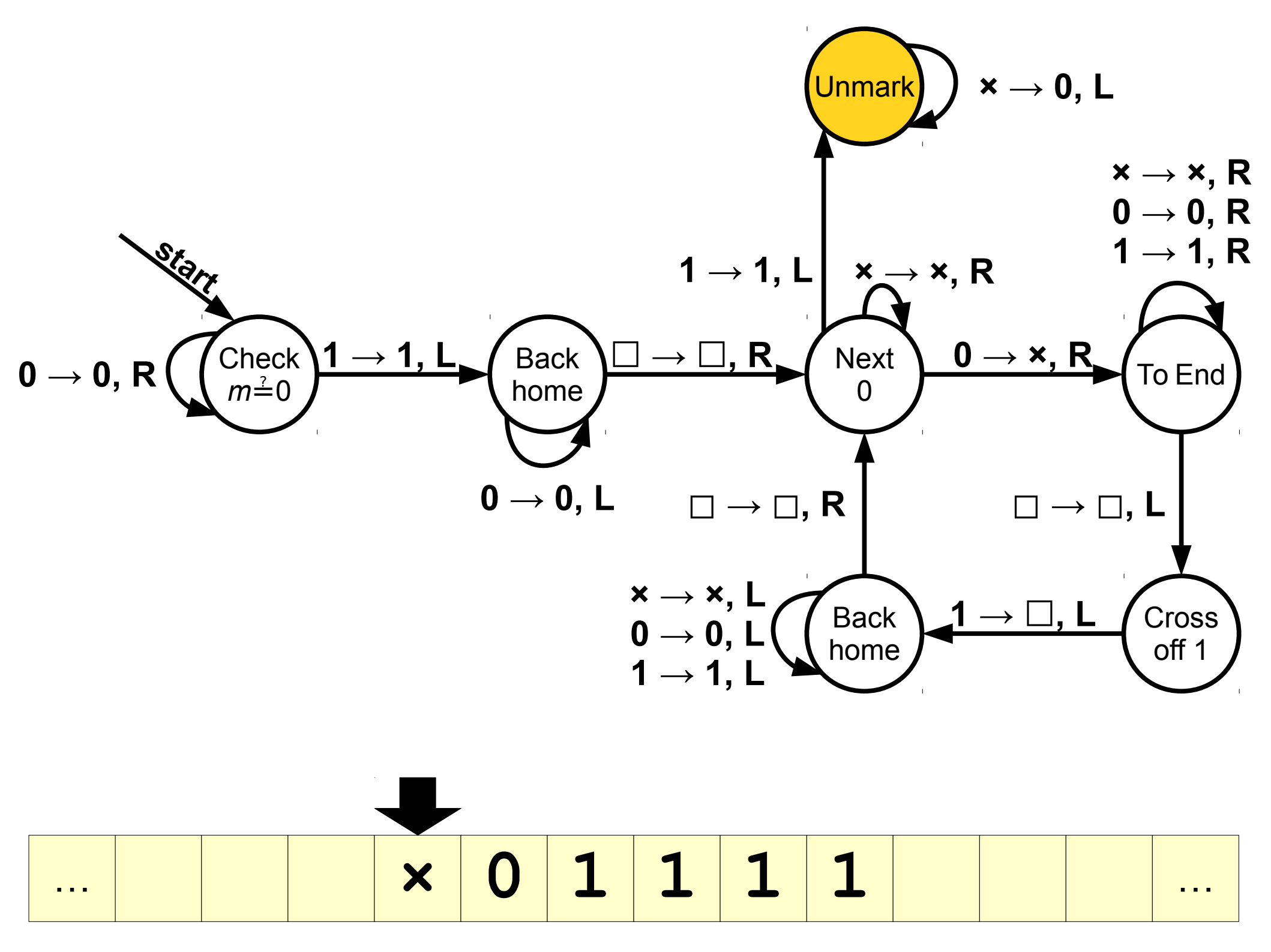

start
Check
$m \stackrel{?}{=} 0$
0 → 0, R
1 → 1, L
Back
home
□ → □, R
0 → 0, L
Next
0
× → ×, R
0 → ×, R
To End
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
Cross
off 1
1 → □, L
Back
home
× → ×, L
0 → 0, L
1 → 1, L
□ → □, R
… × × 1 1 1 1 …

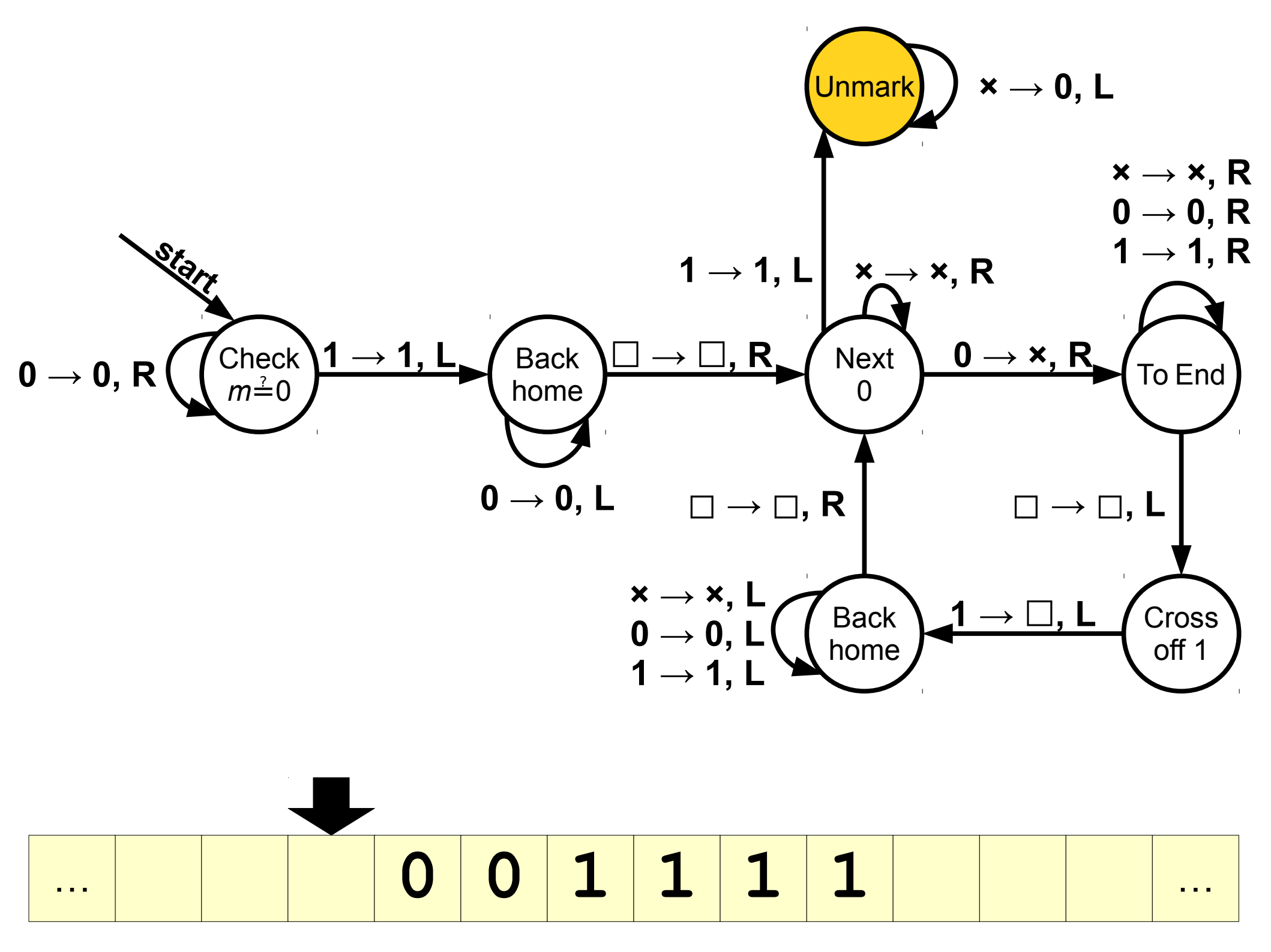

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
1 → 1, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 …

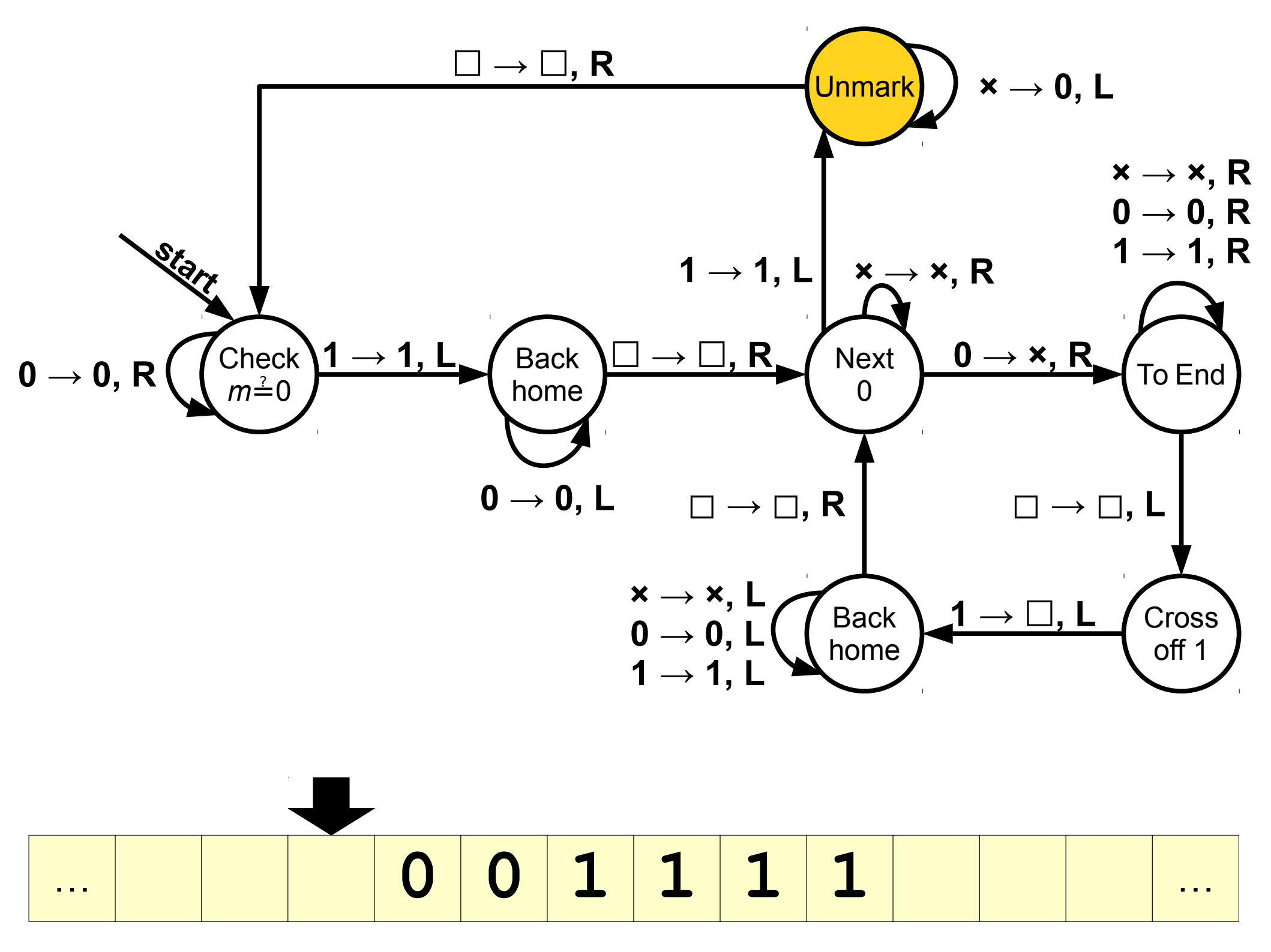

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
1 → 1, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 …

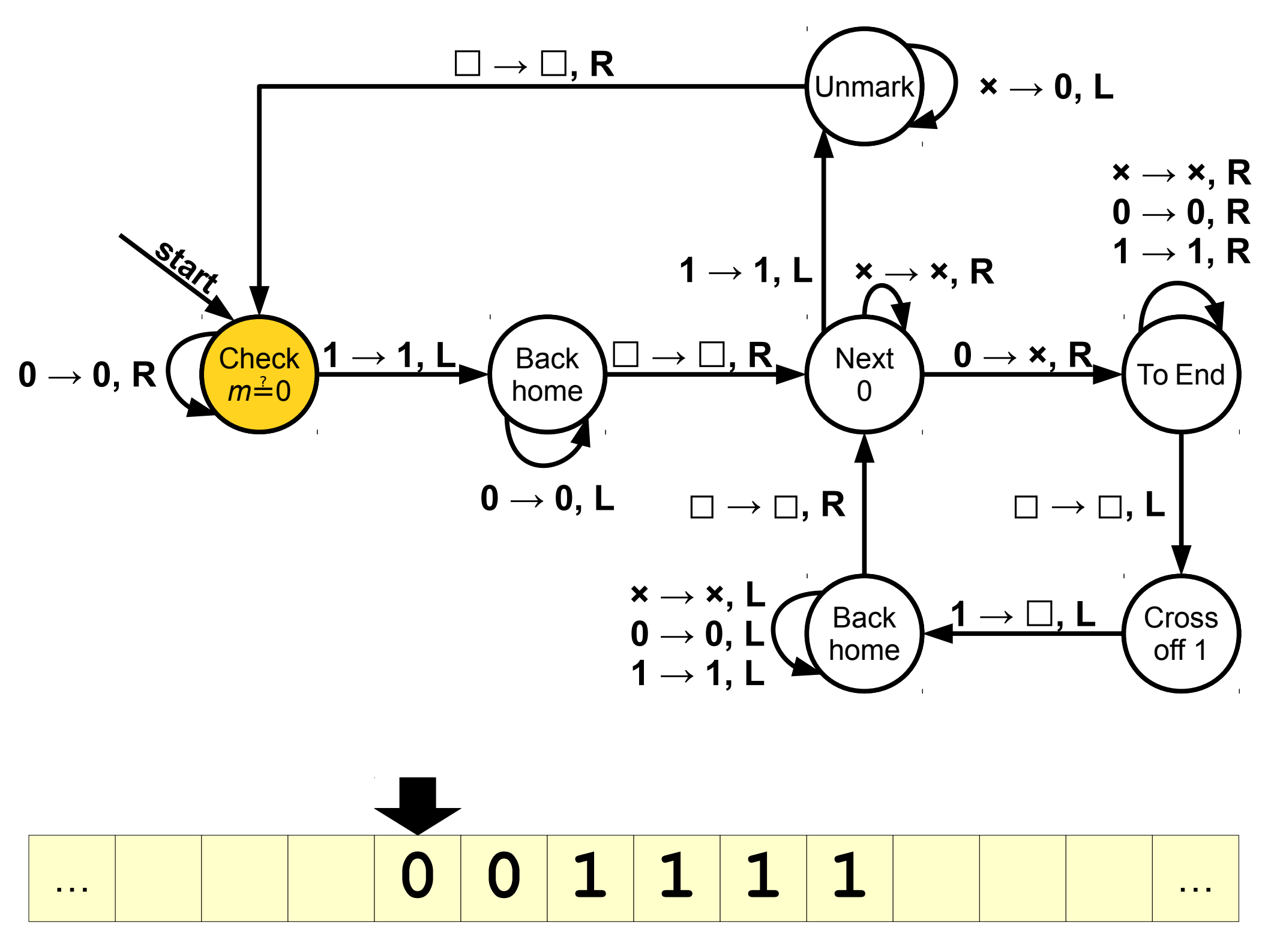

Unmark
start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
1 → 1, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 1 1 …

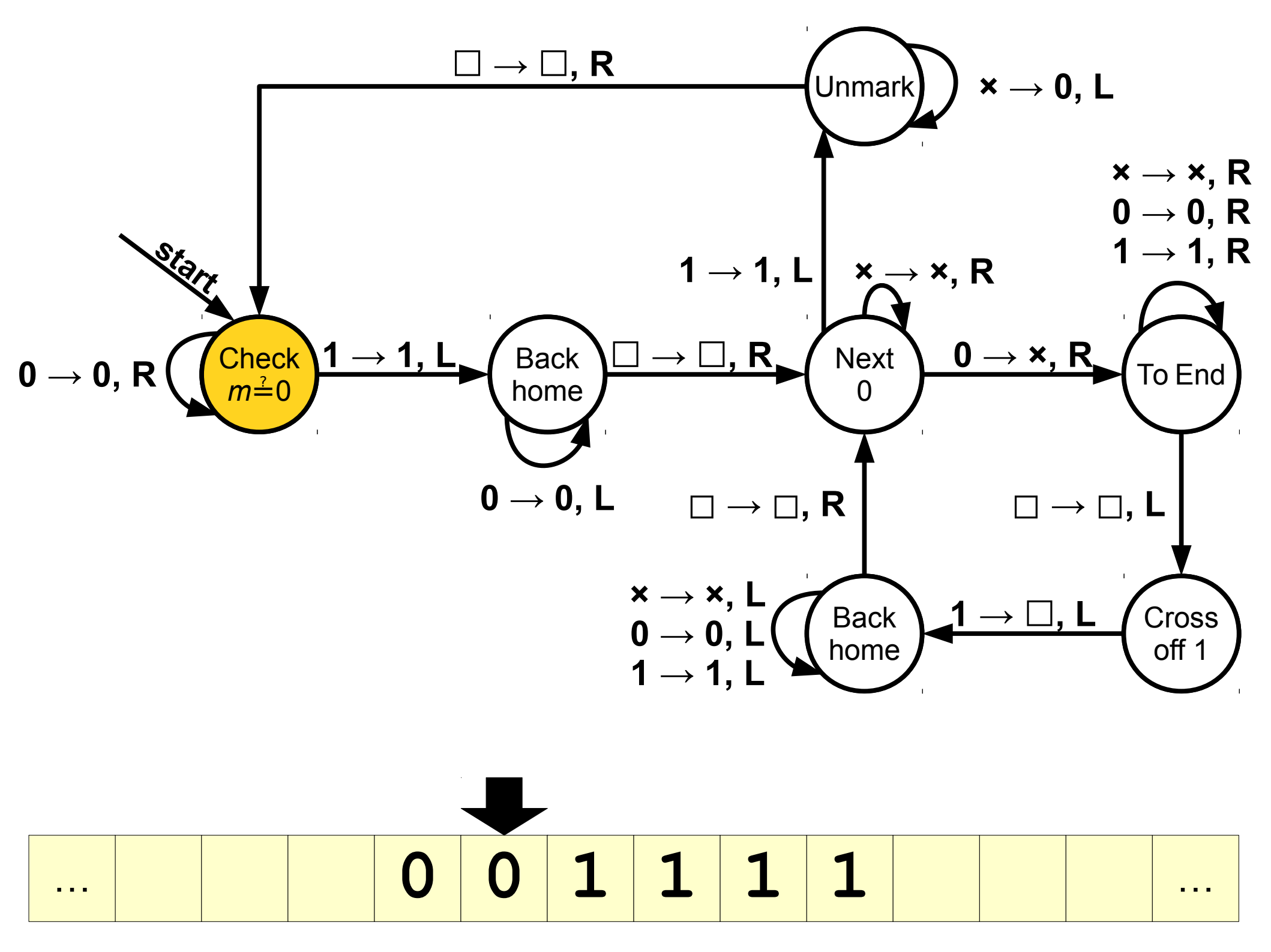

Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m=?0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 1 1 …

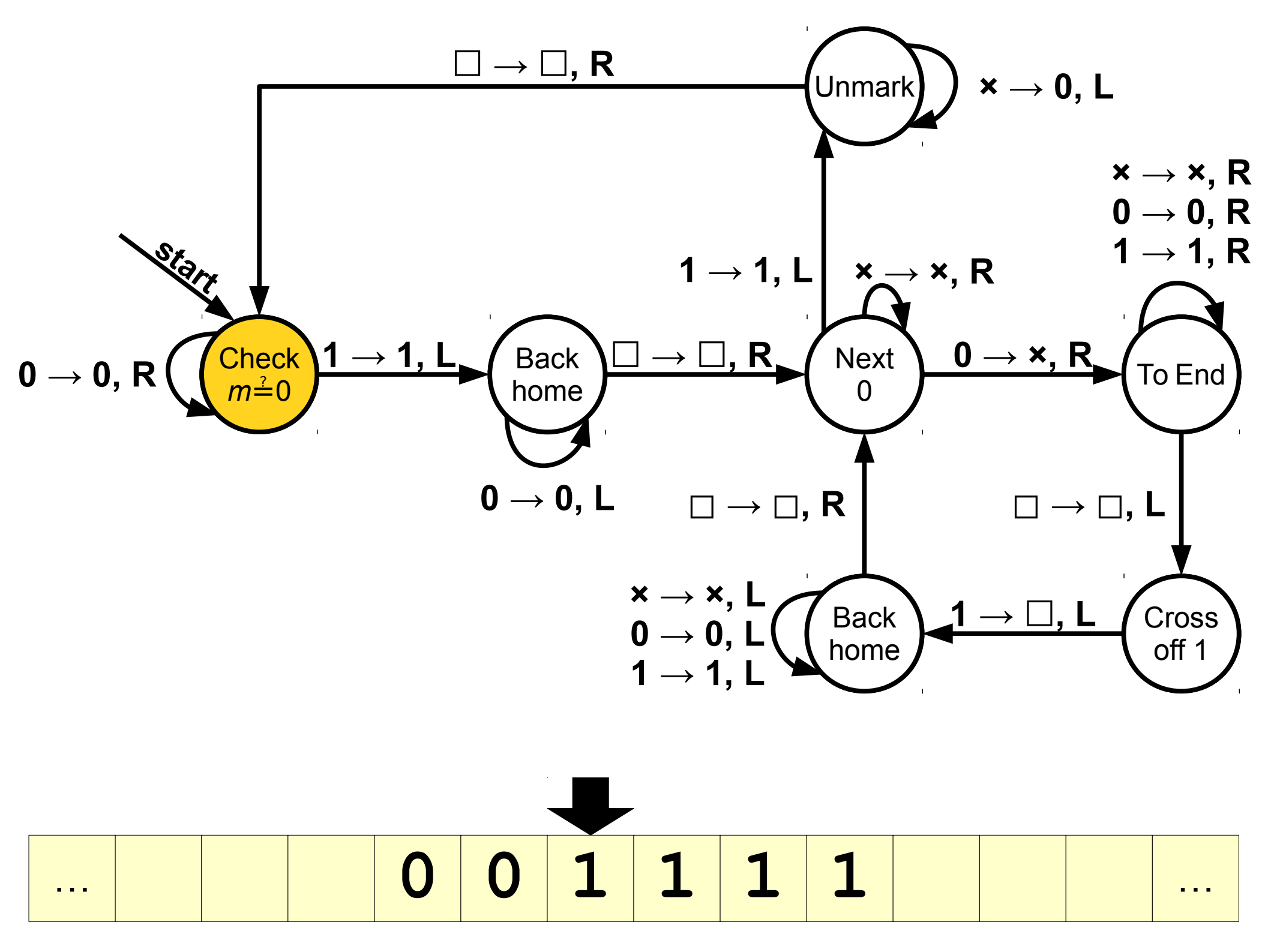

Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
1
1
…

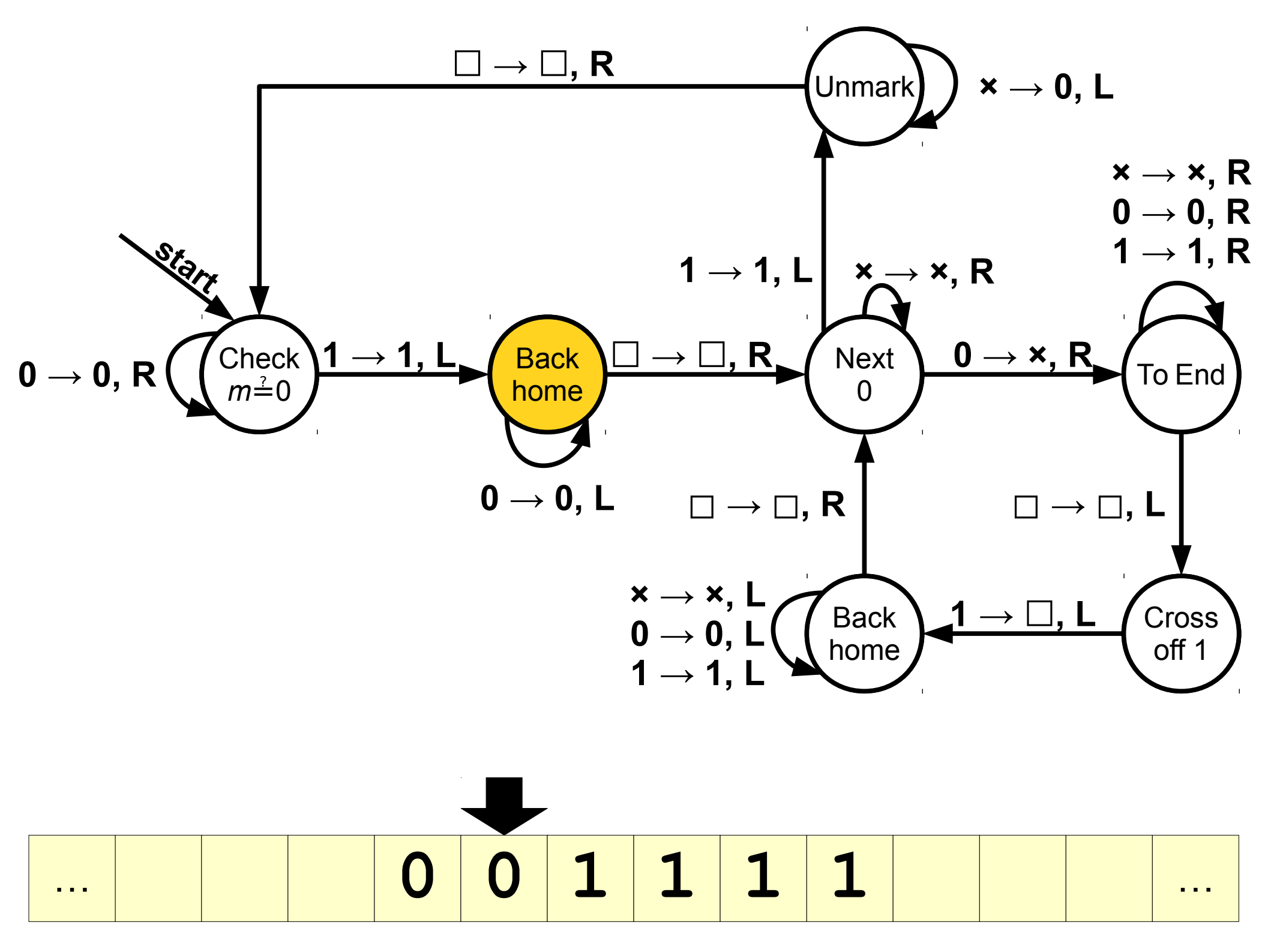

Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… 0 0 1 1 1 1 …

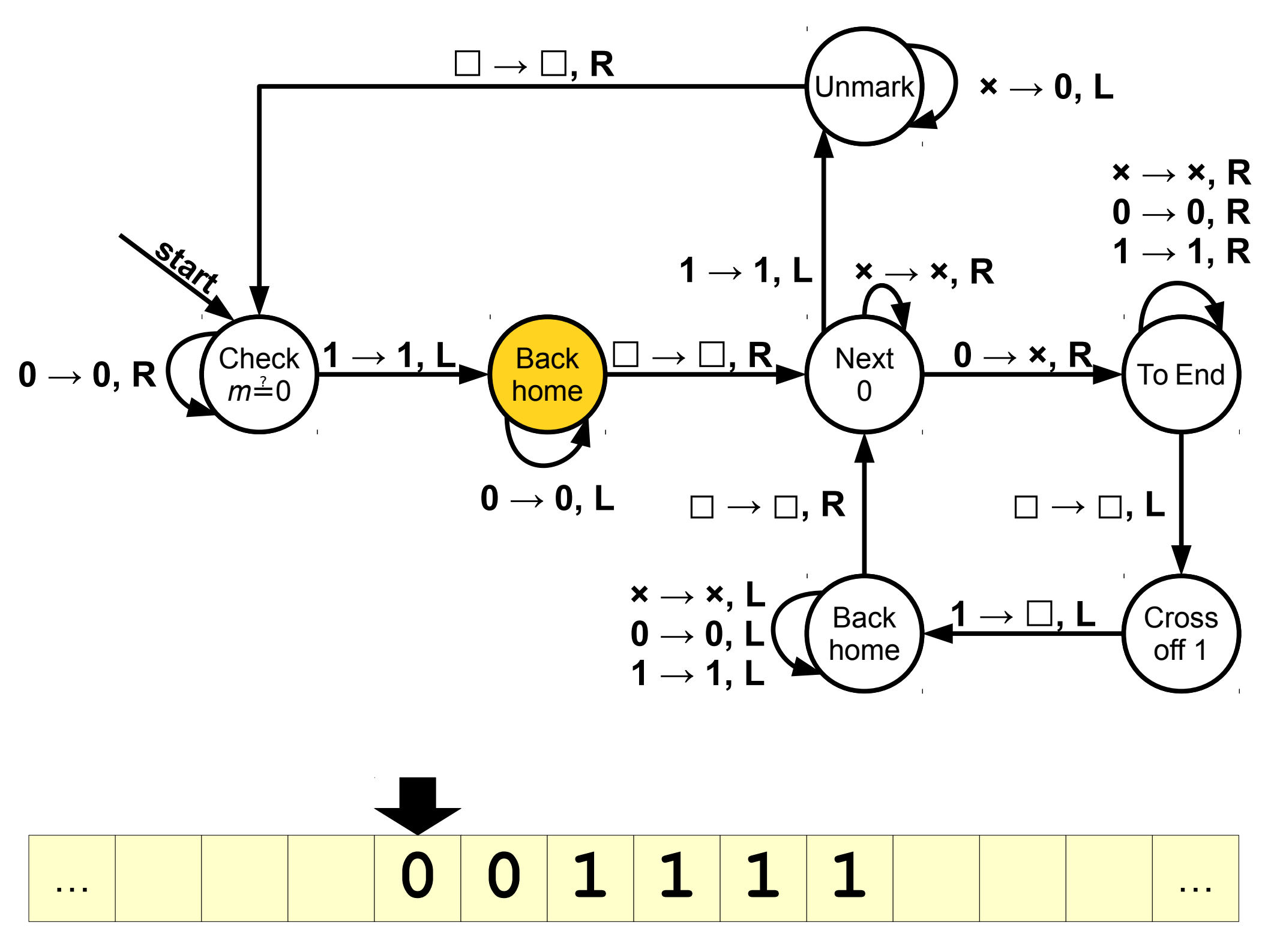

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0 0 1 1 1 1
…

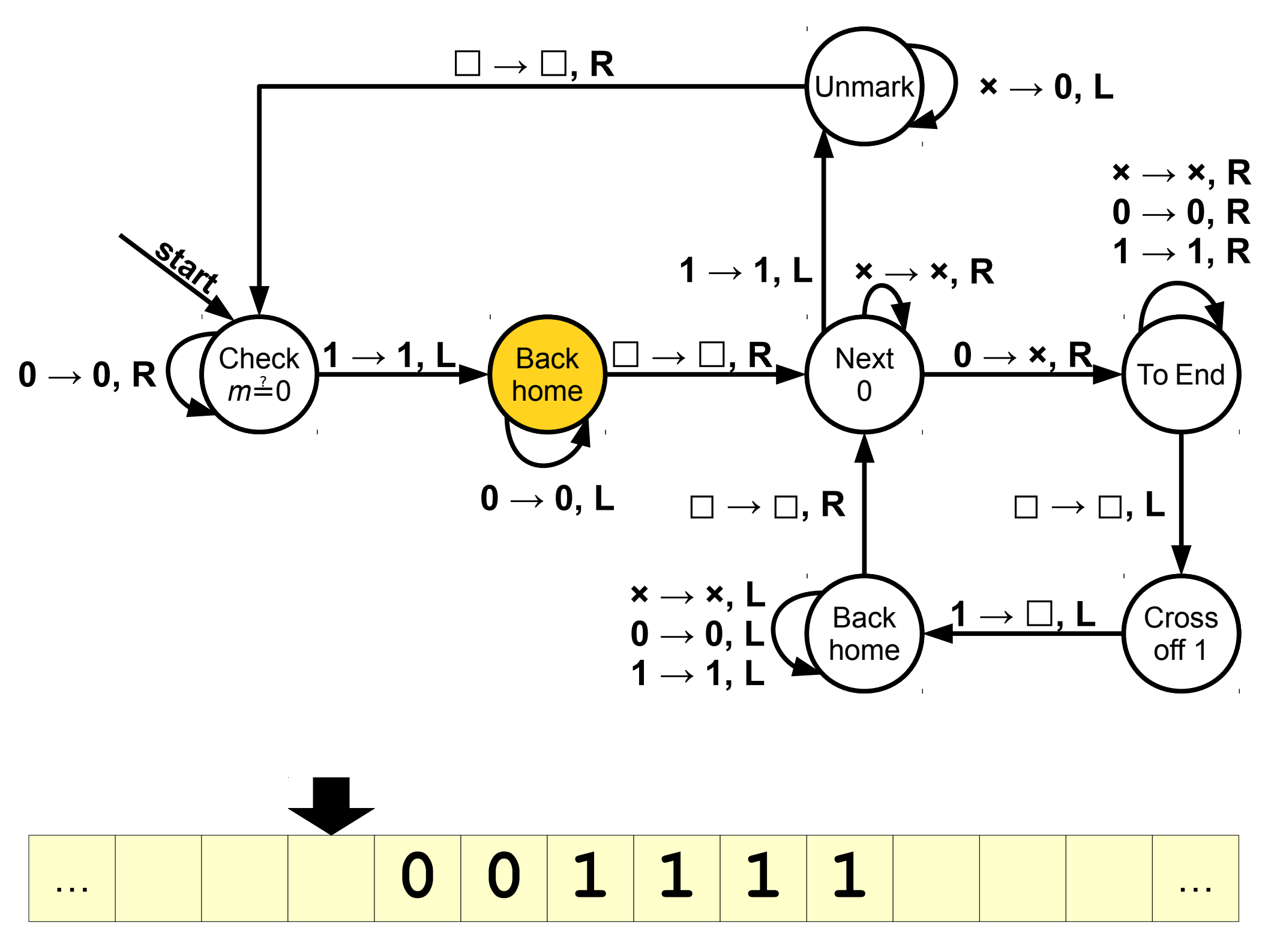

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
1
1
…

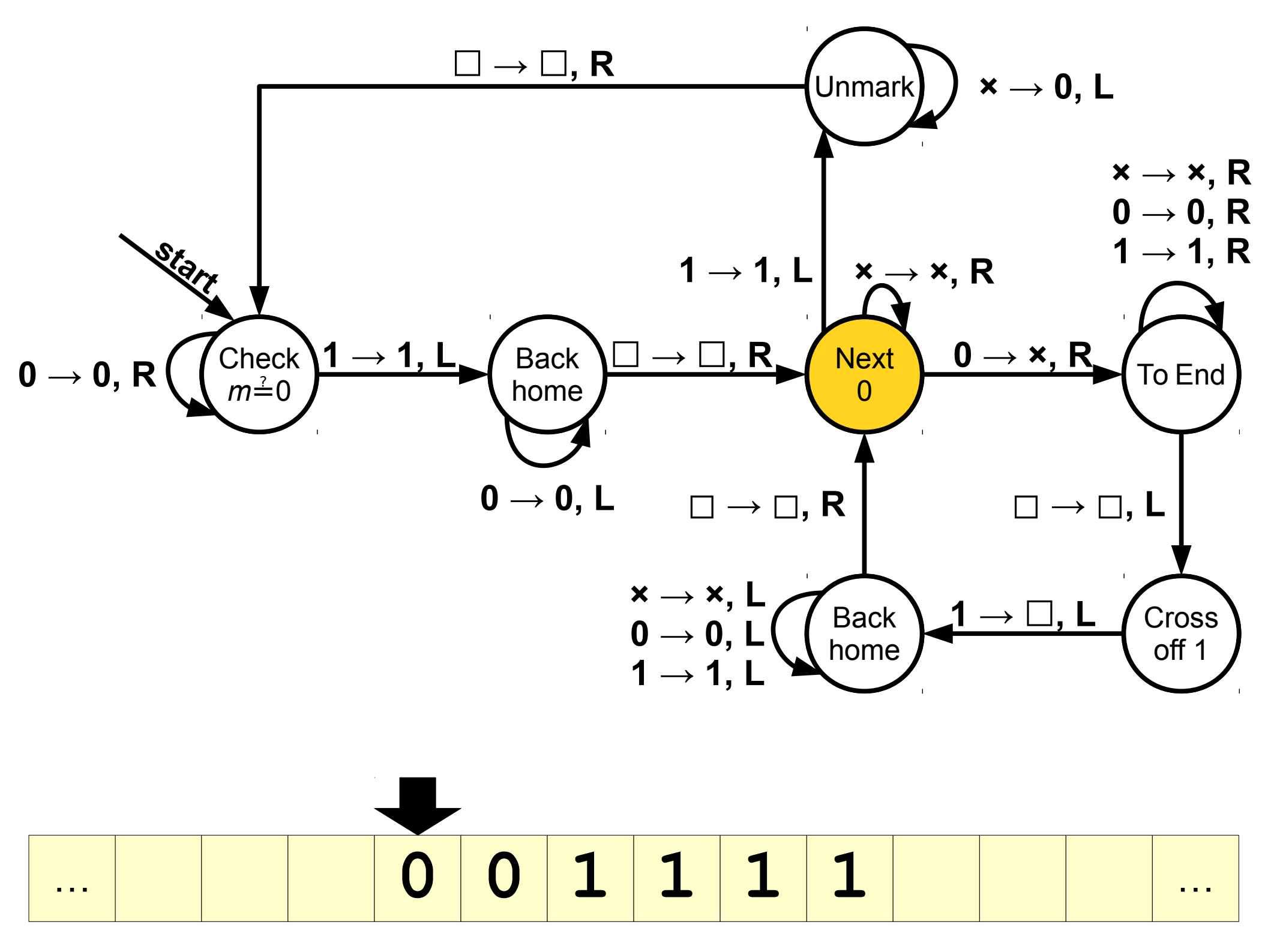

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0 0 1 1 1 1
…

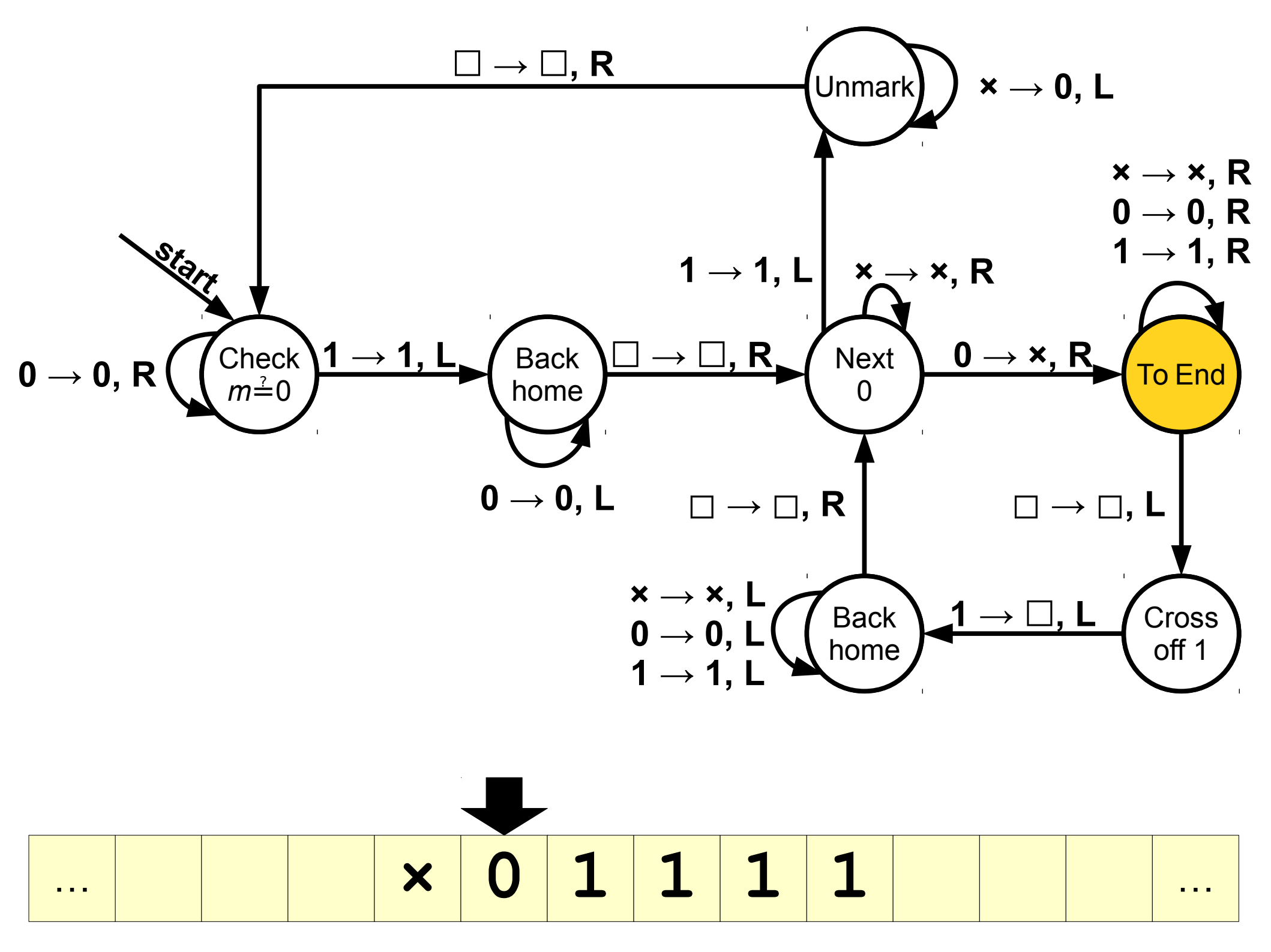

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
1
1
…

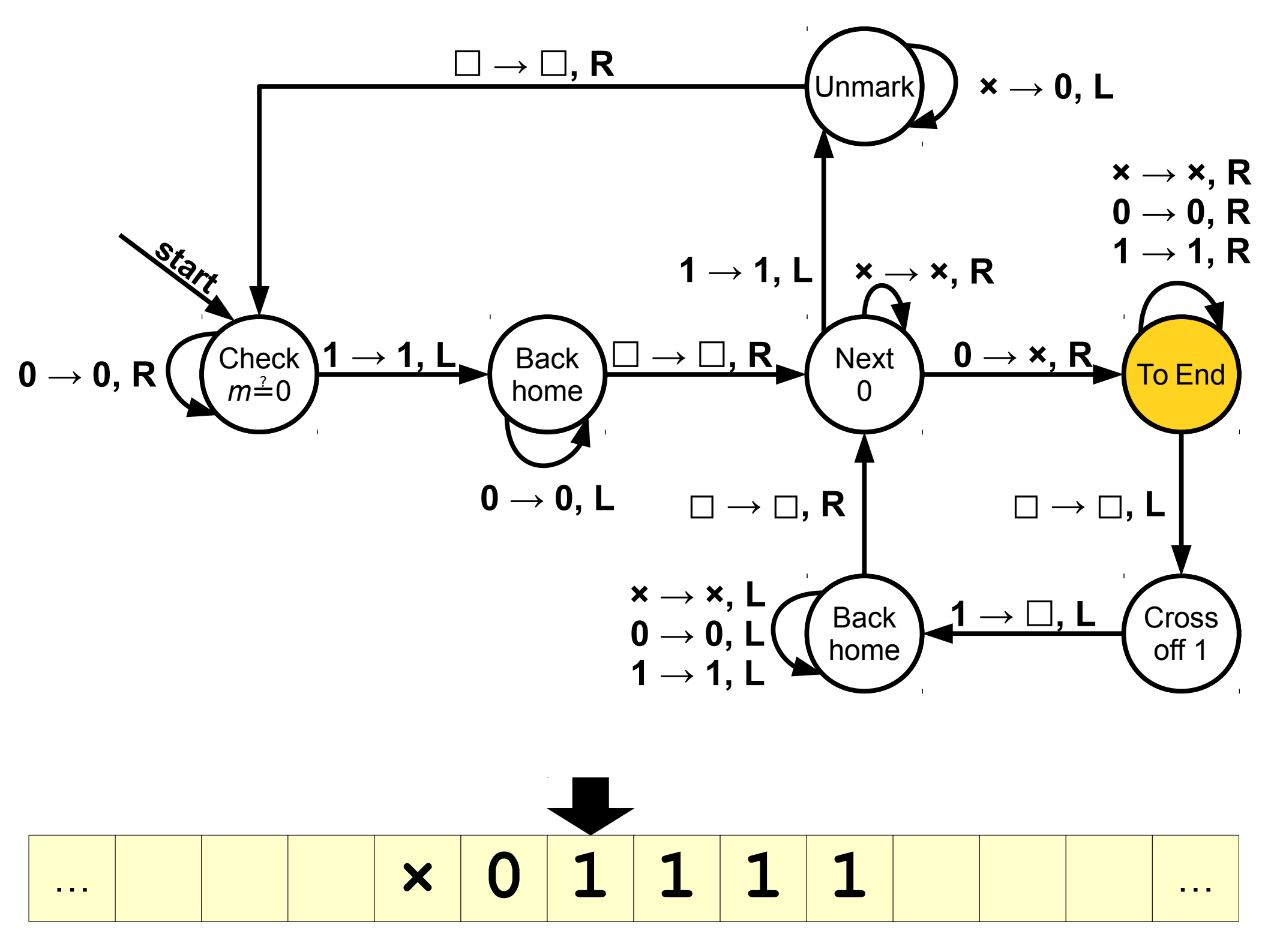

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0 0 1 1 1 1
…

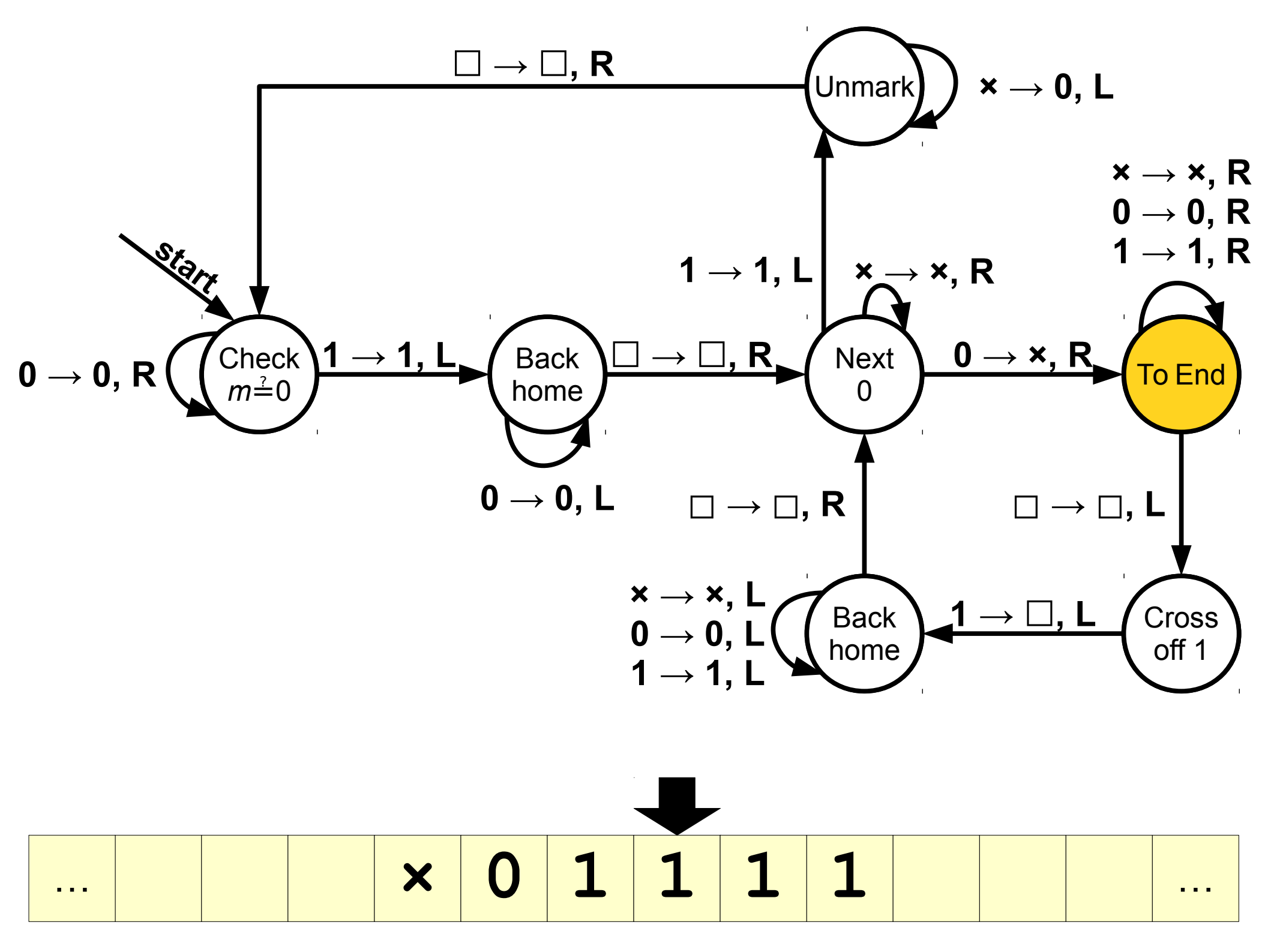

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
1
1
…

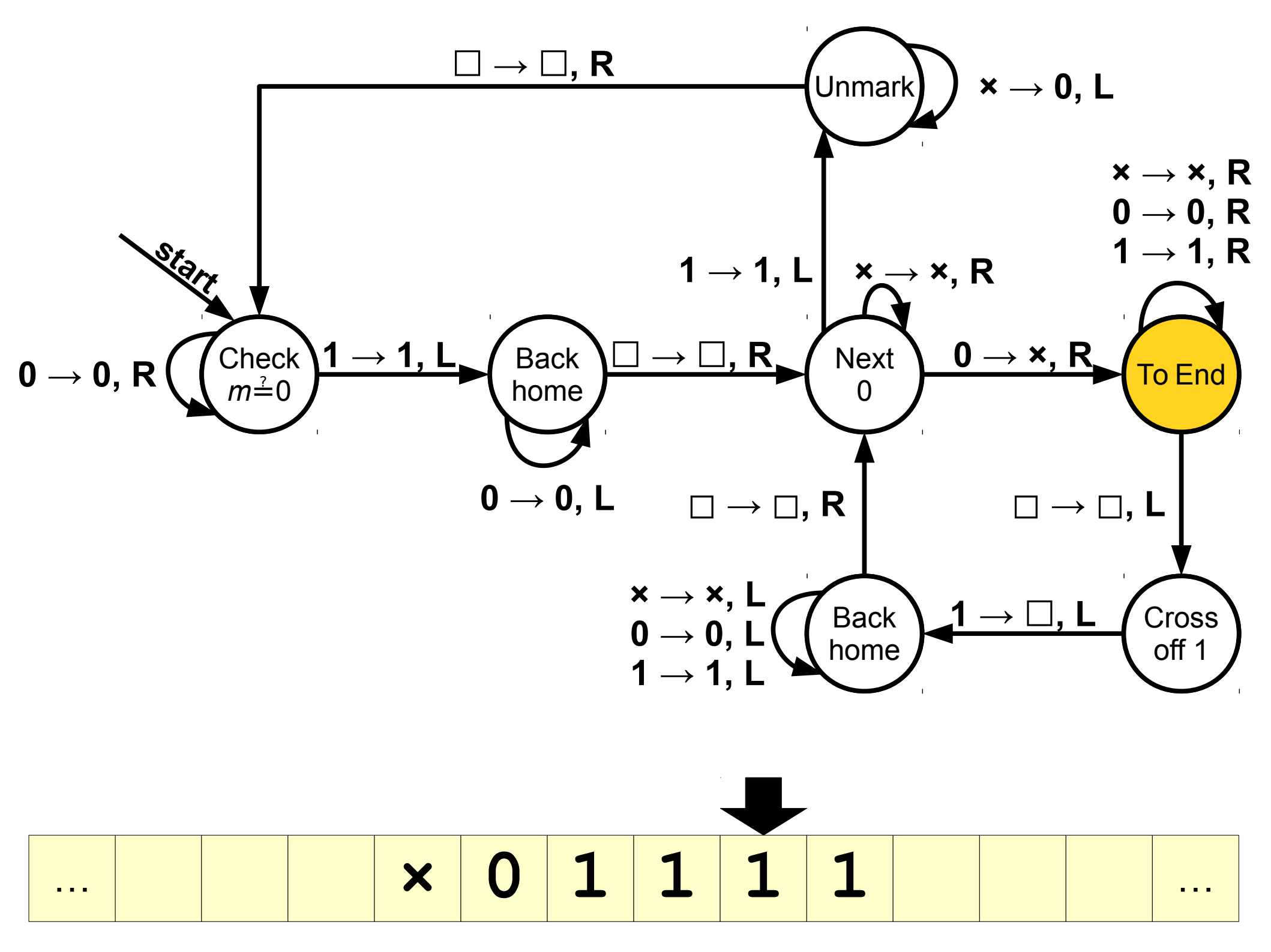

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
1
1
…

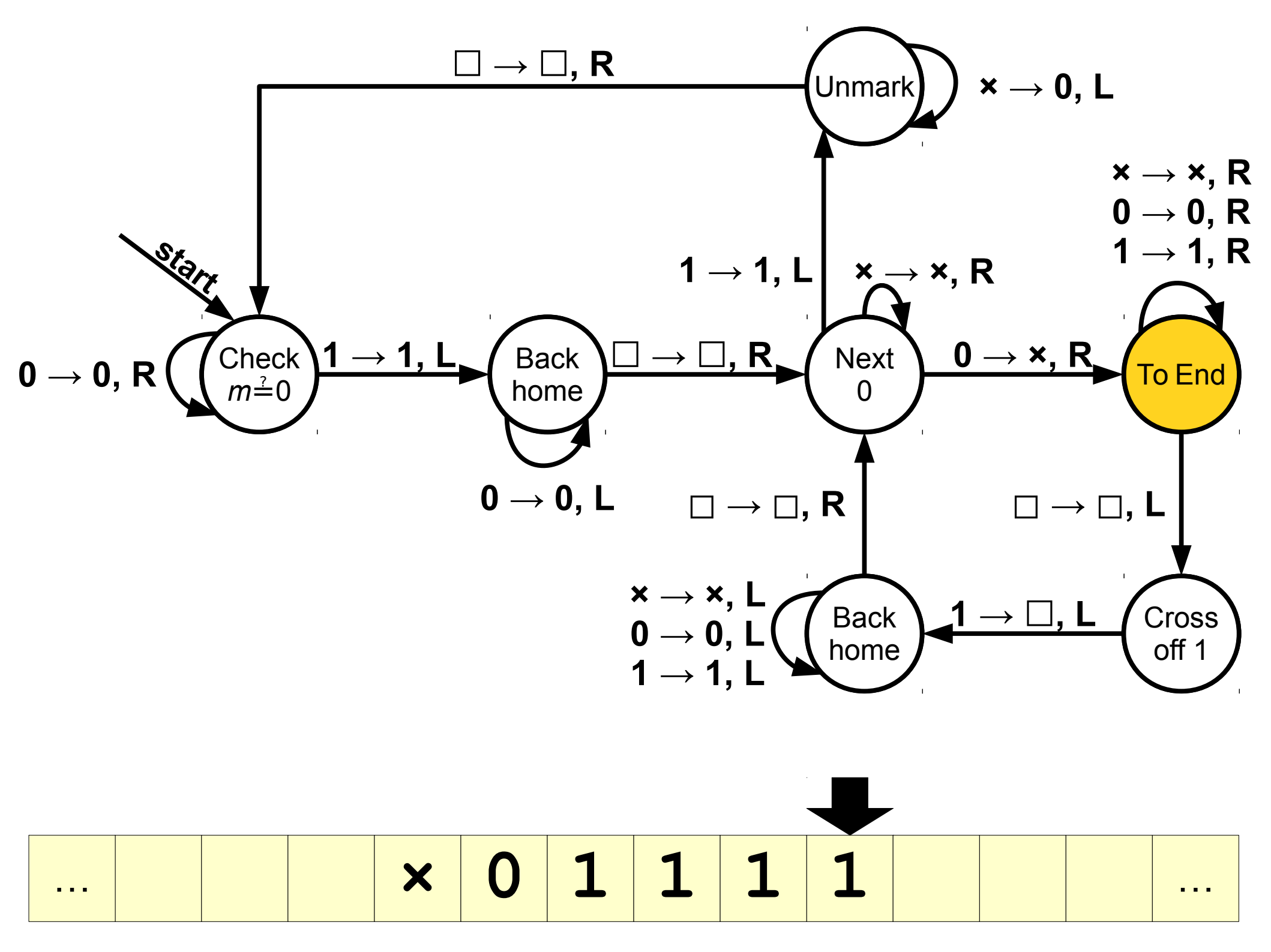

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
1
1
…

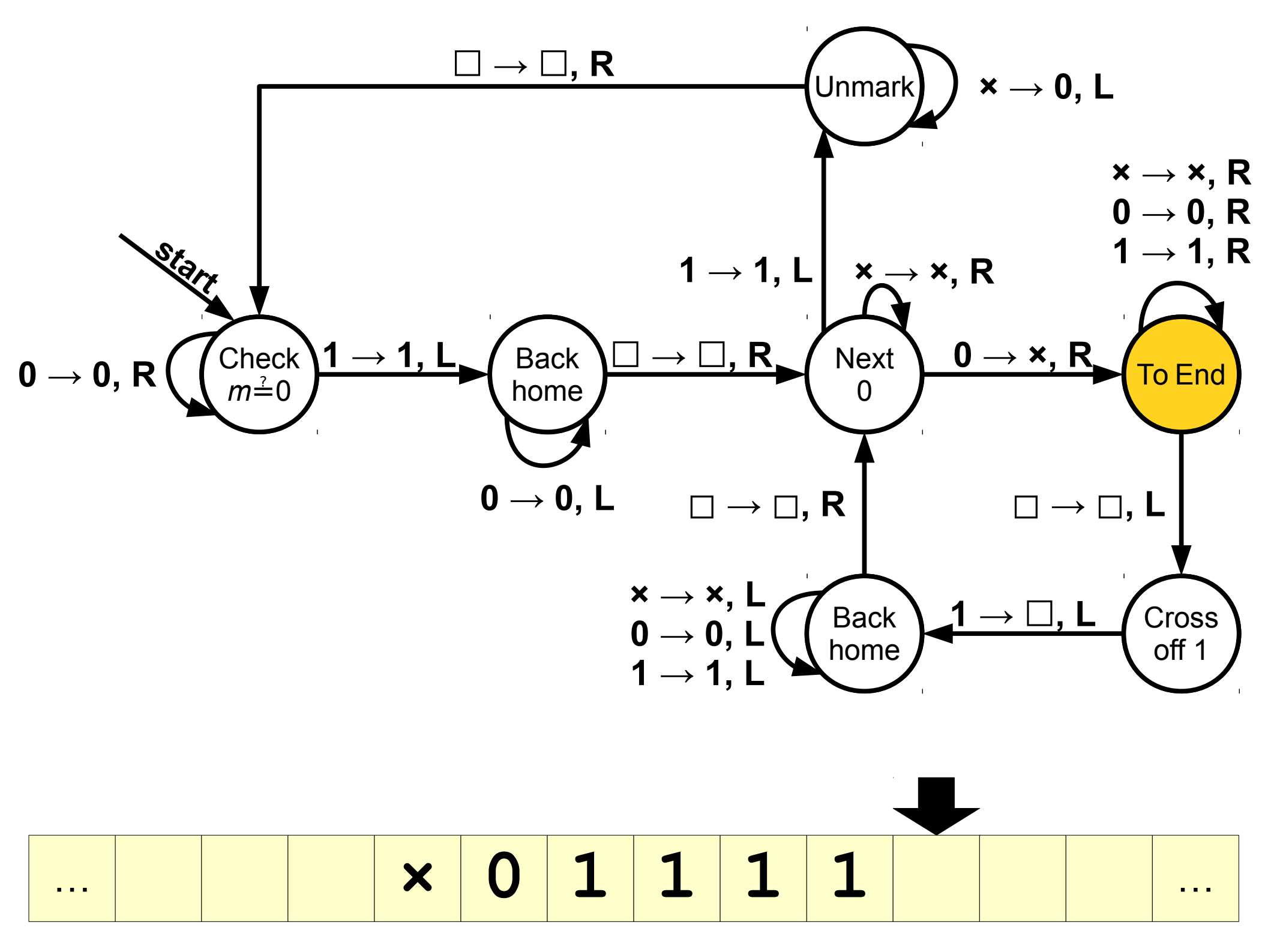

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

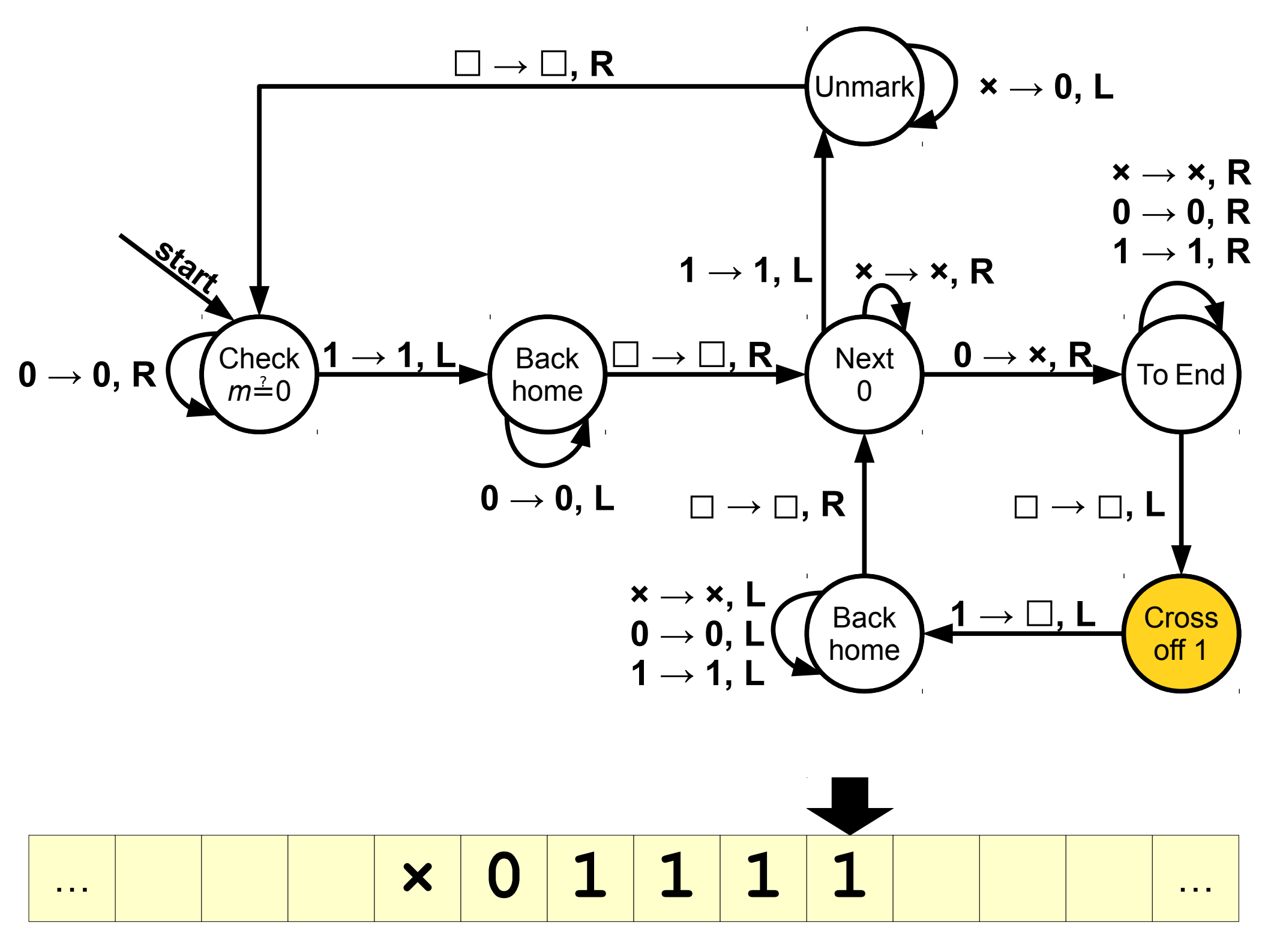

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

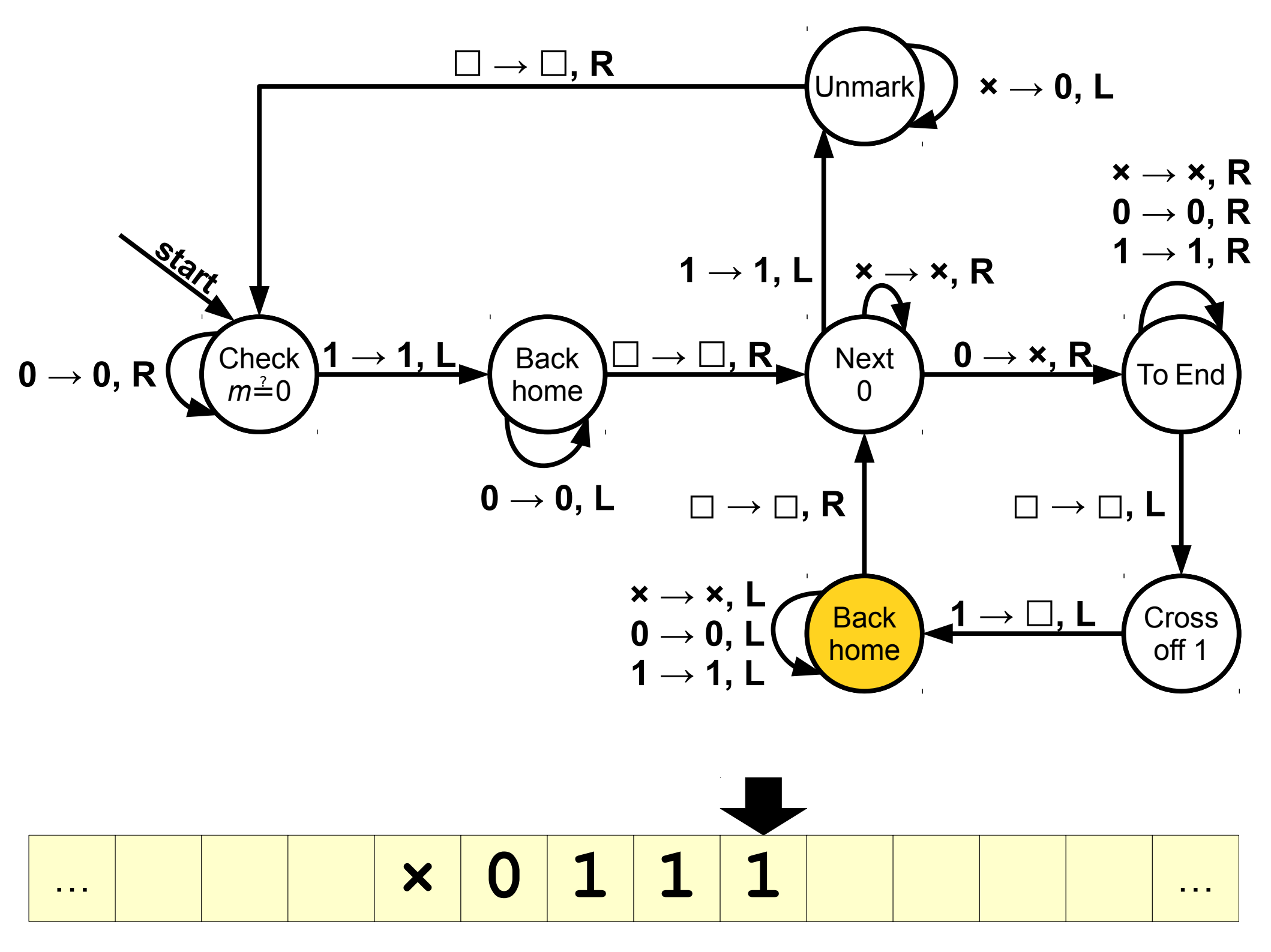

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

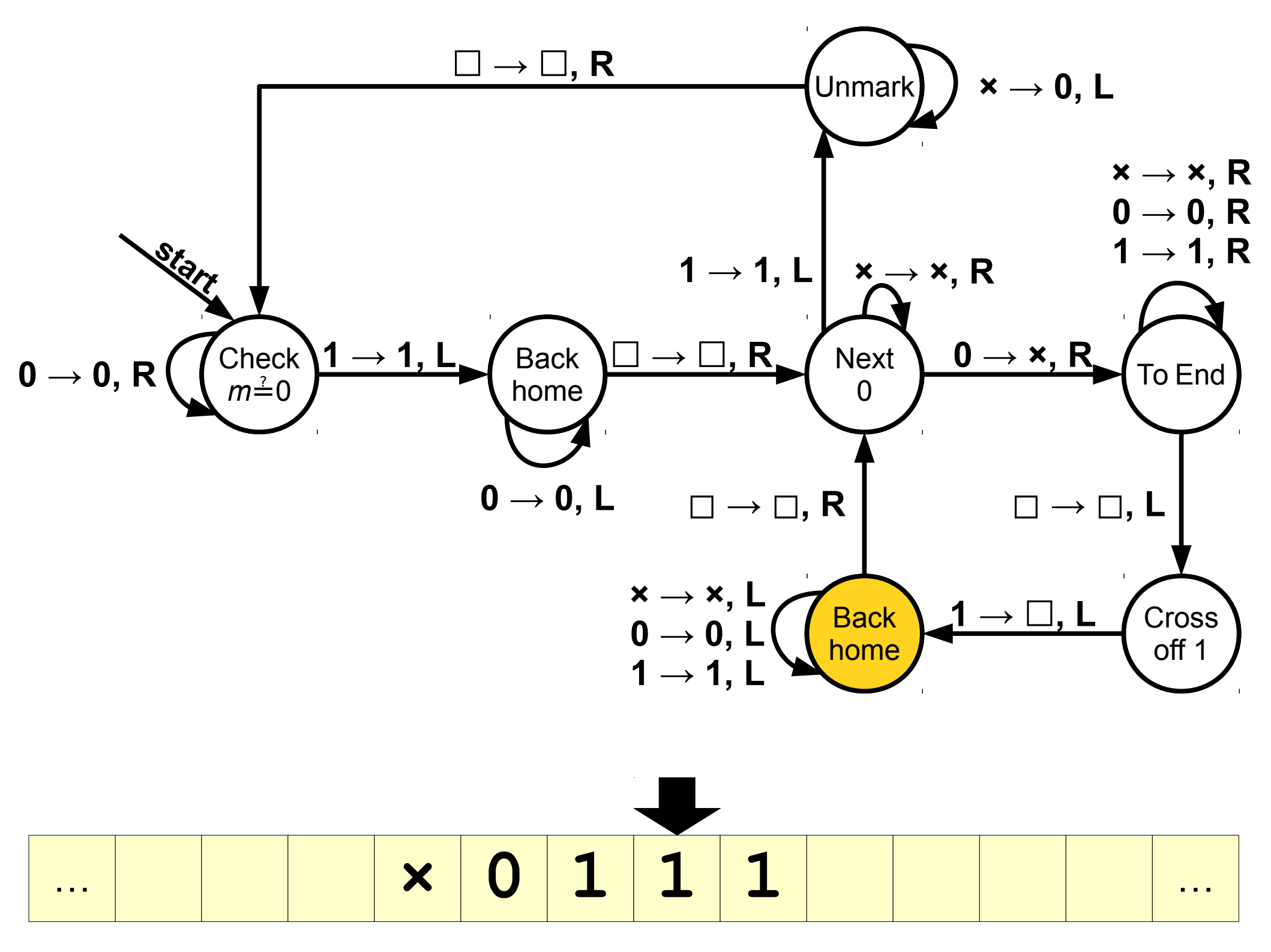

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

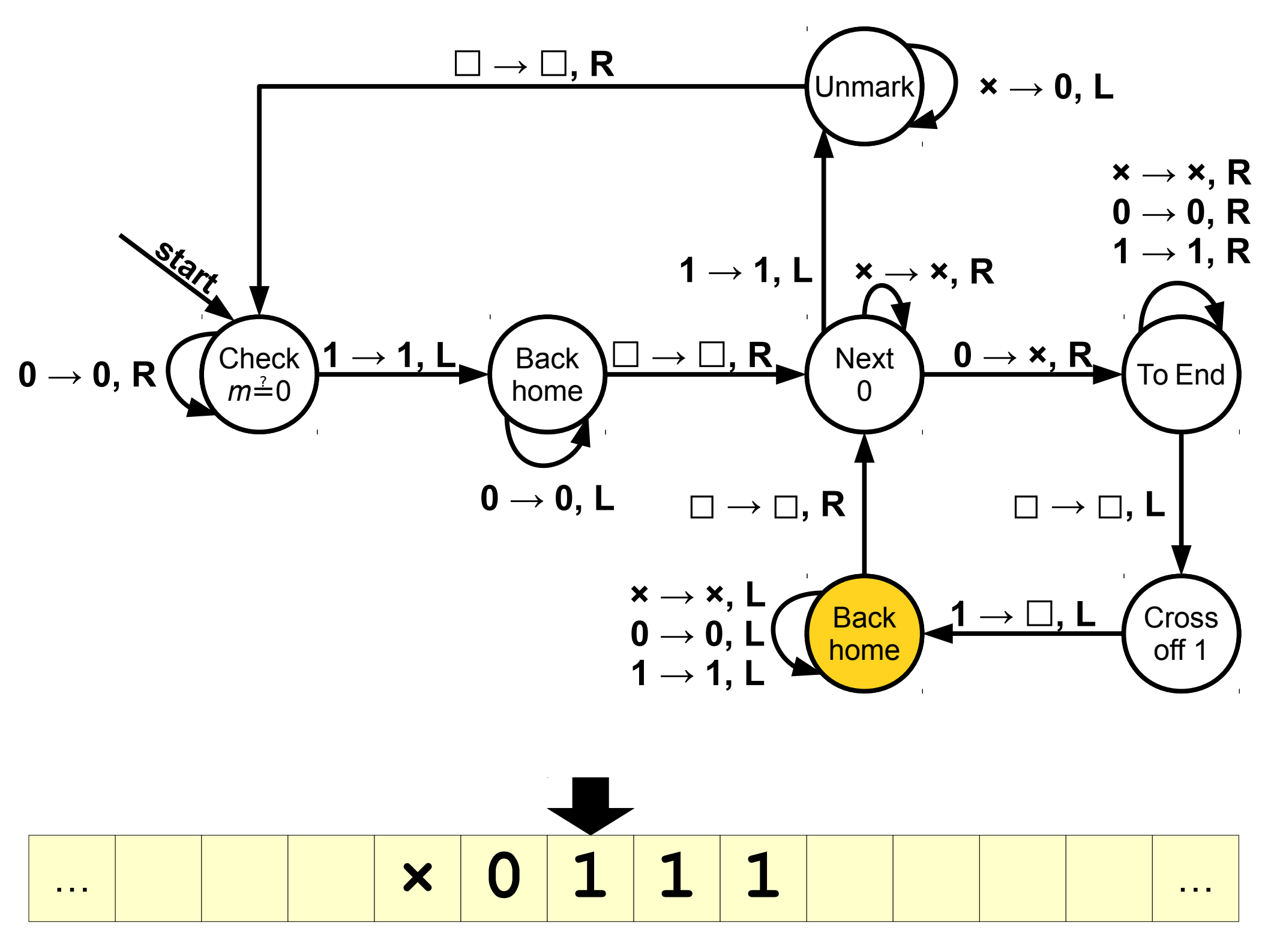

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

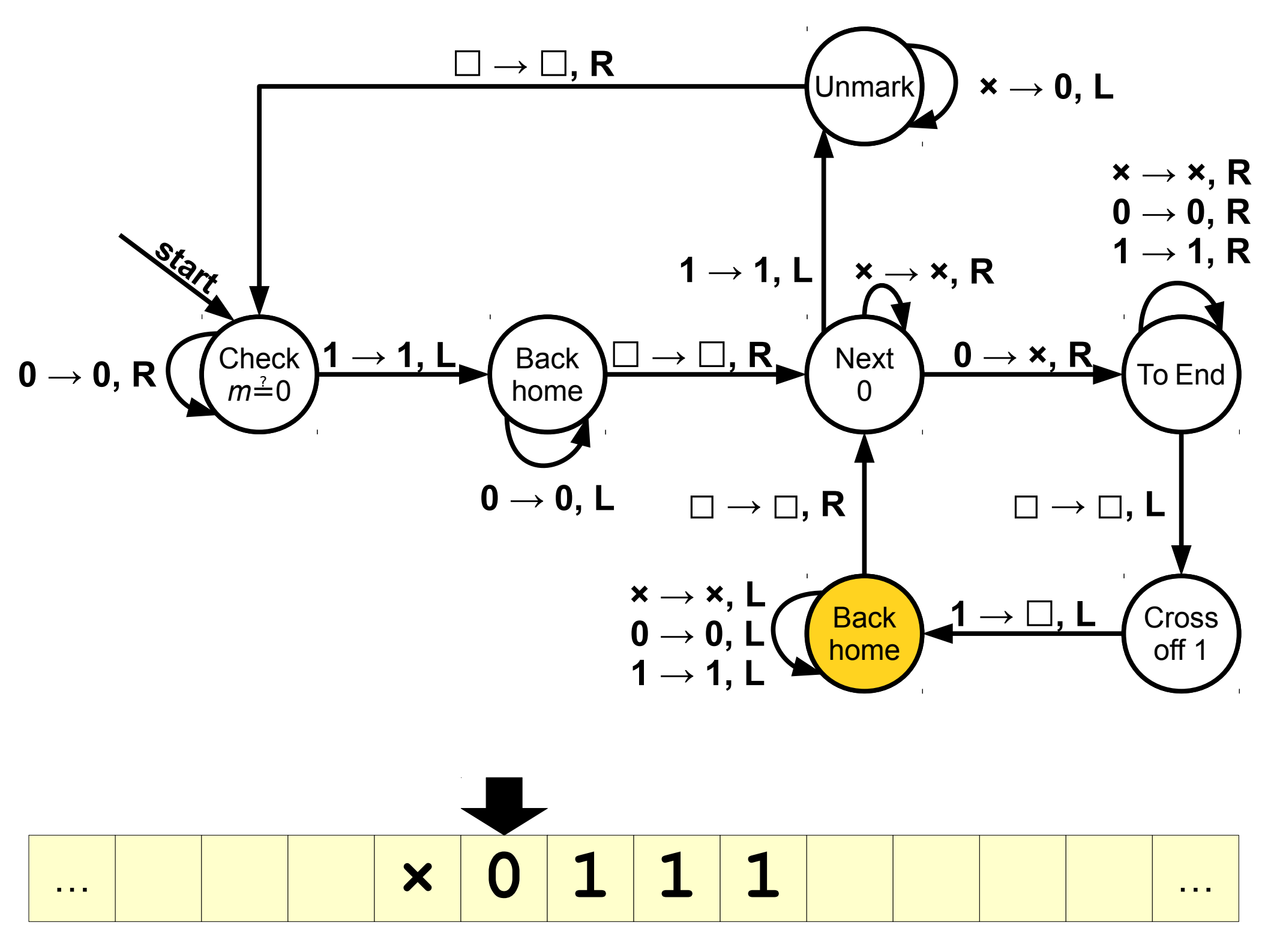

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

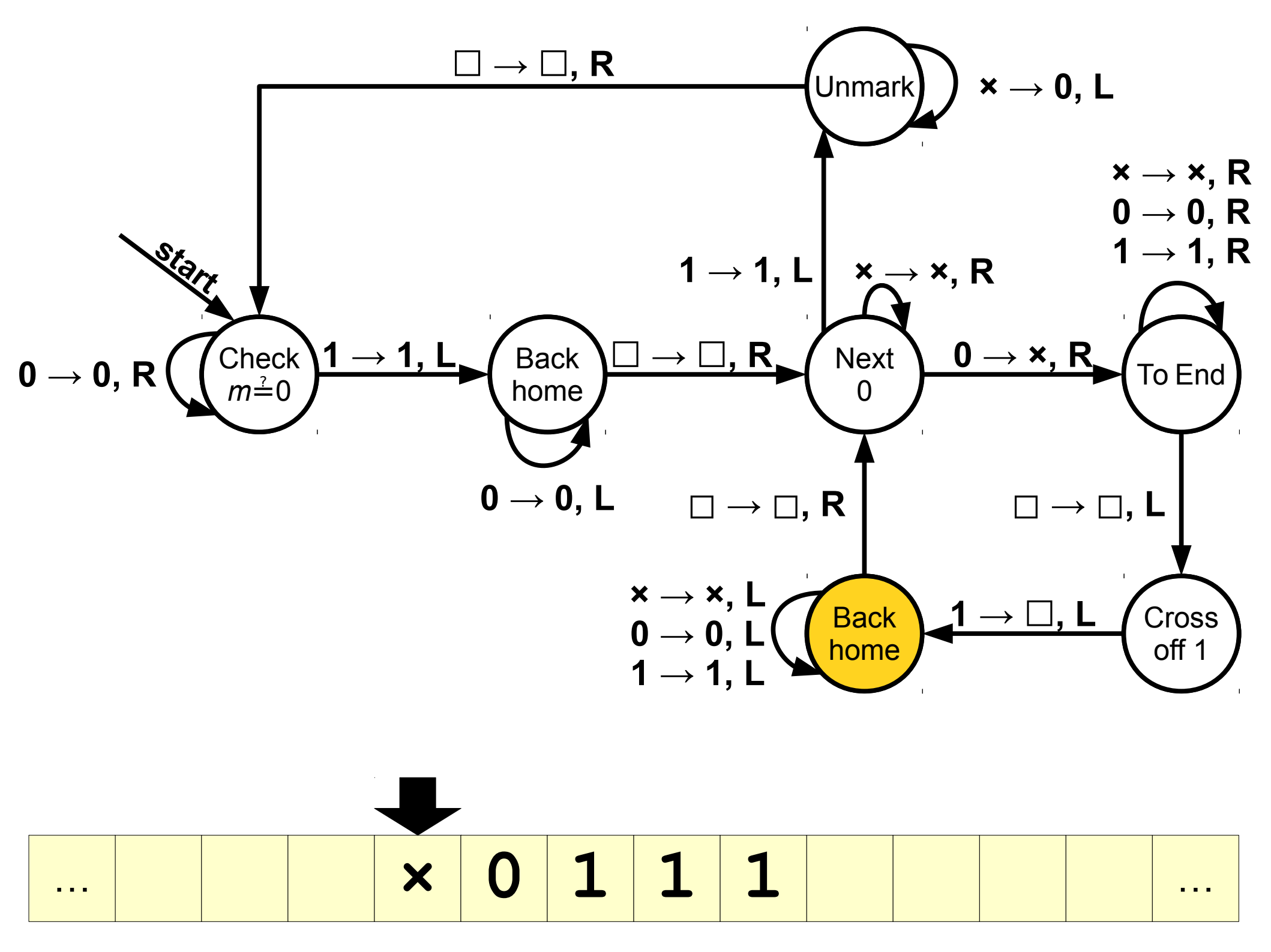

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 1 …

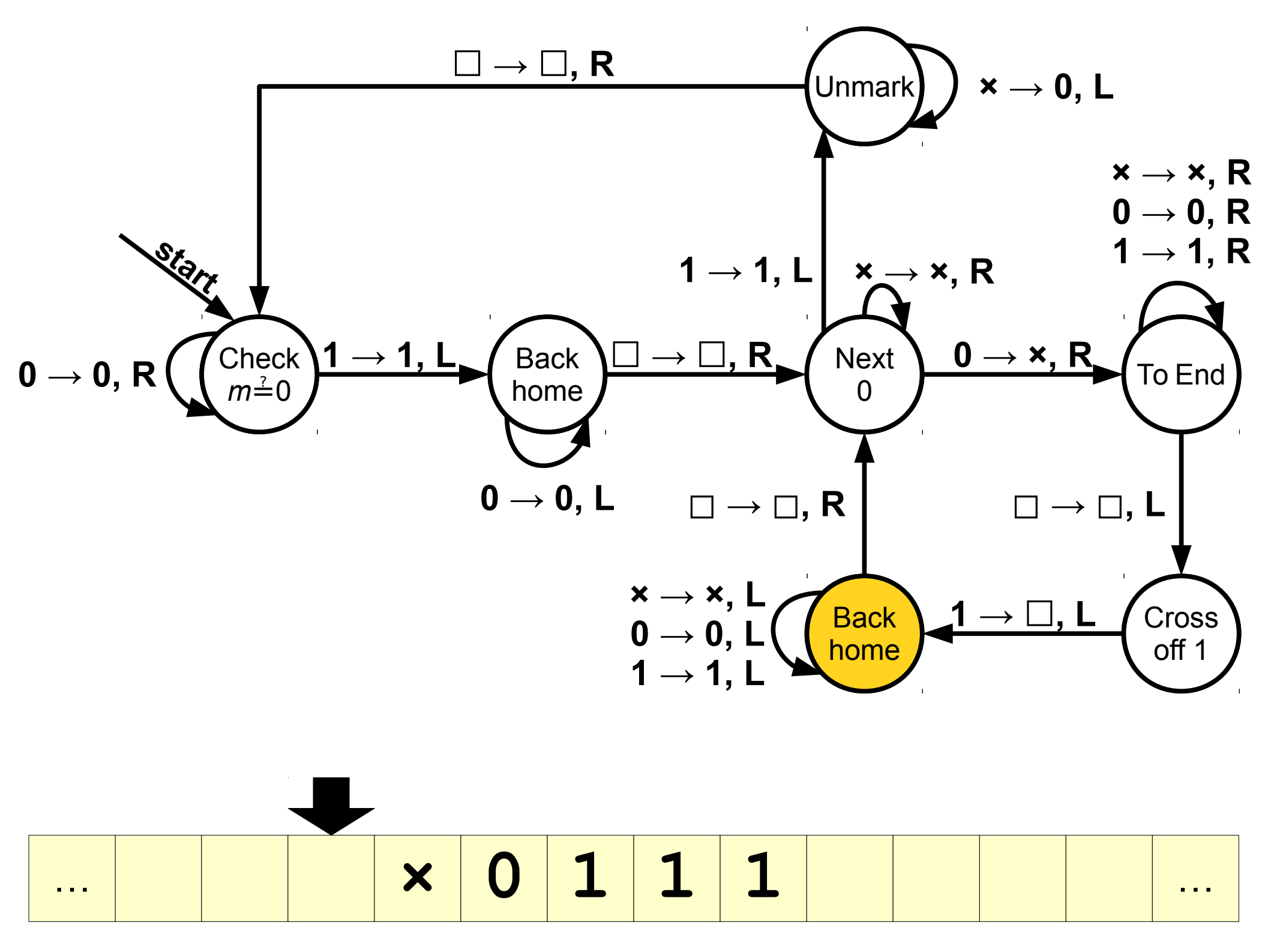

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 …

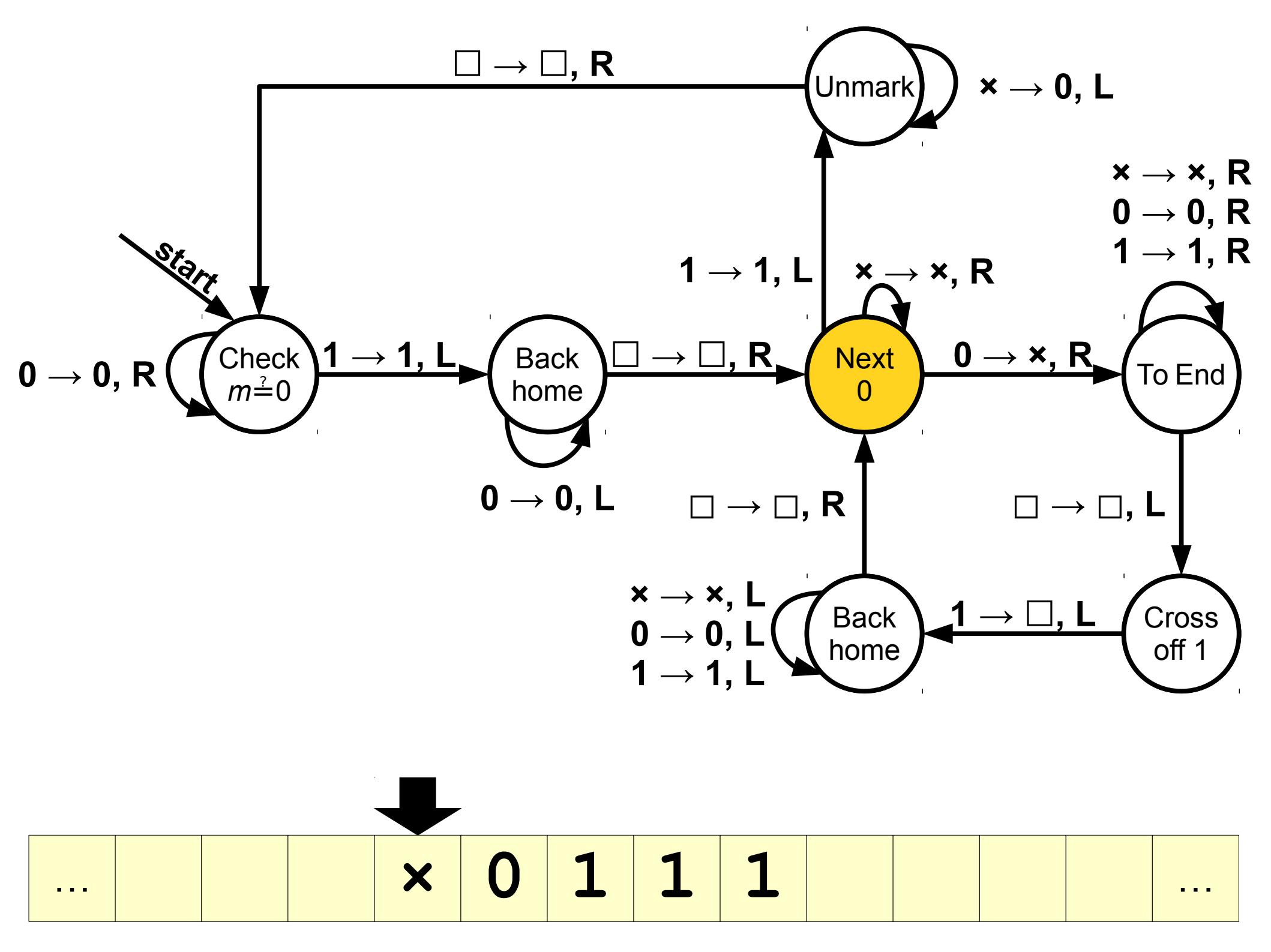

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 …

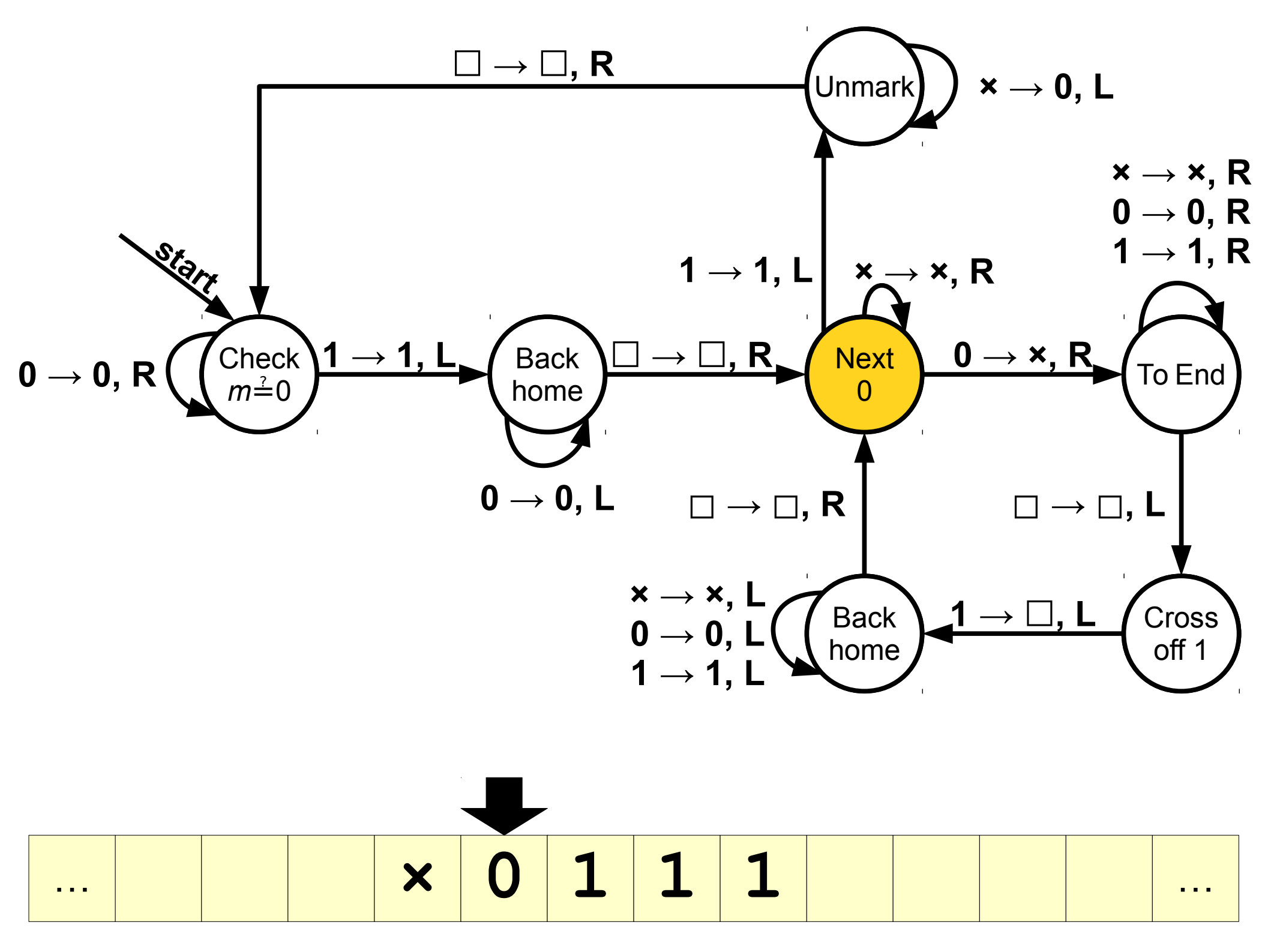

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 …

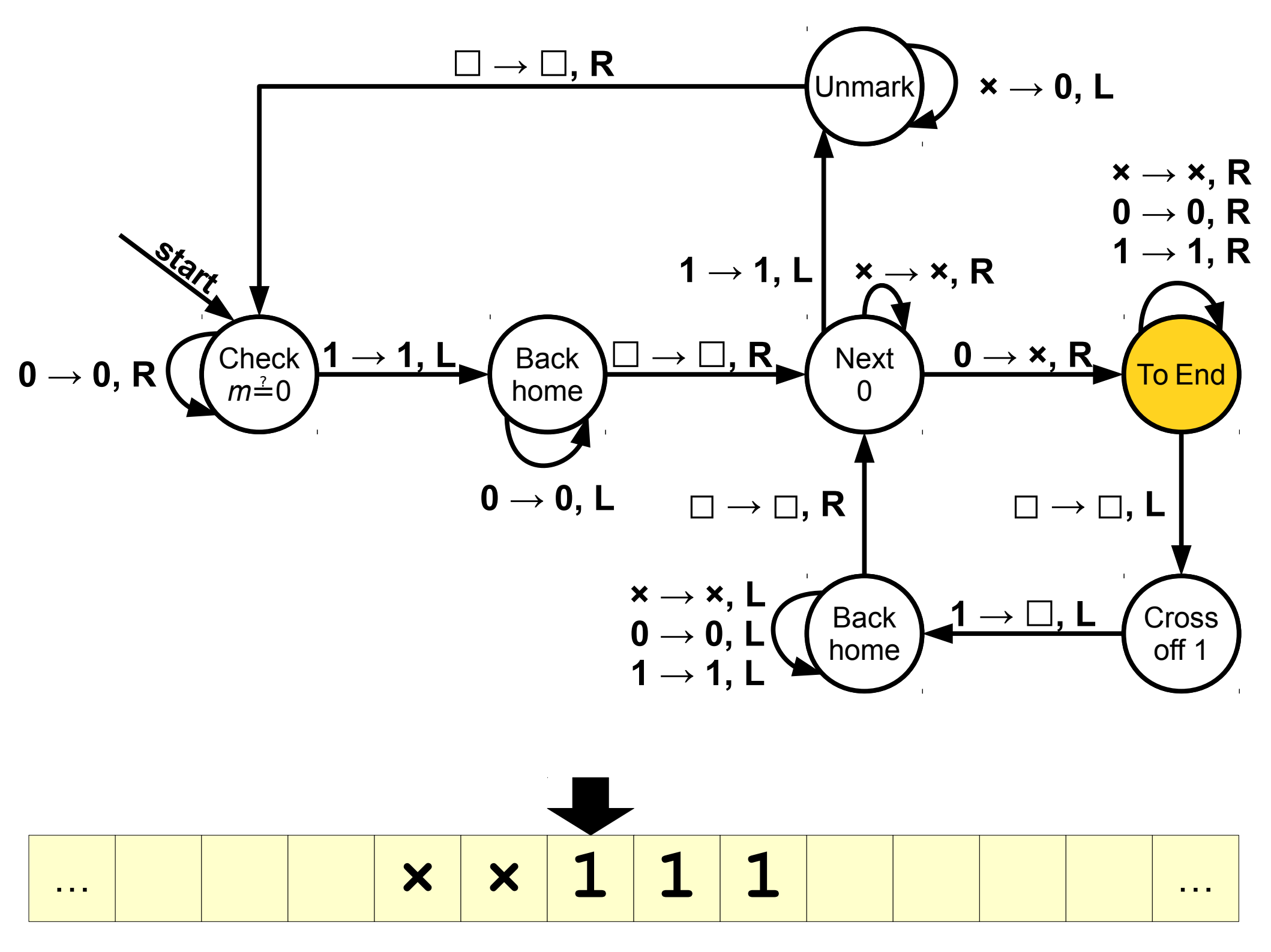

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
1
…

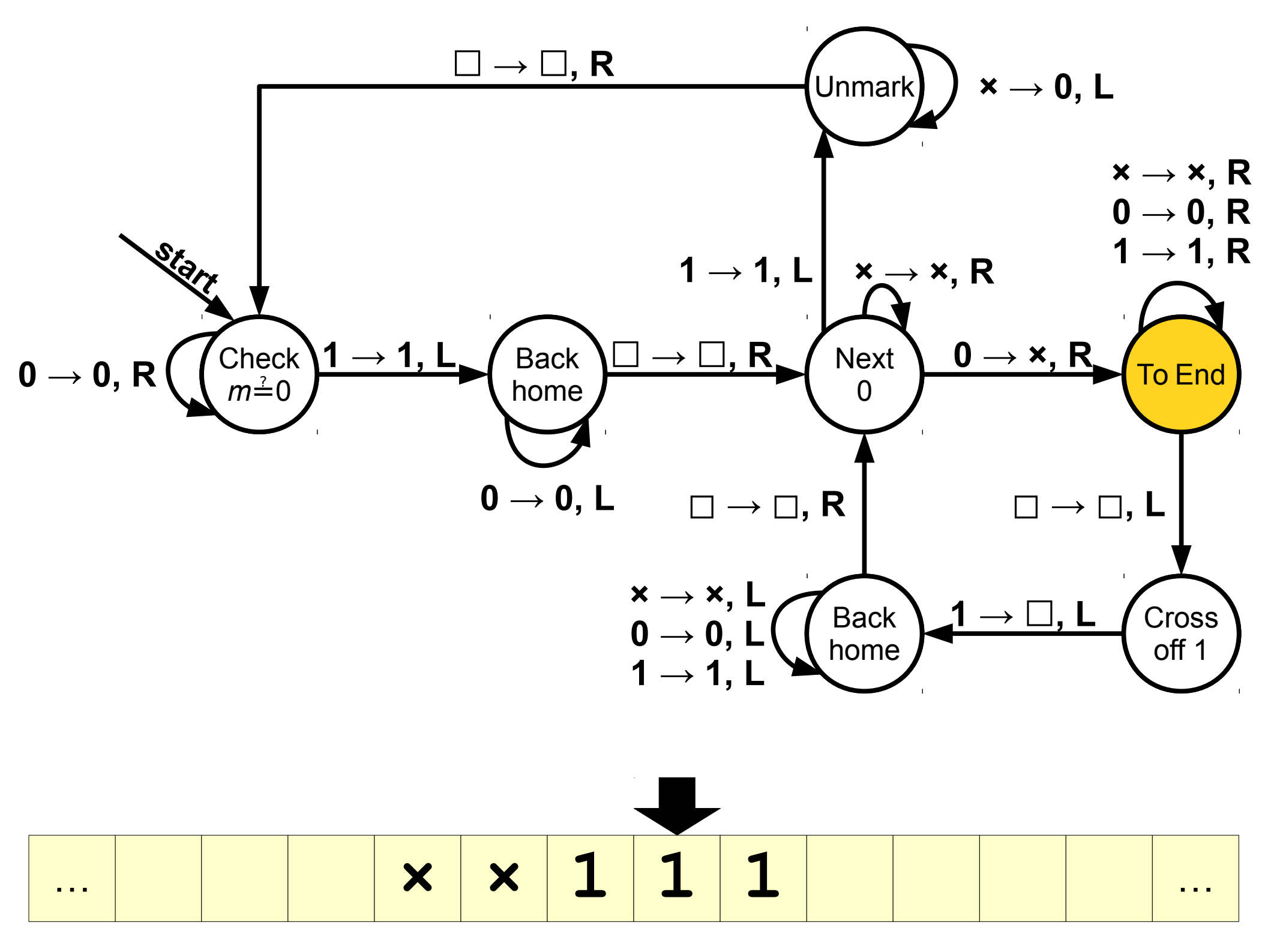

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
1
…

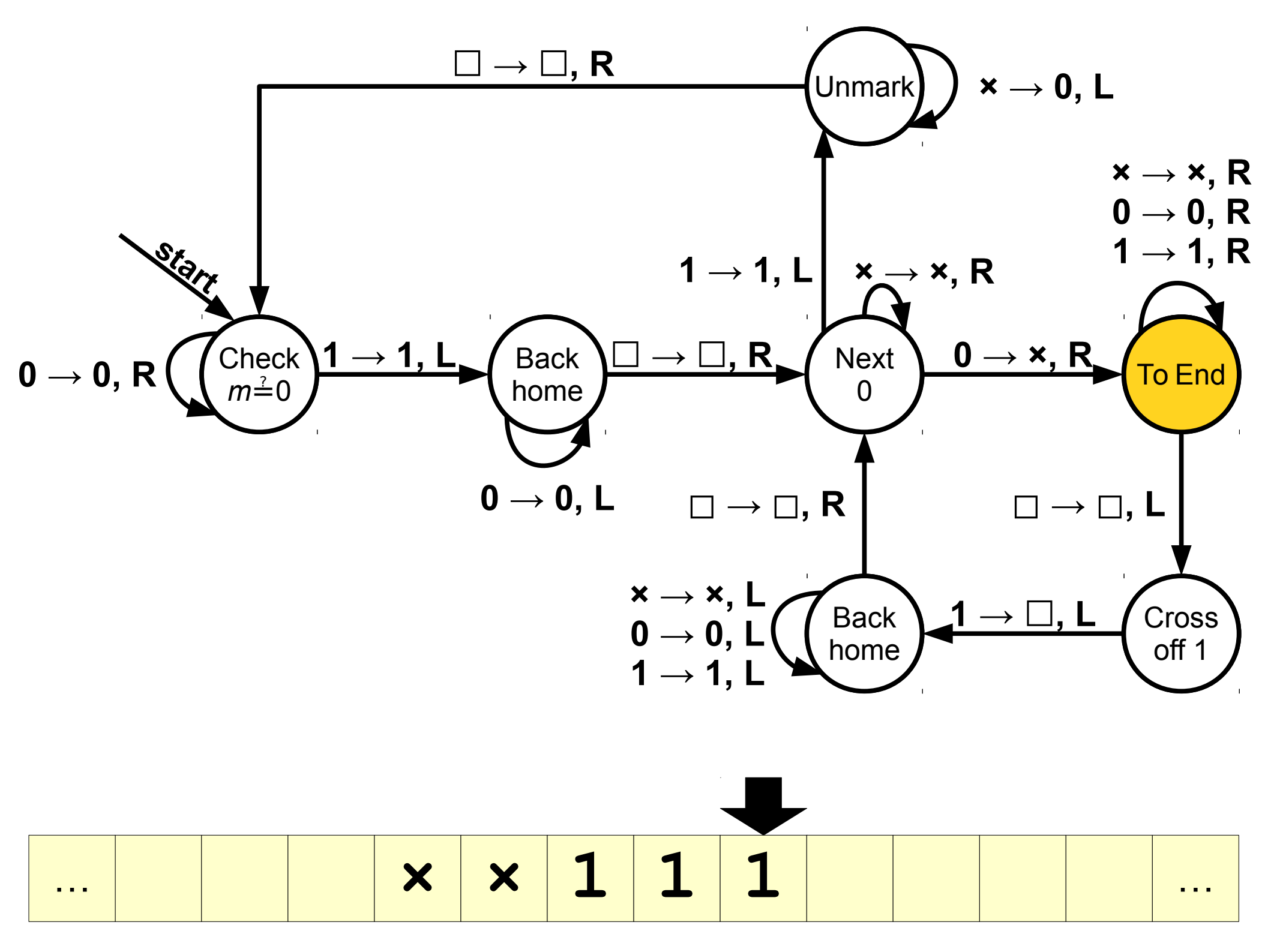

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
1
…

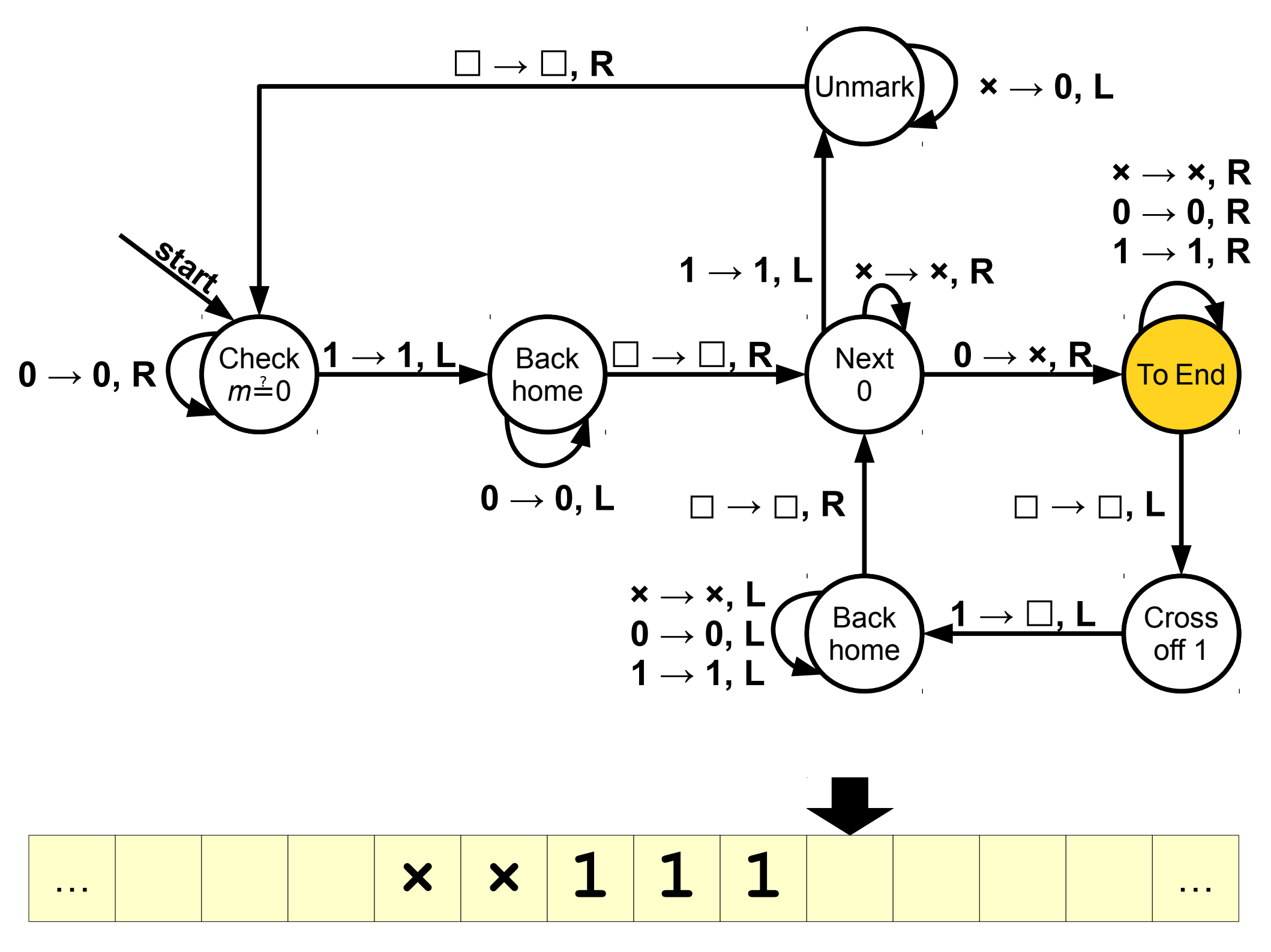

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 …

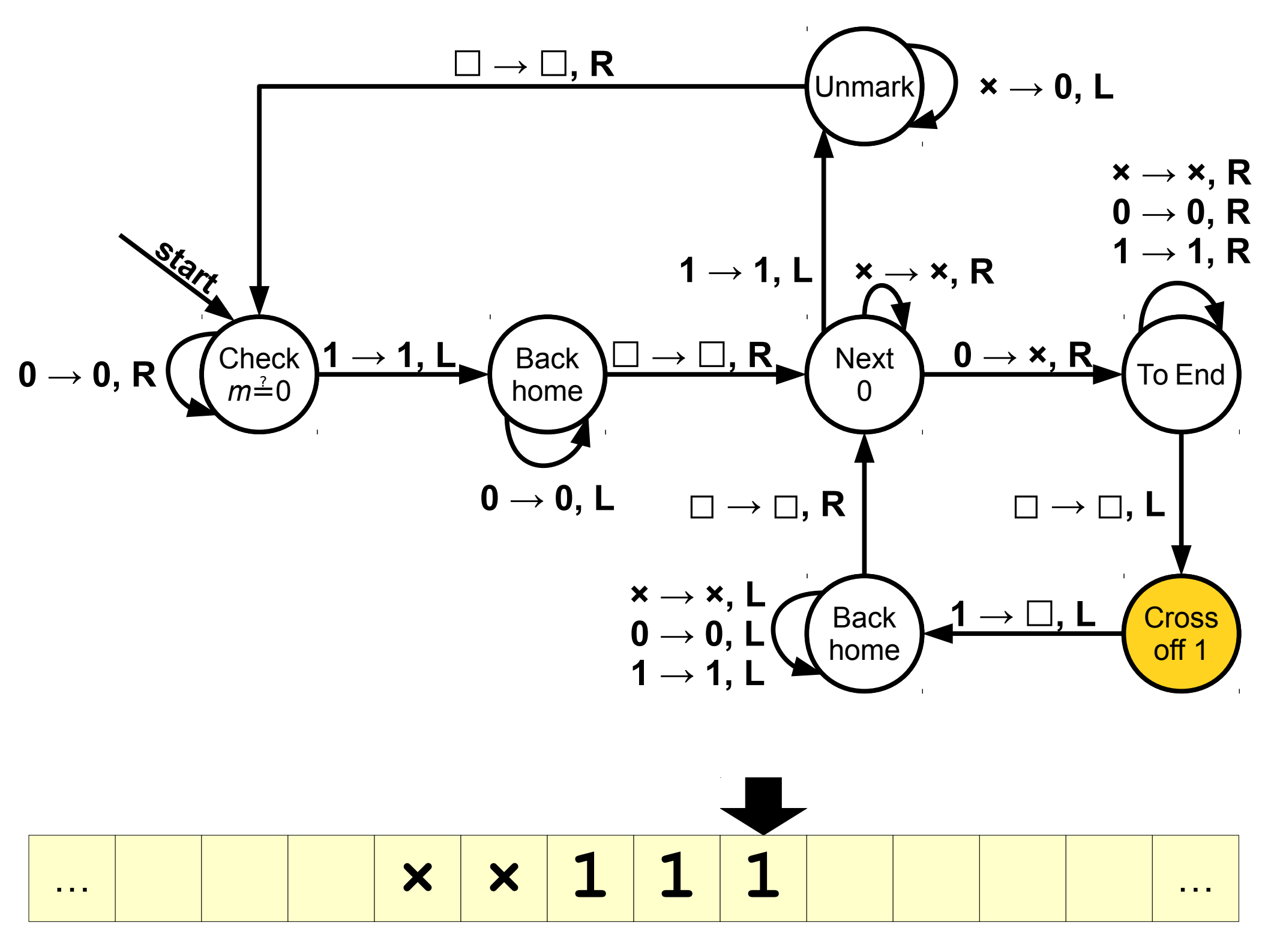

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 1 …

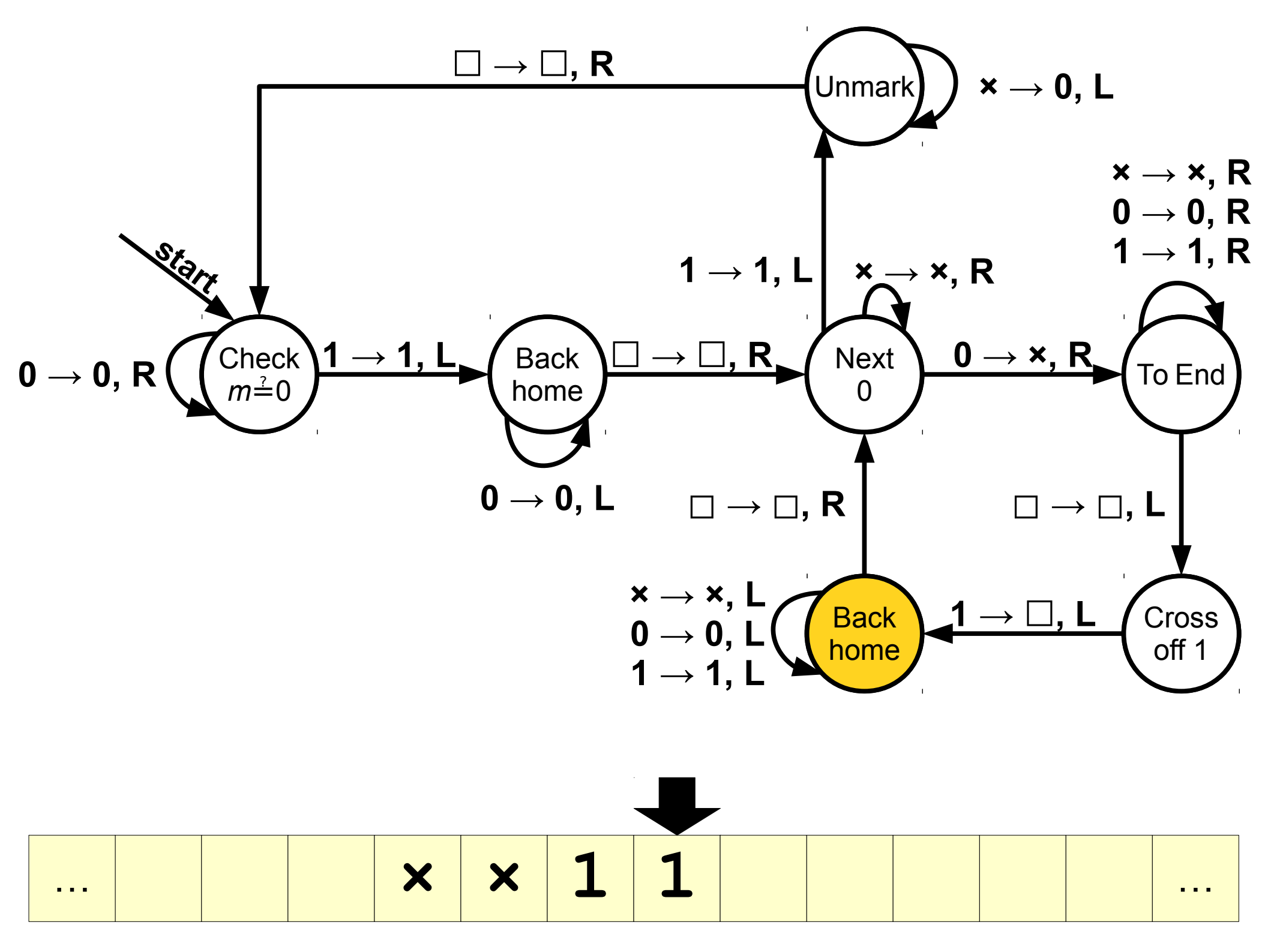

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 1 …

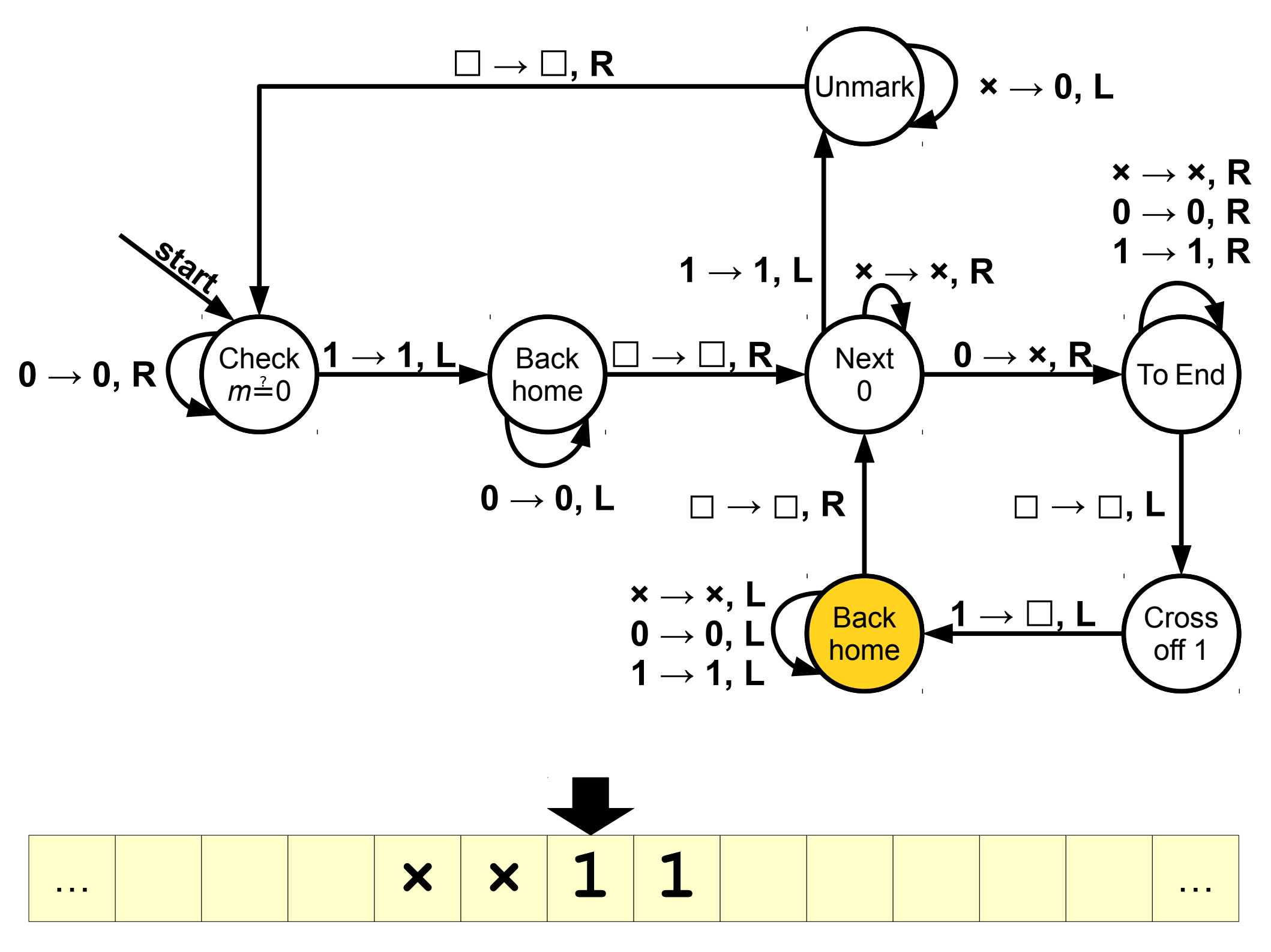

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 1 …

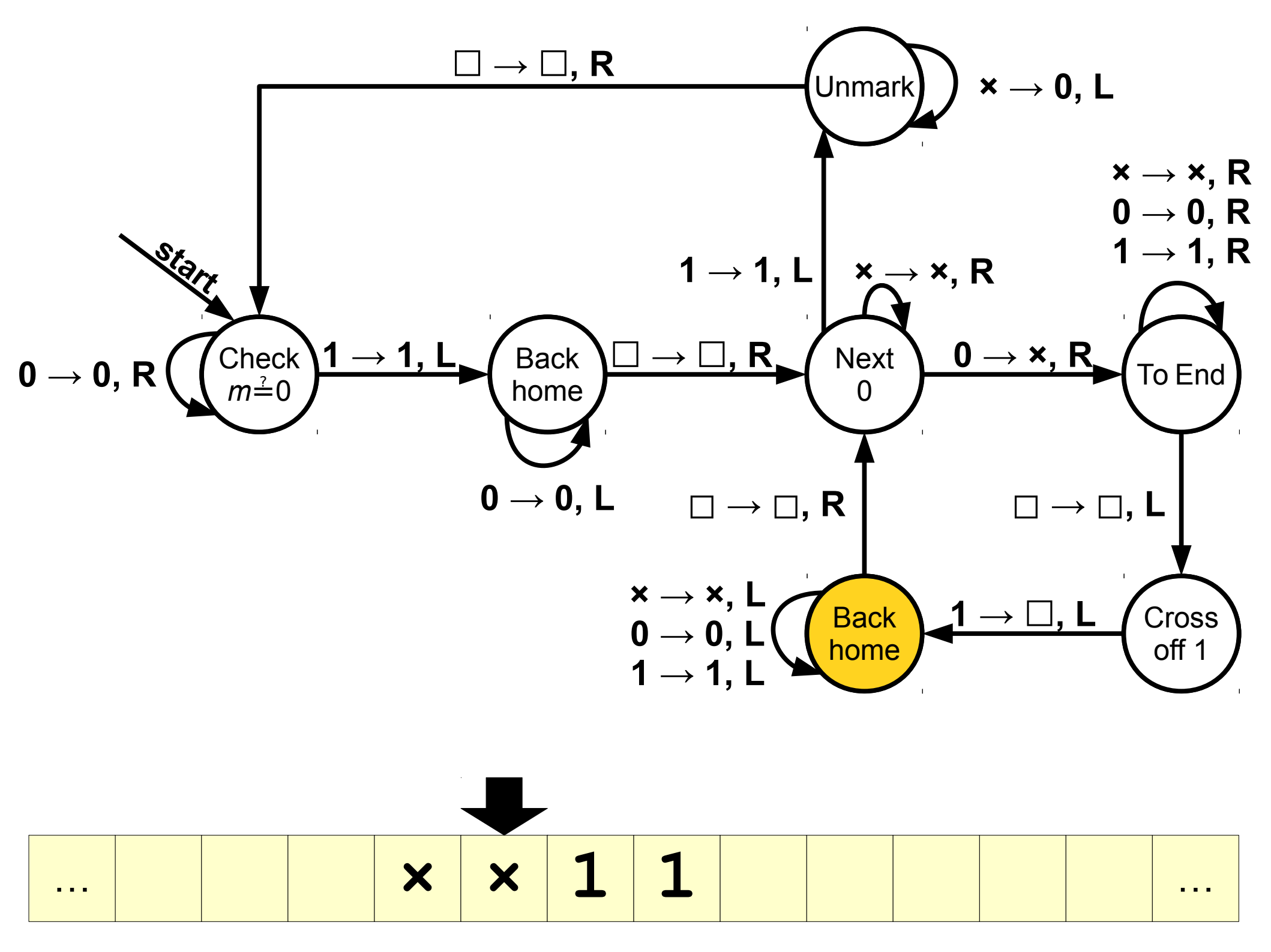

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 1 …

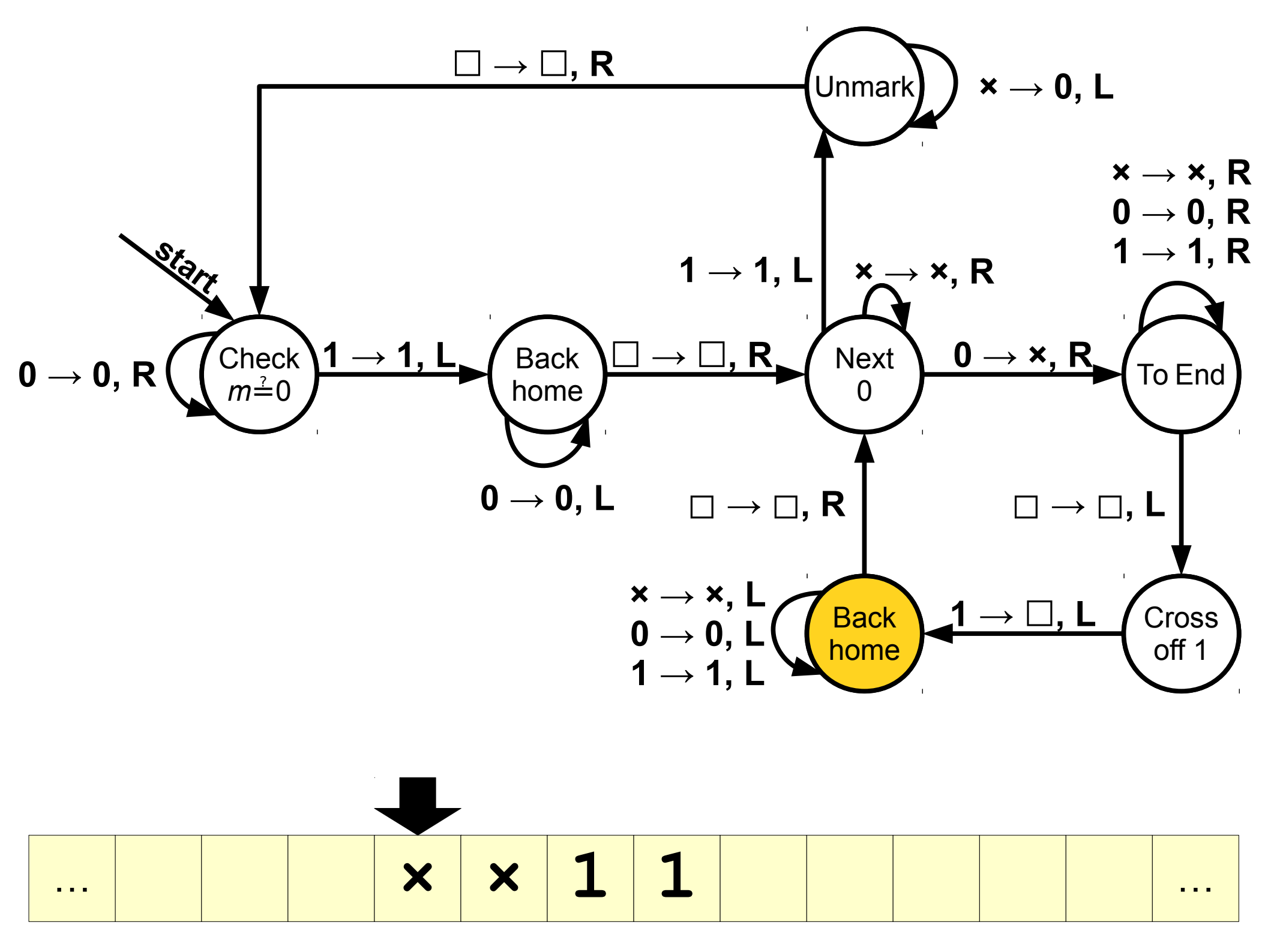

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 1 …

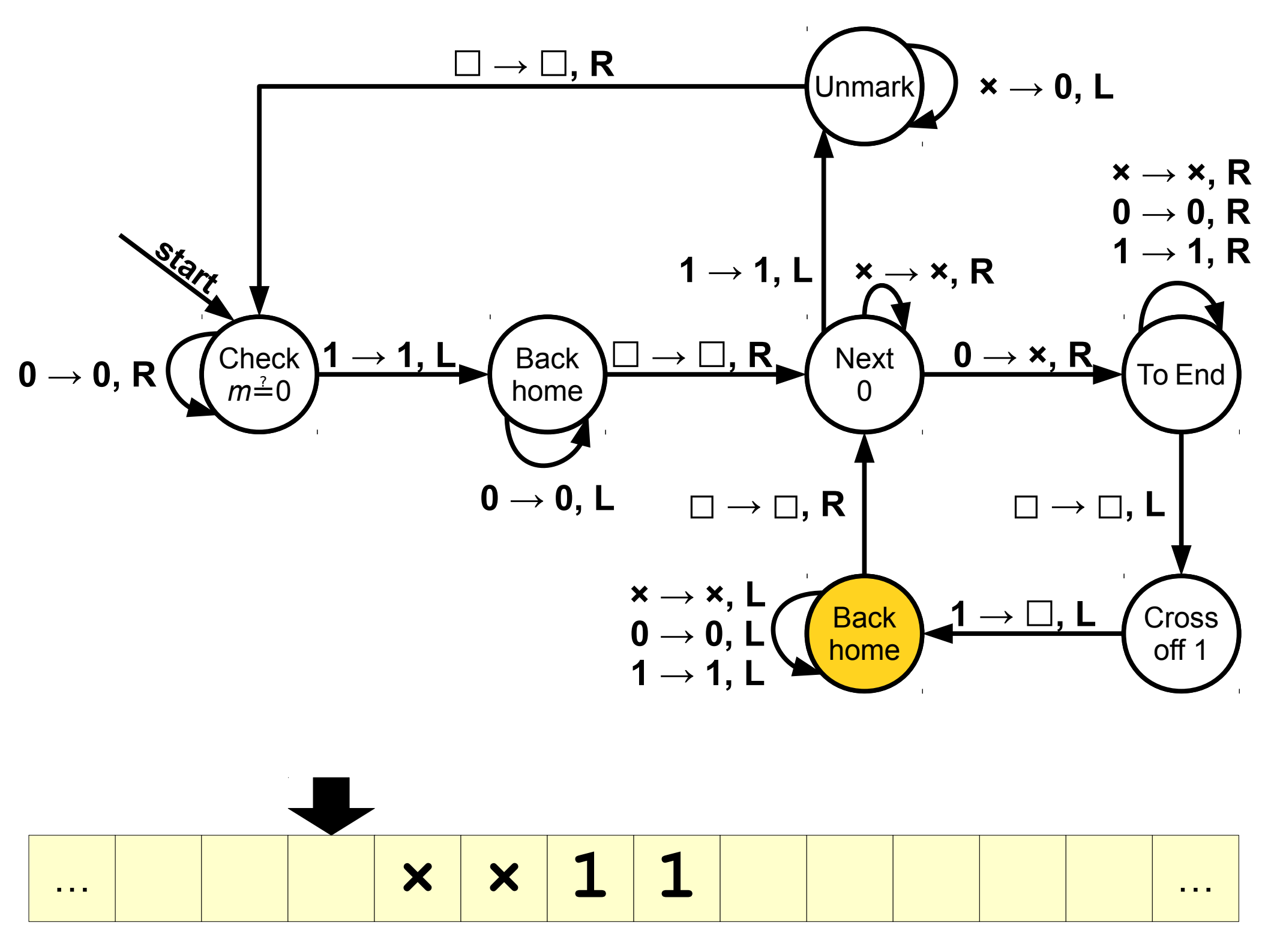

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 1 …

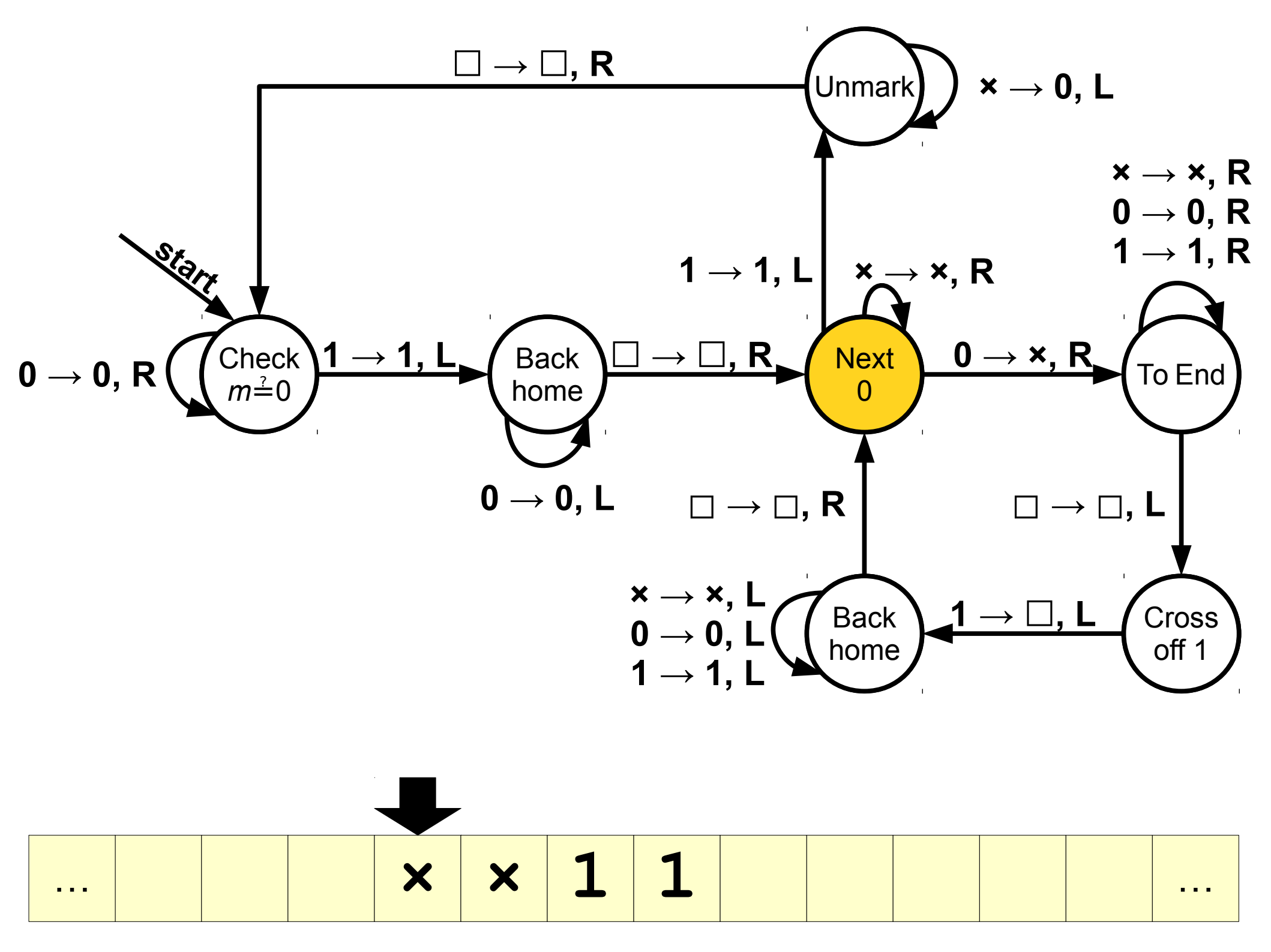

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
1
1
…

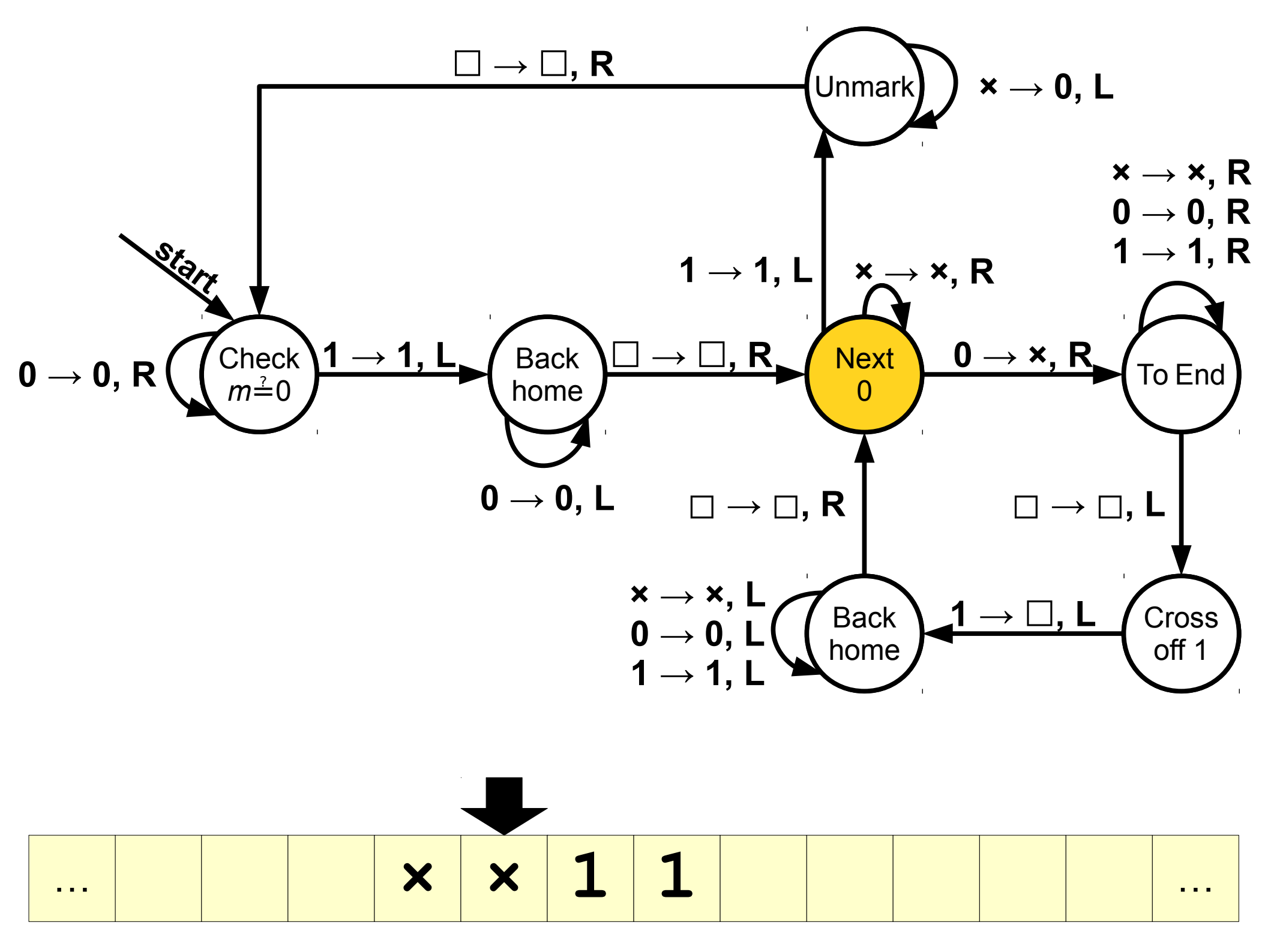

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
1
1
…

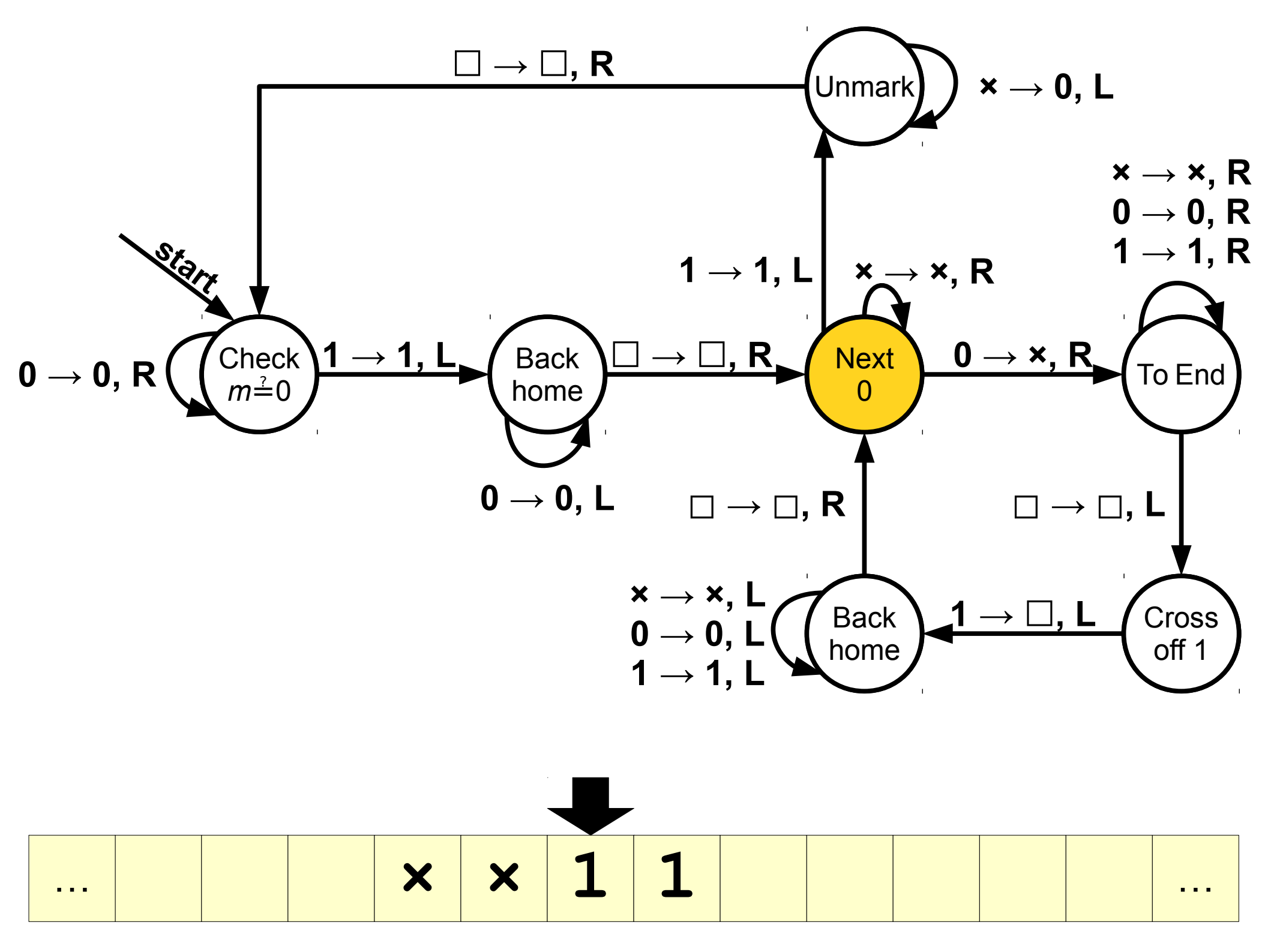

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 …

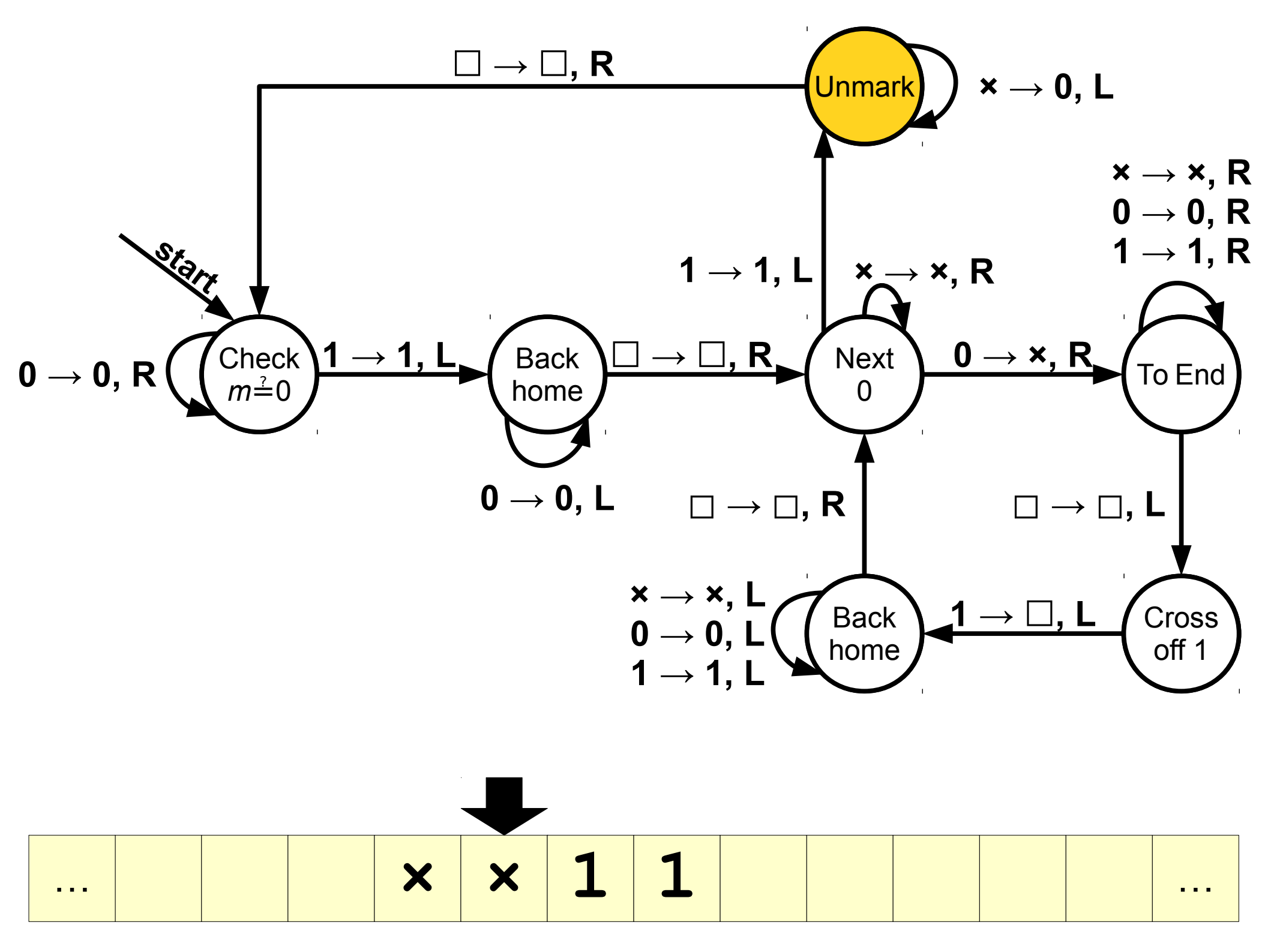

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
× × 1 1
…

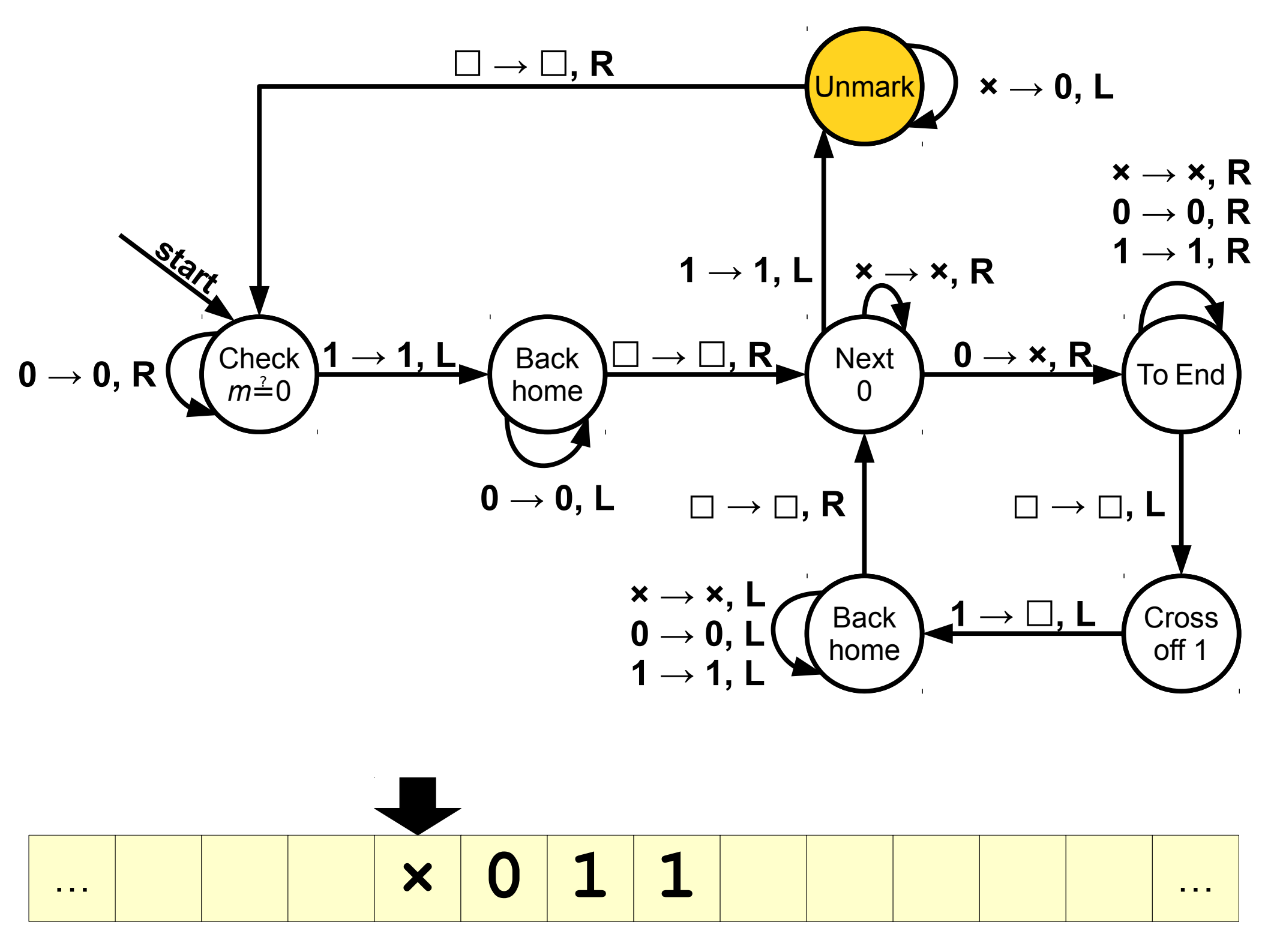

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 …

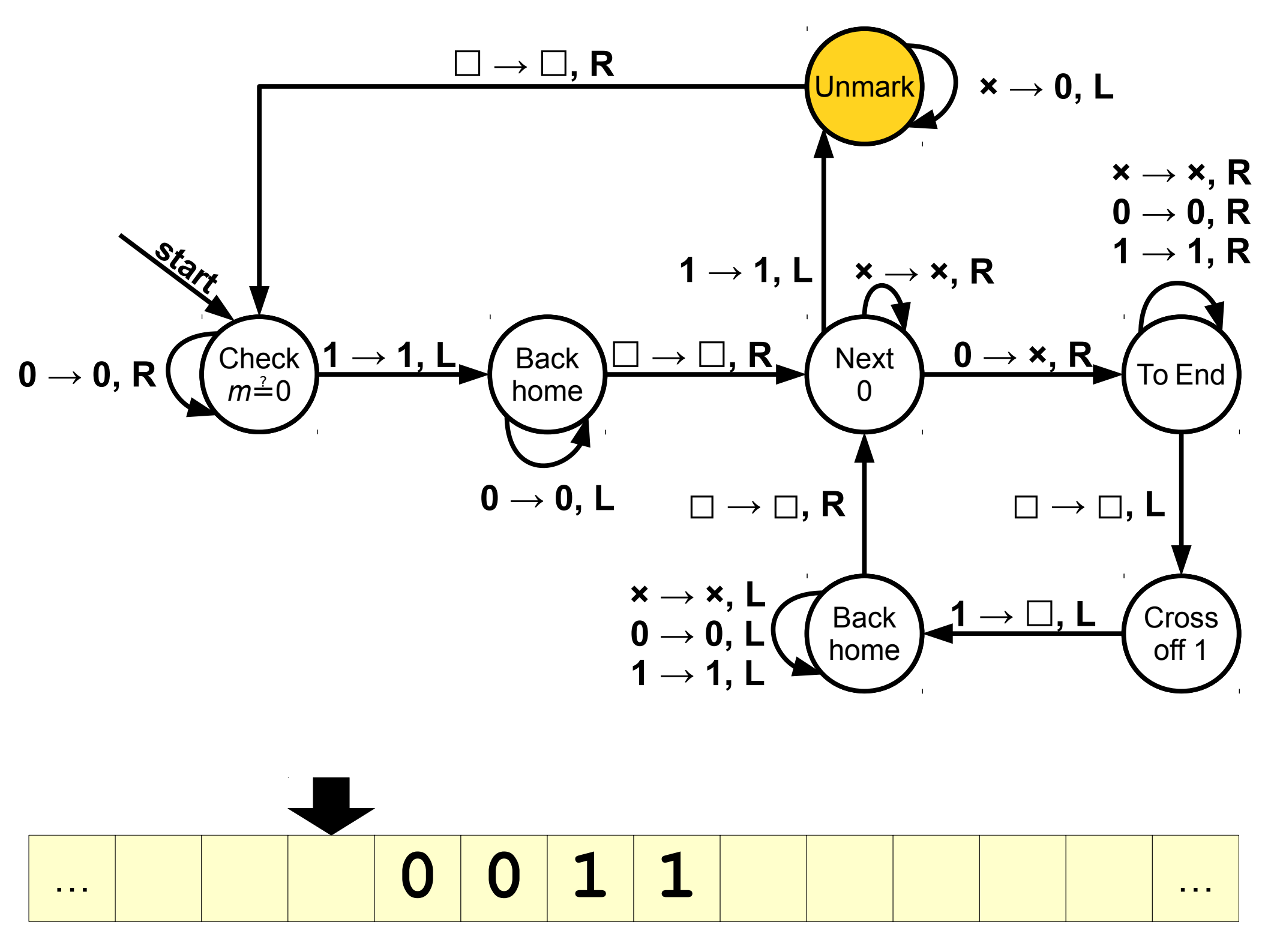

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
1
1
…

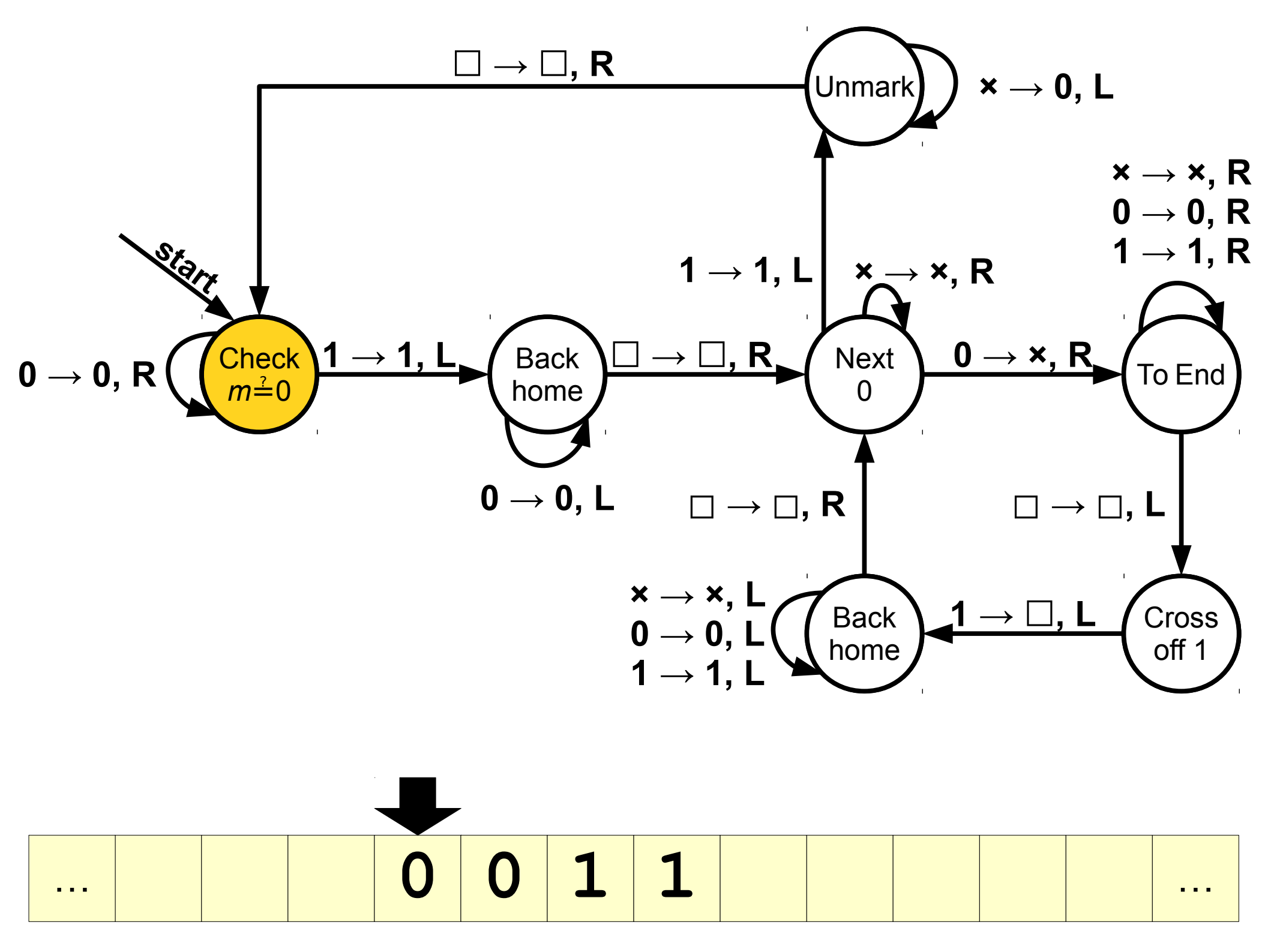

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Unmark
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
1 → 1, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
… × × 1 1 …

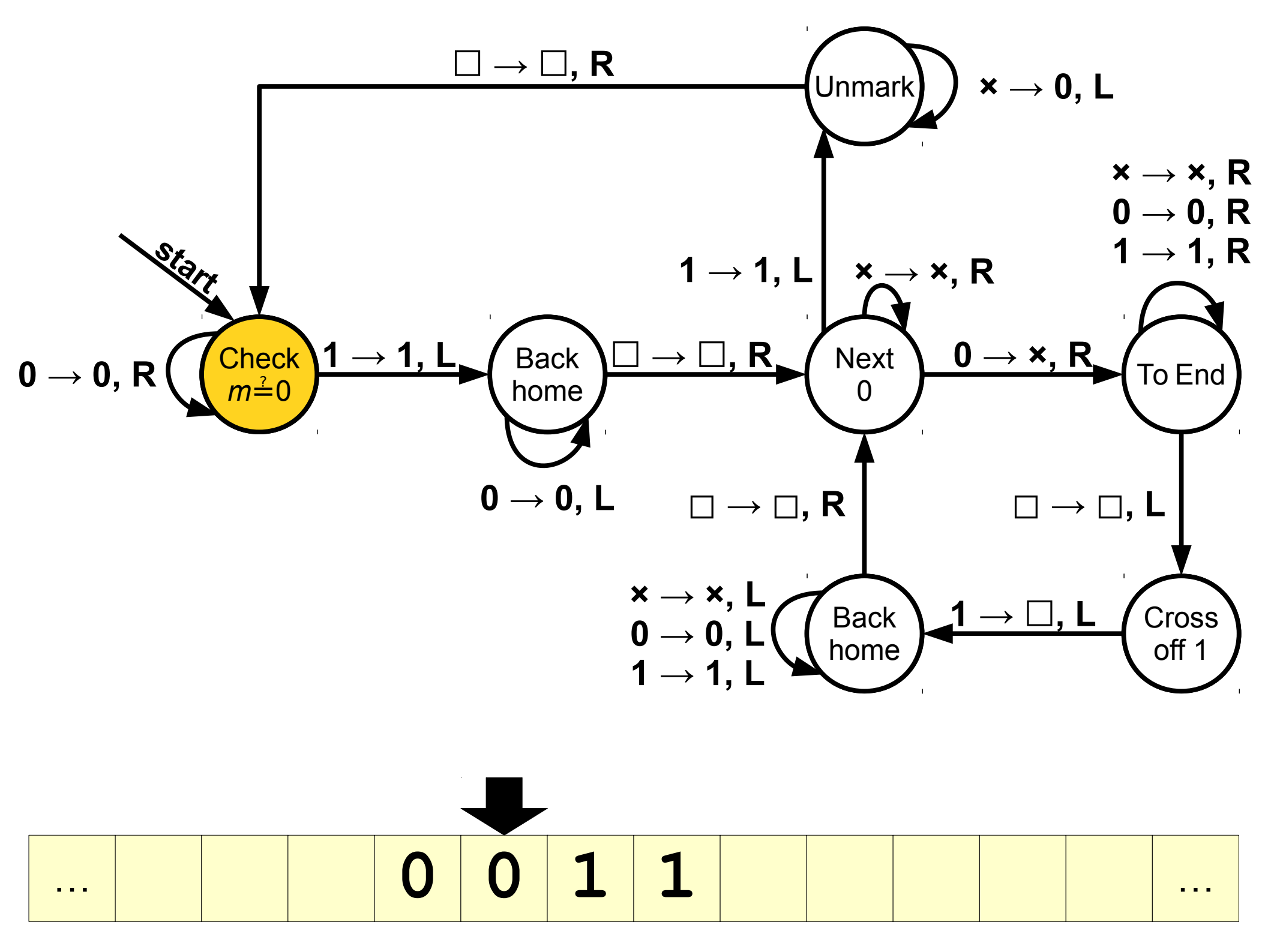

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 …

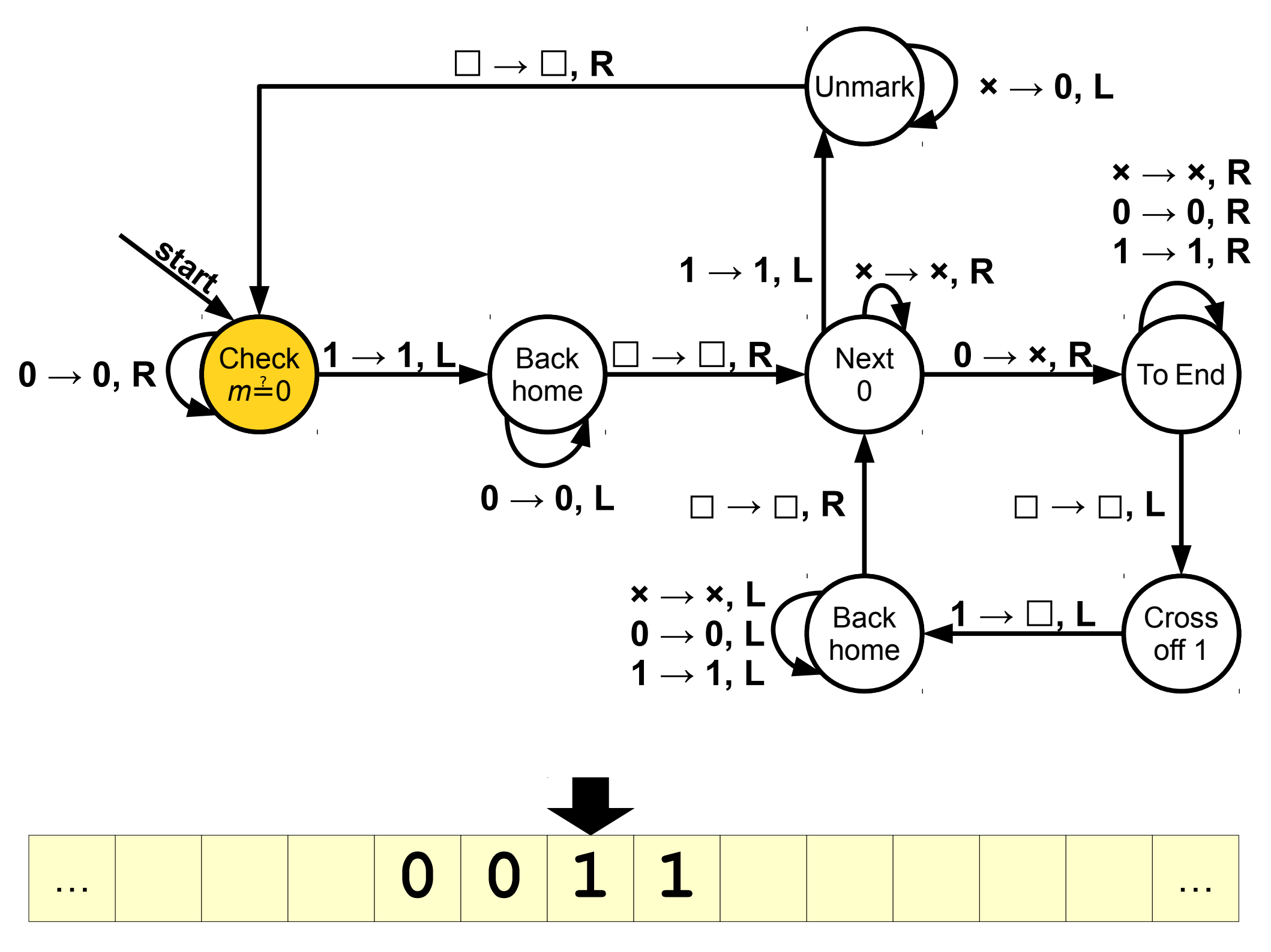

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 1 …

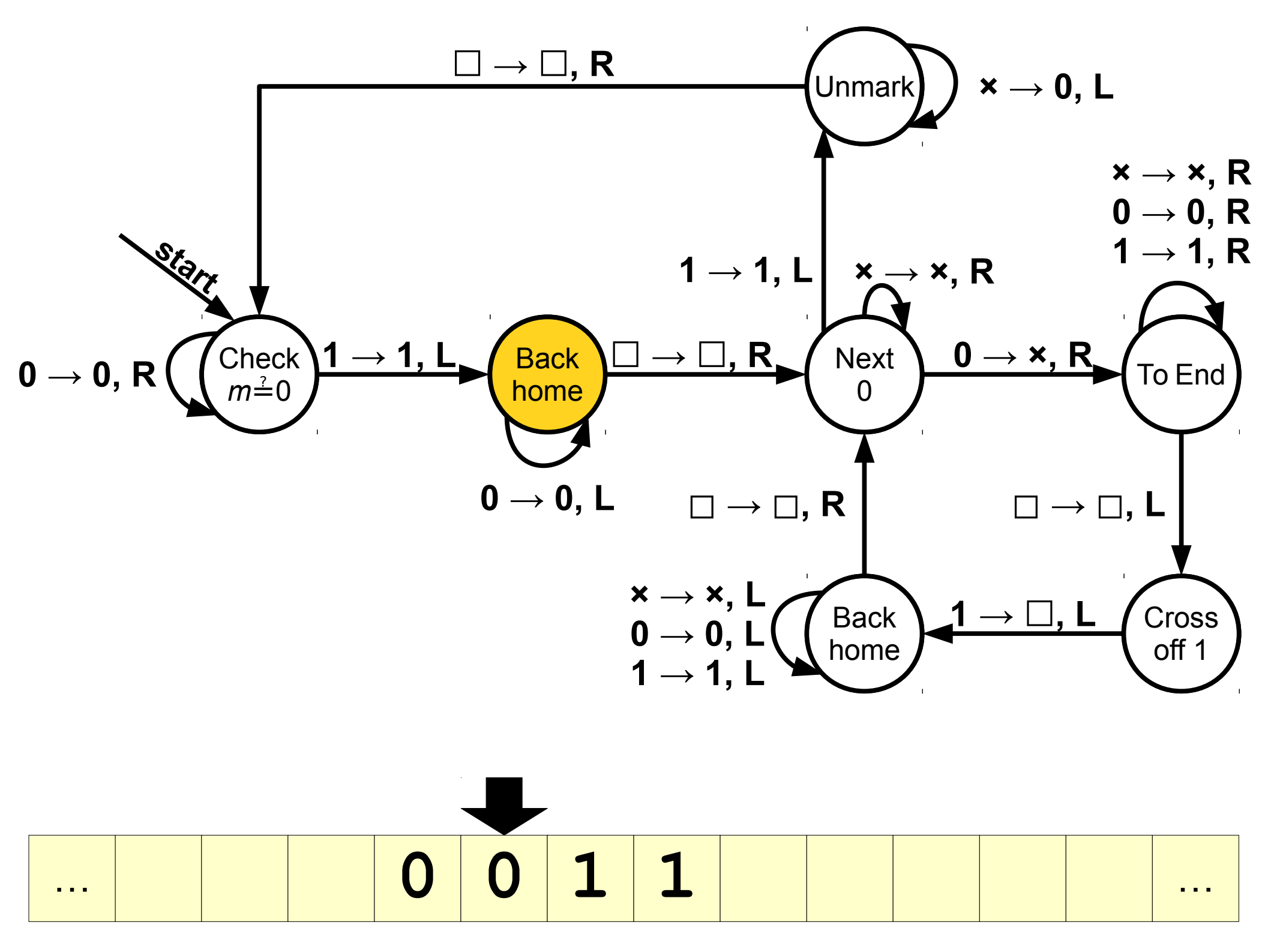

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
…

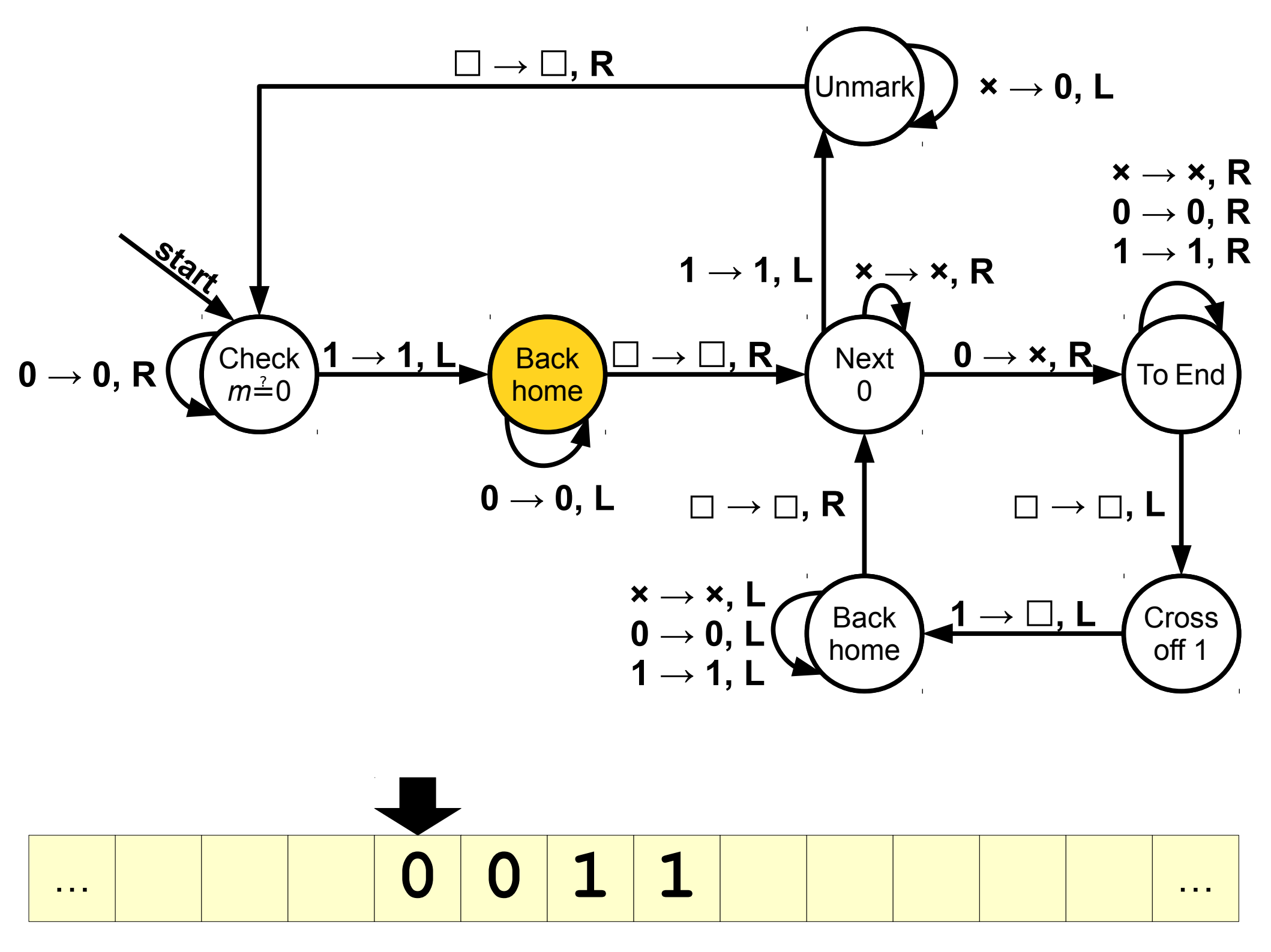

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
…

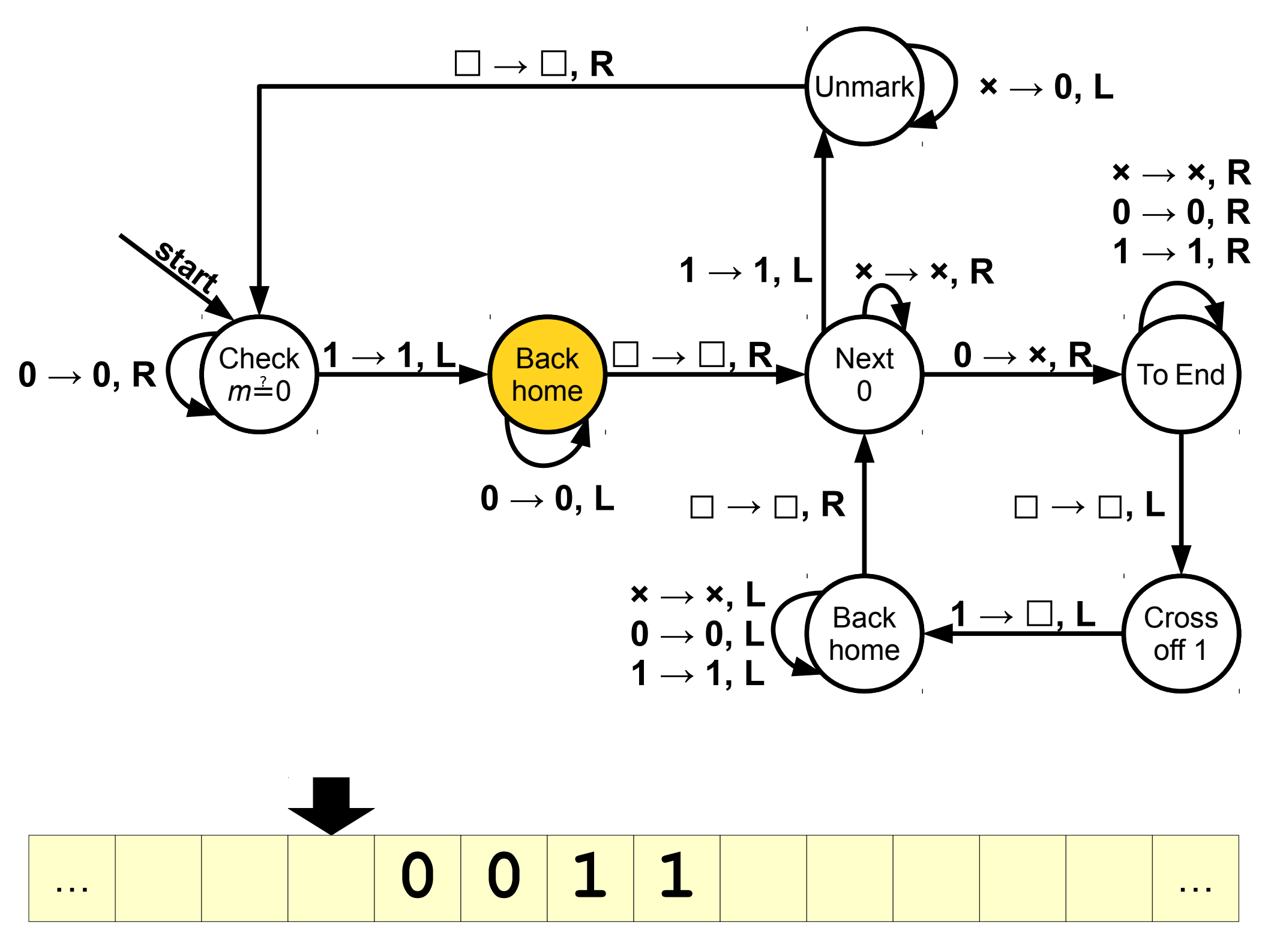

start
Check
m=?0
Back
home
Next
0
Unmark
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
1 → 1, L
□ → □, R
□ → □, L
1 → □, L
× → ×, L
0 → 0, L
1 → 1, L
… 0 0 1 1 …

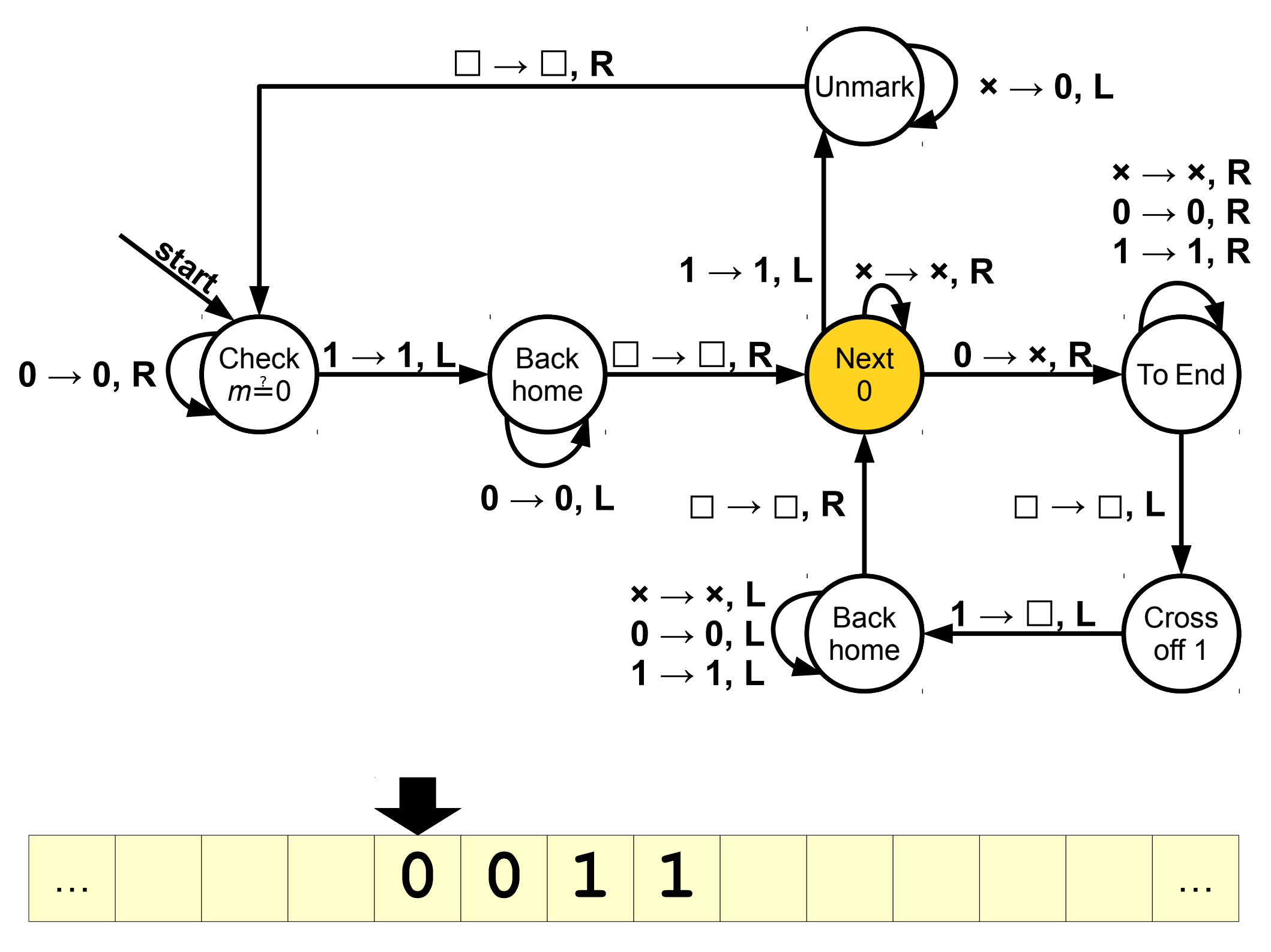

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
…

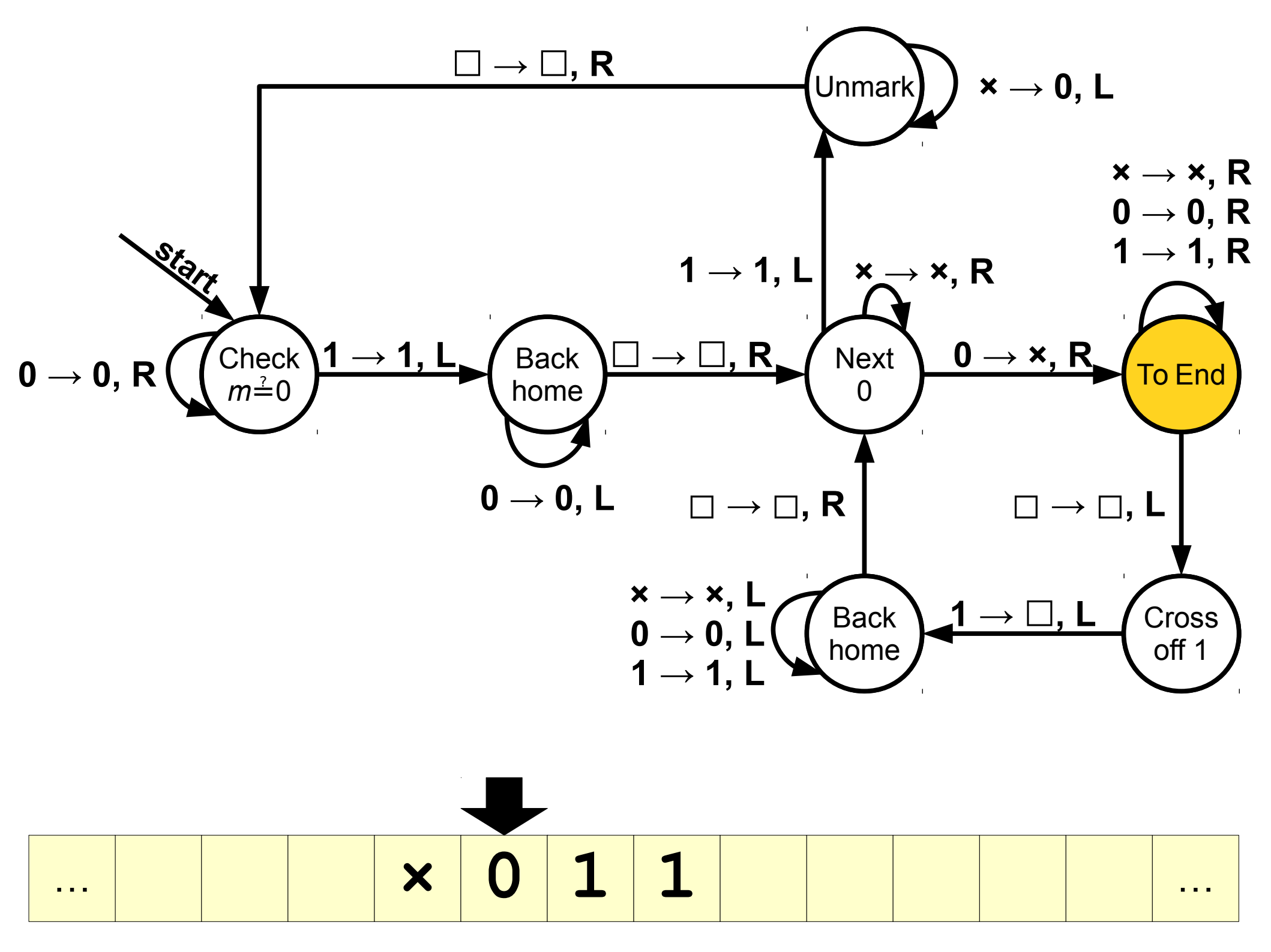

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
…

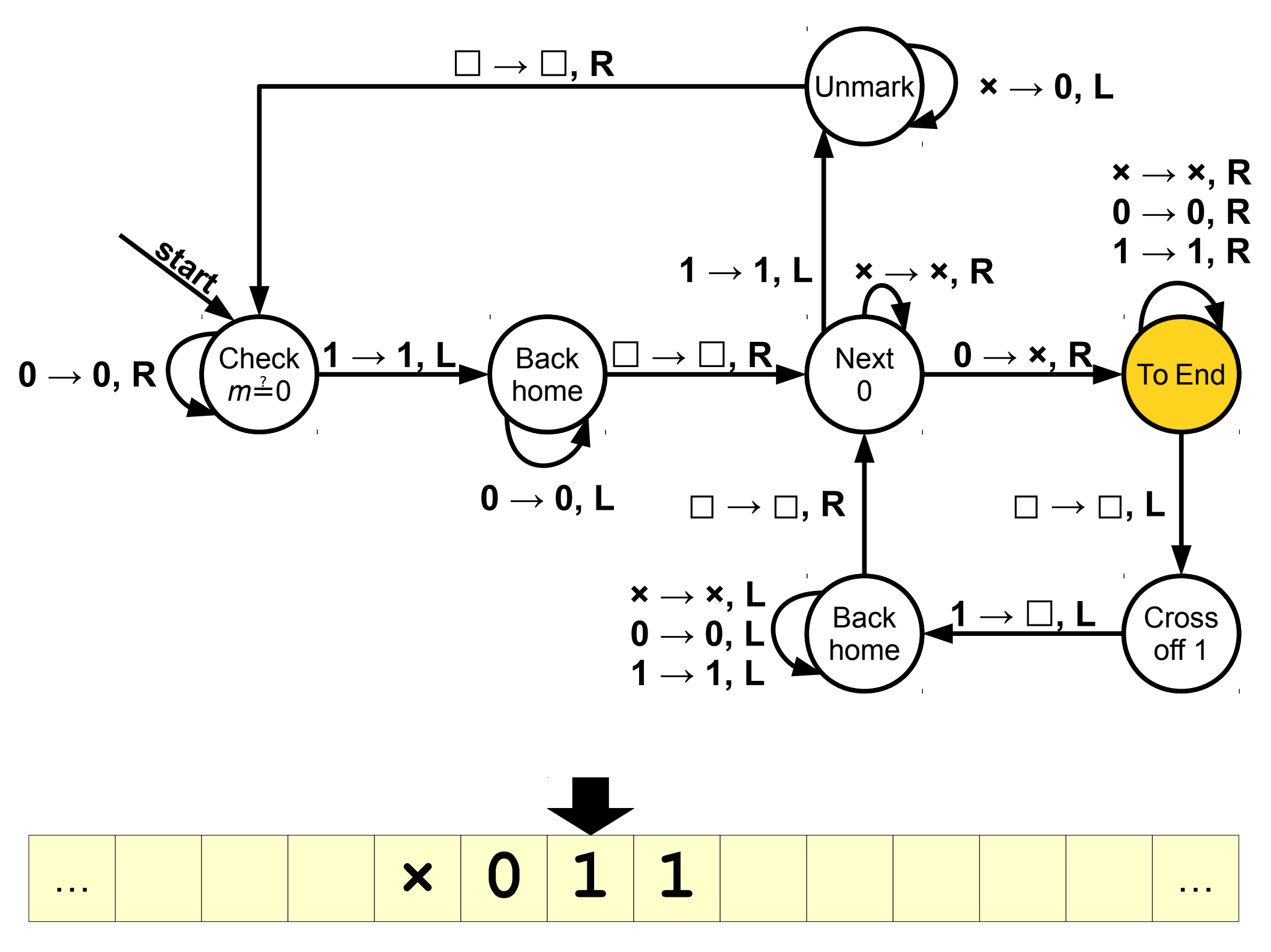

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
…

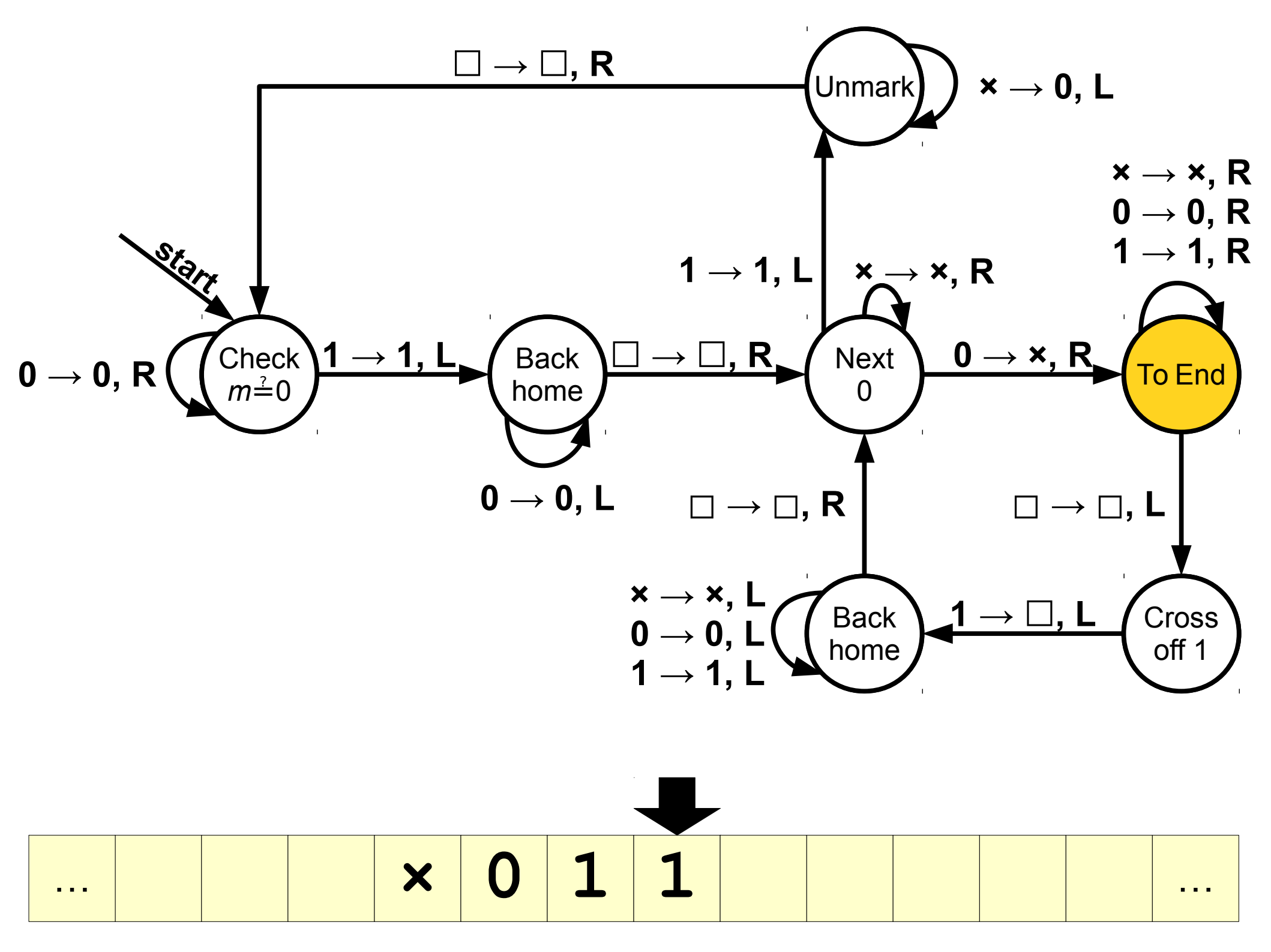

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
…

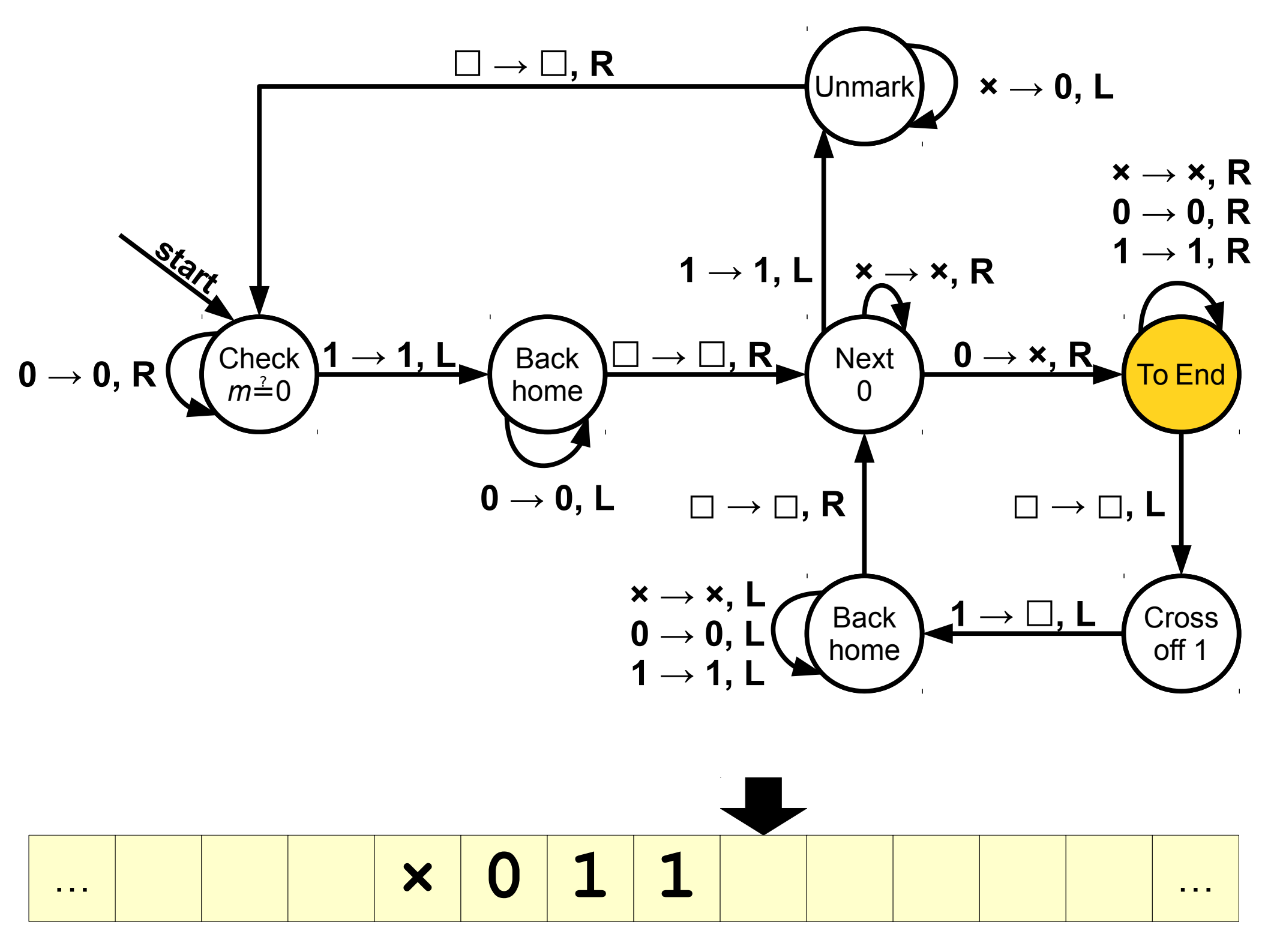

start
Check
$m \stackrel{?}{=} 0$
Back
home
Unmark
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
1 → 1, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
…
0 0 1 1
…

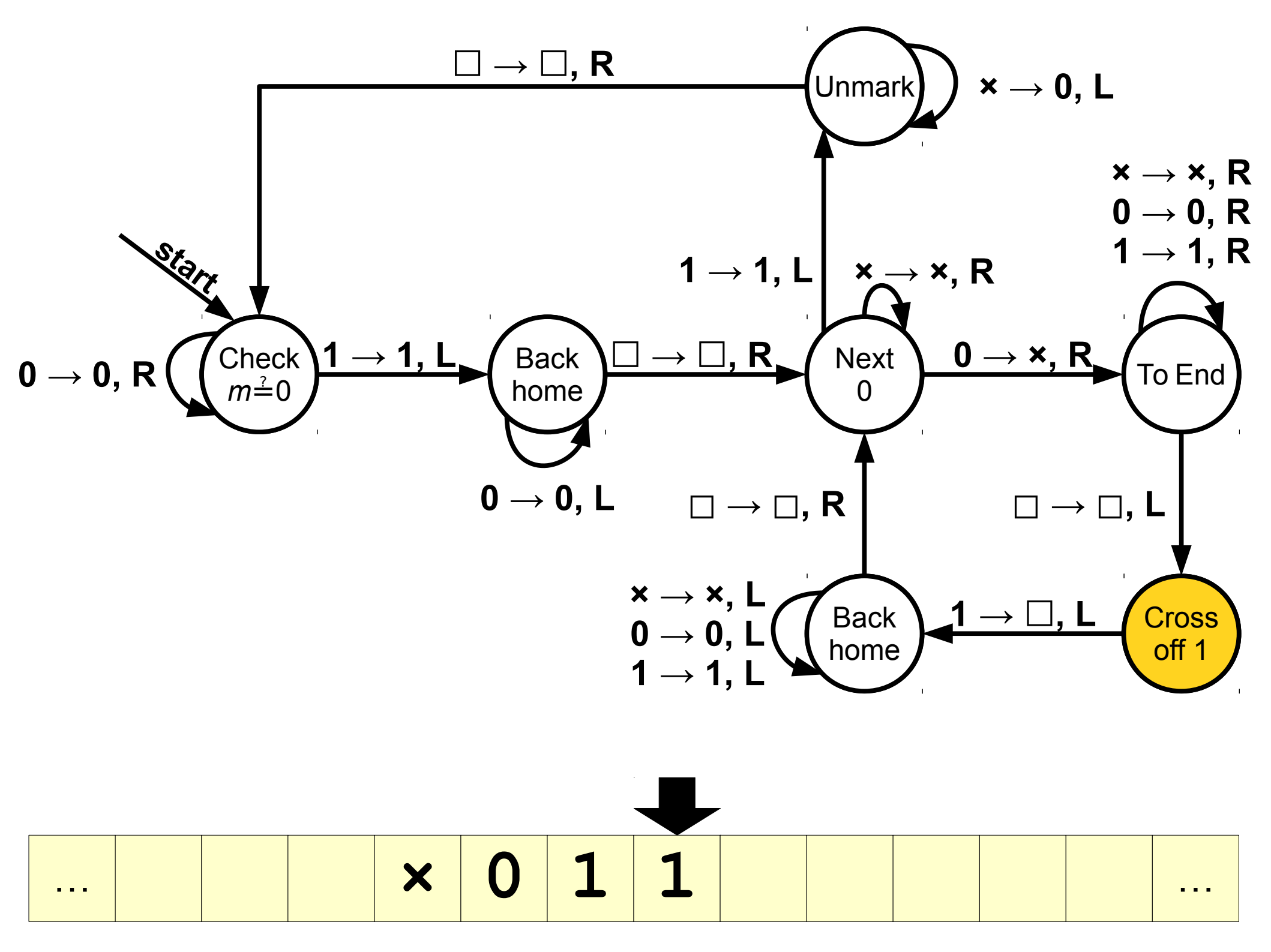

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m=0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
1
…

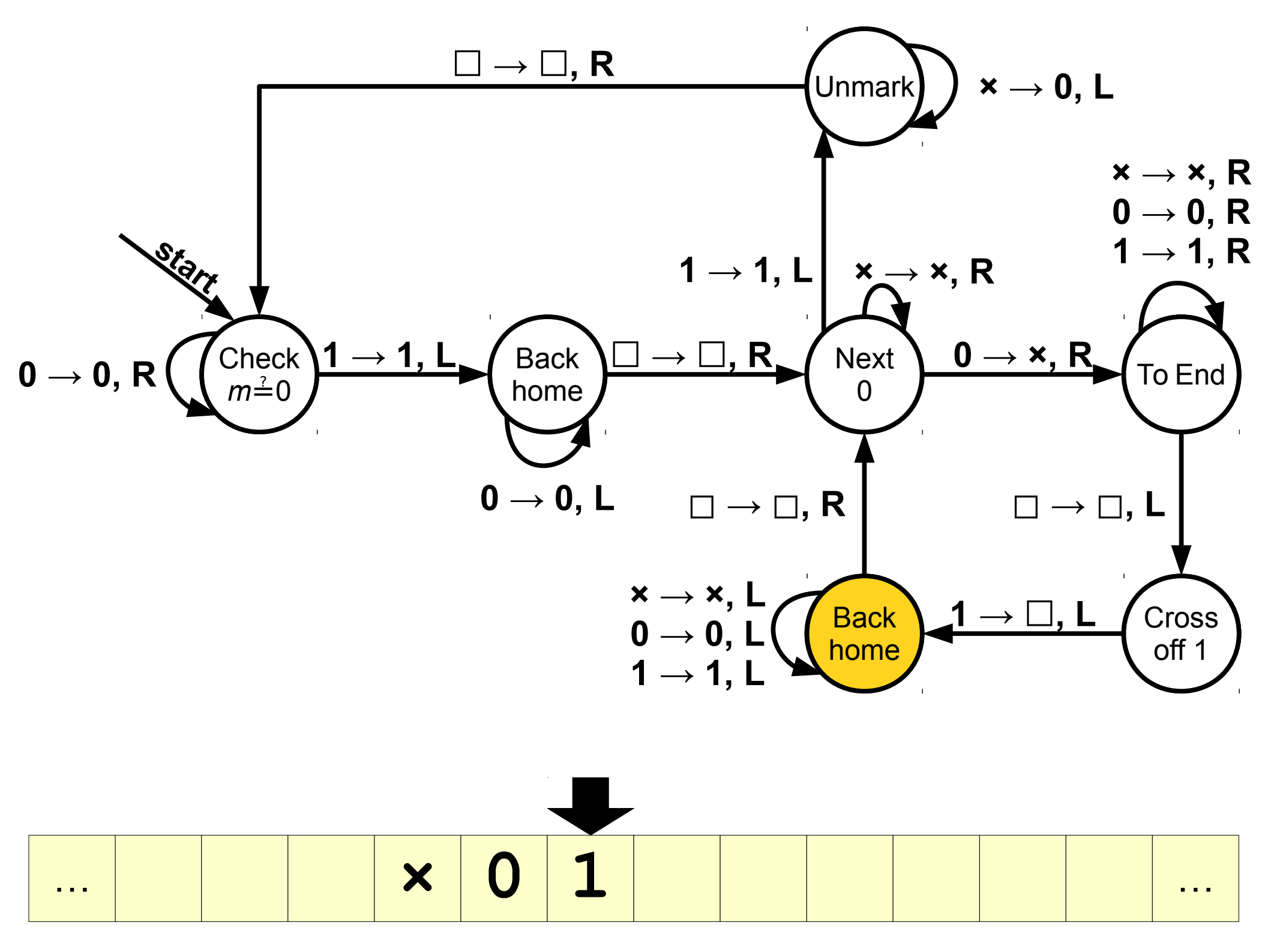

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
…

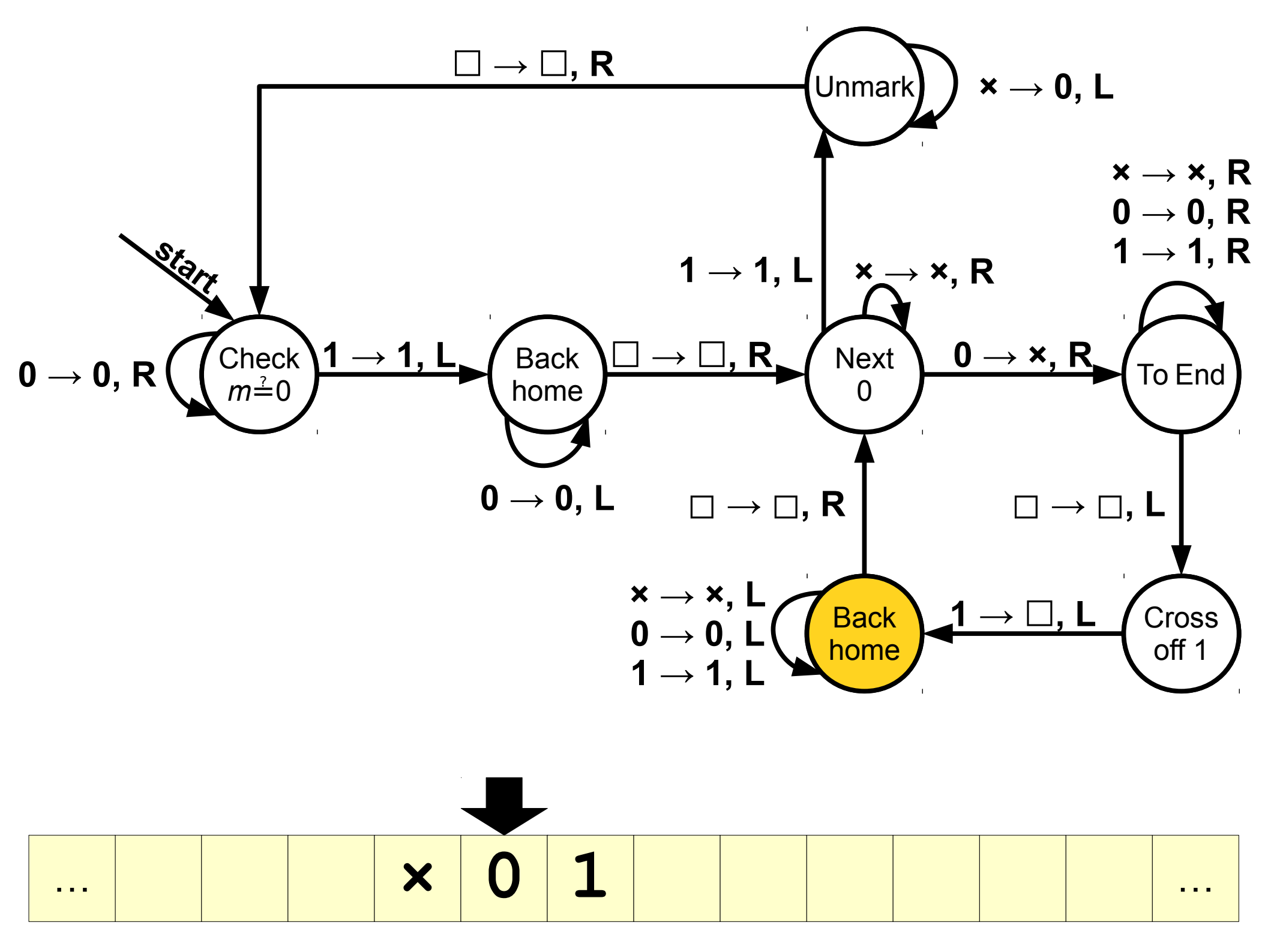

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 …

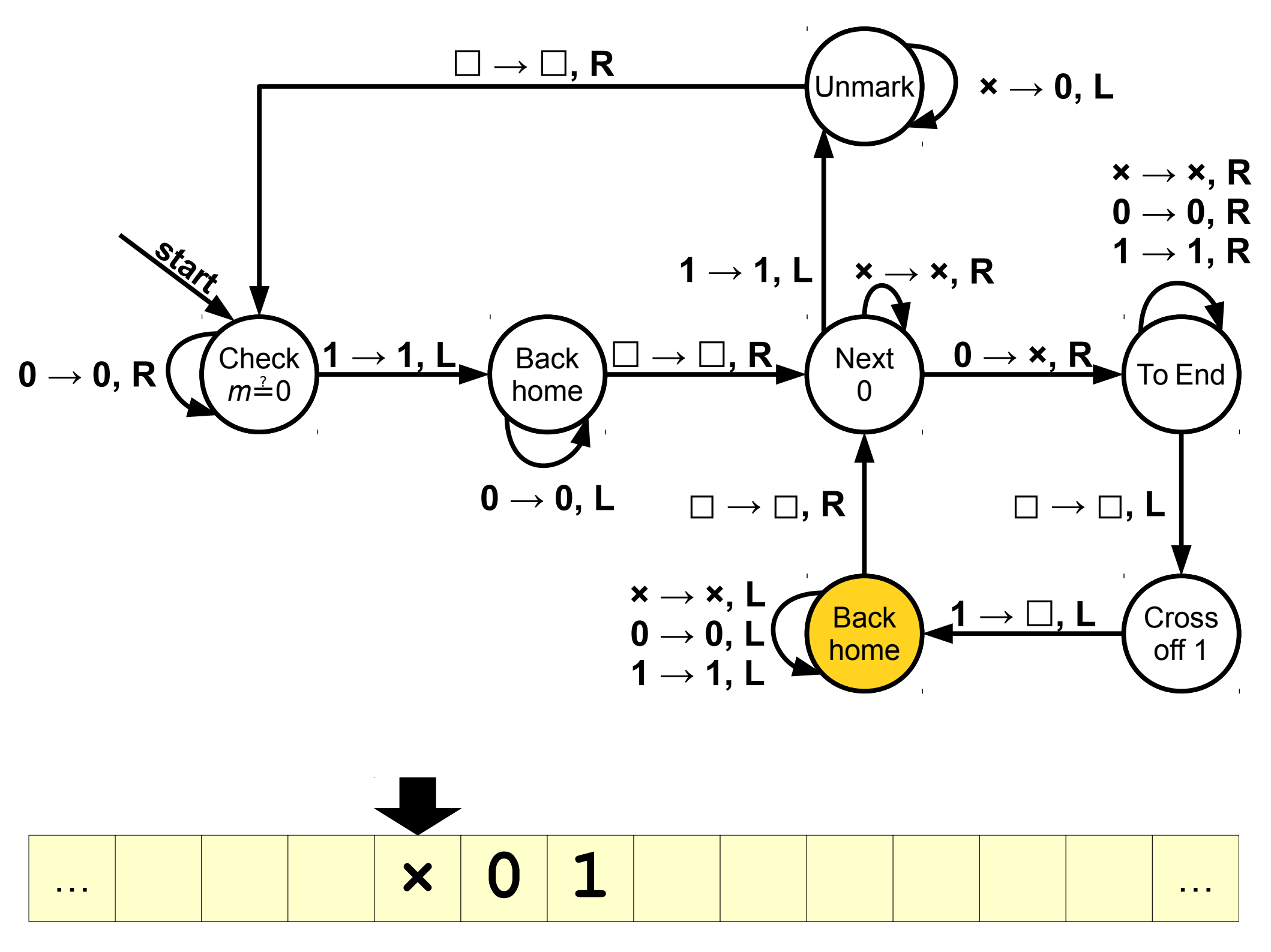

start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
Unmark
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
1 → 1, L
× → ×, R
0 → ×, R
× → ×, R
0 → 0, R
1 → 1, R
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
…
×
0
1
1
…

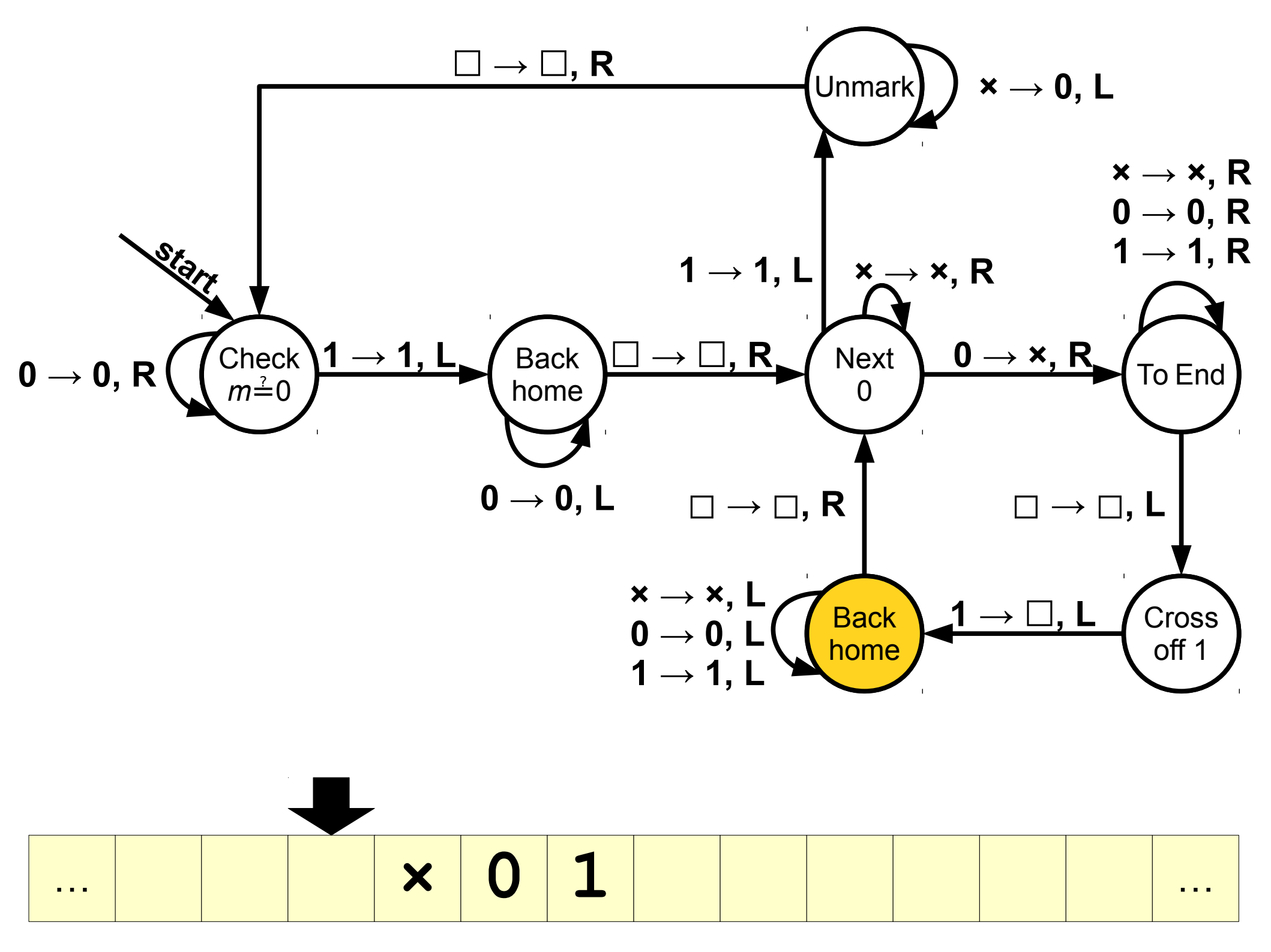

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 1 …

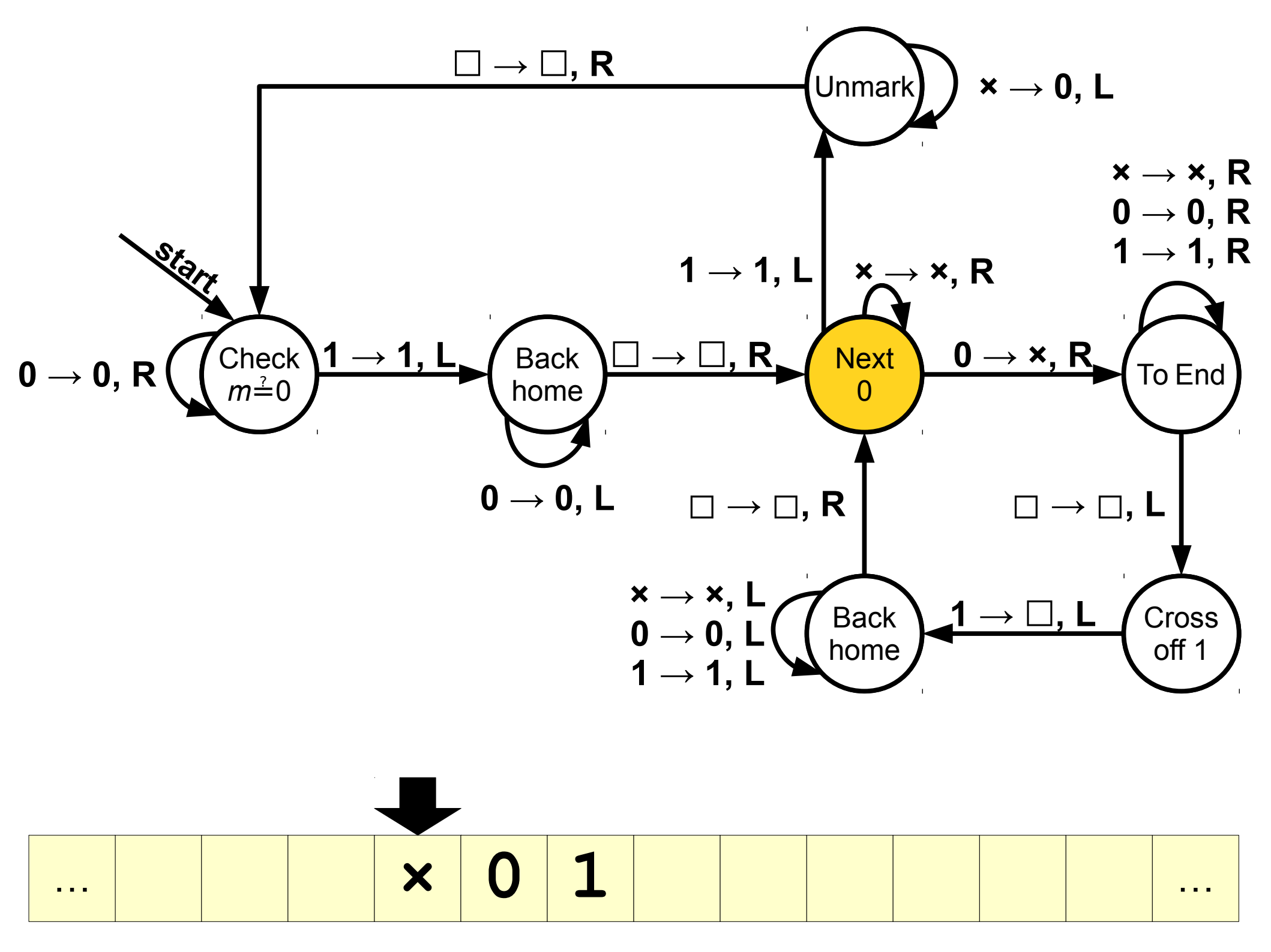

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
1
…

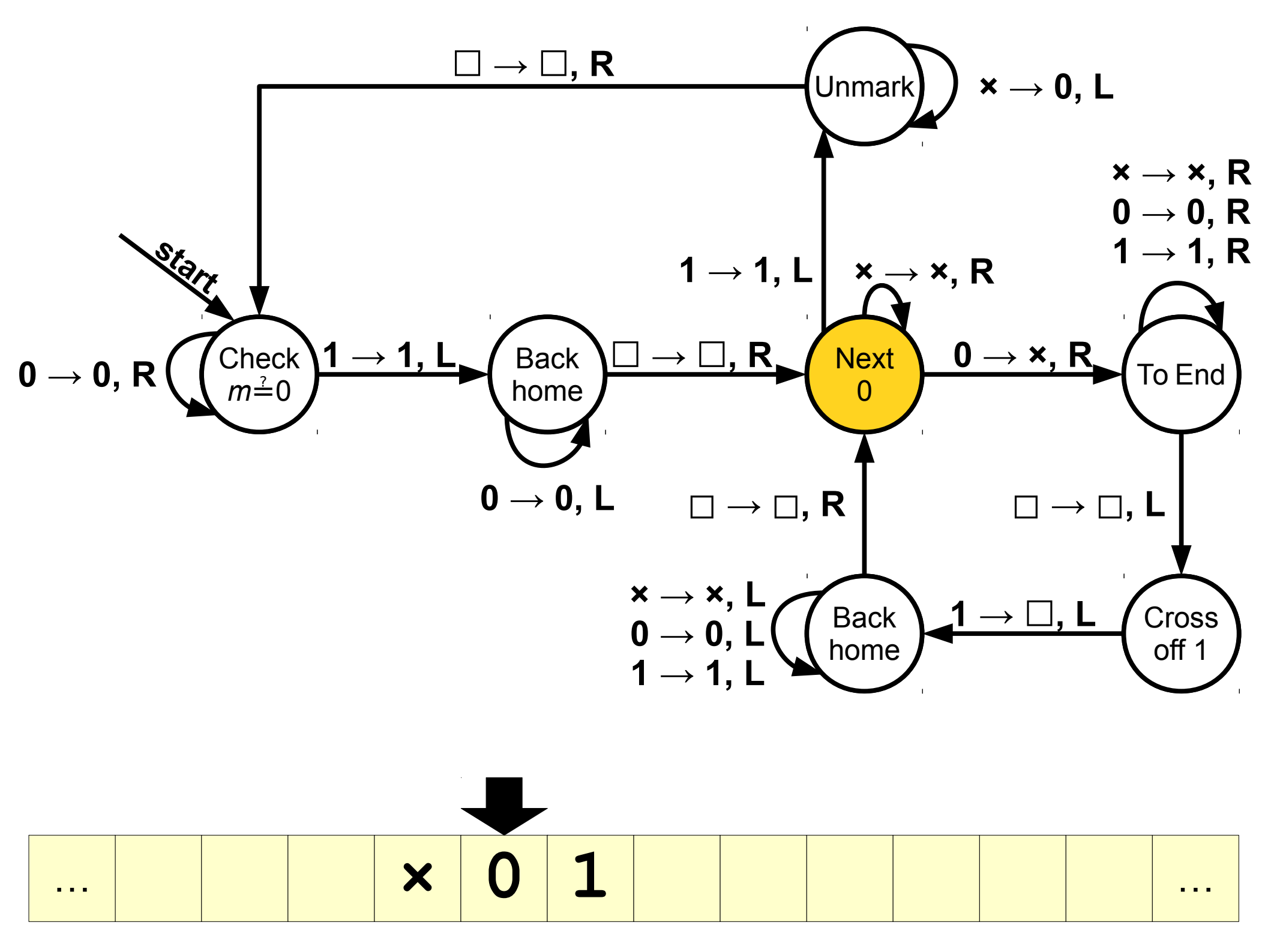

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

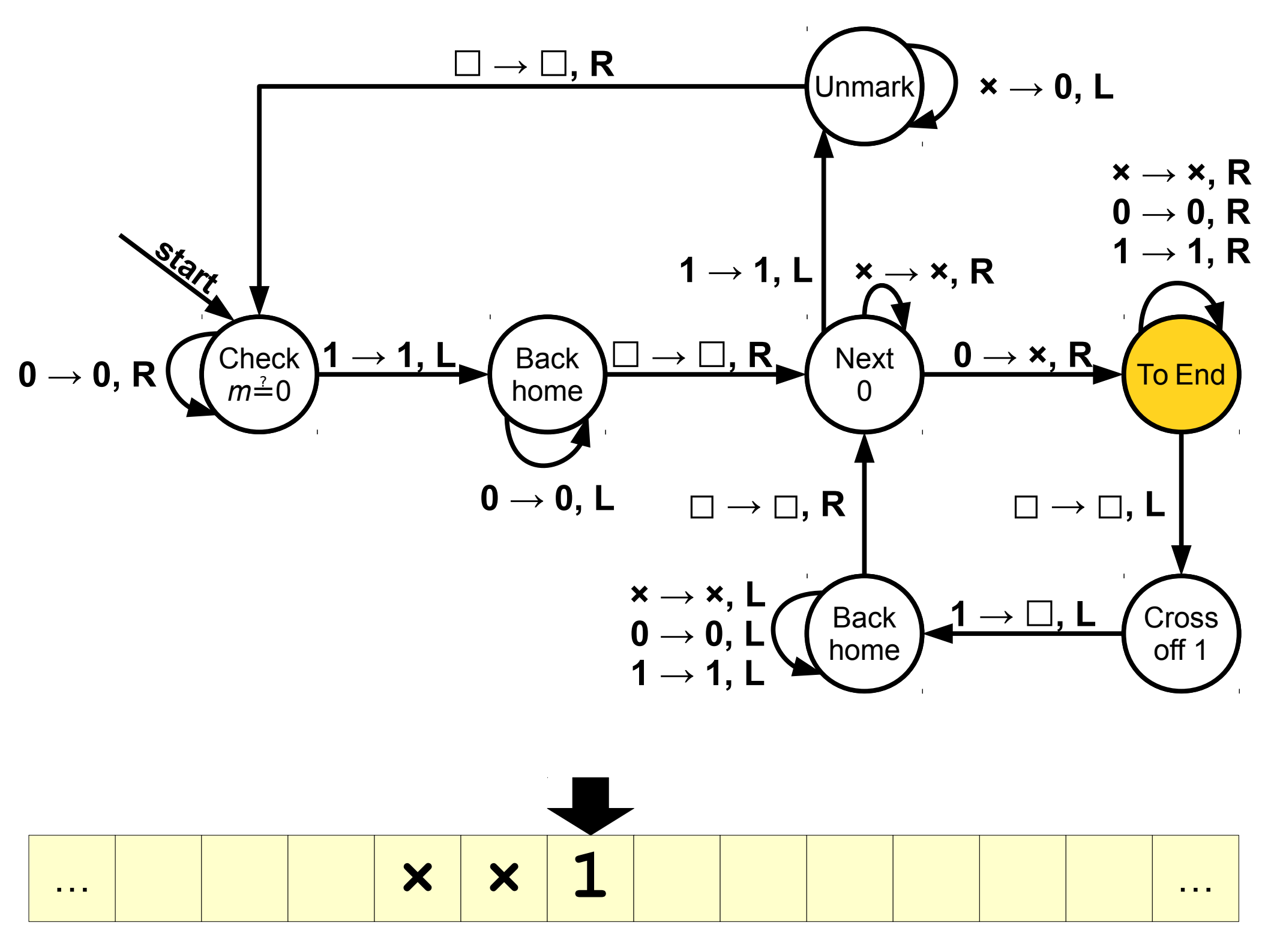

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

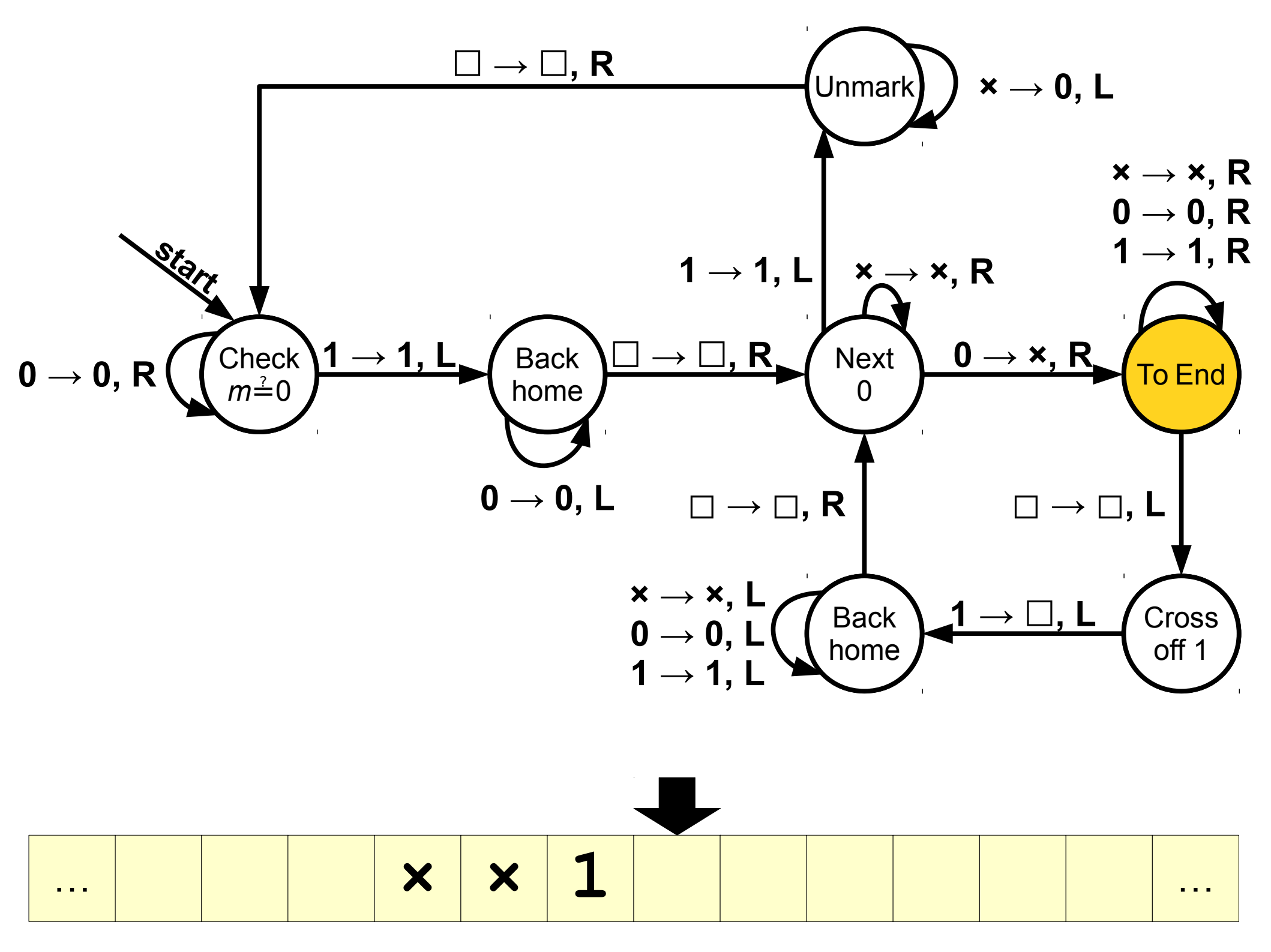

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

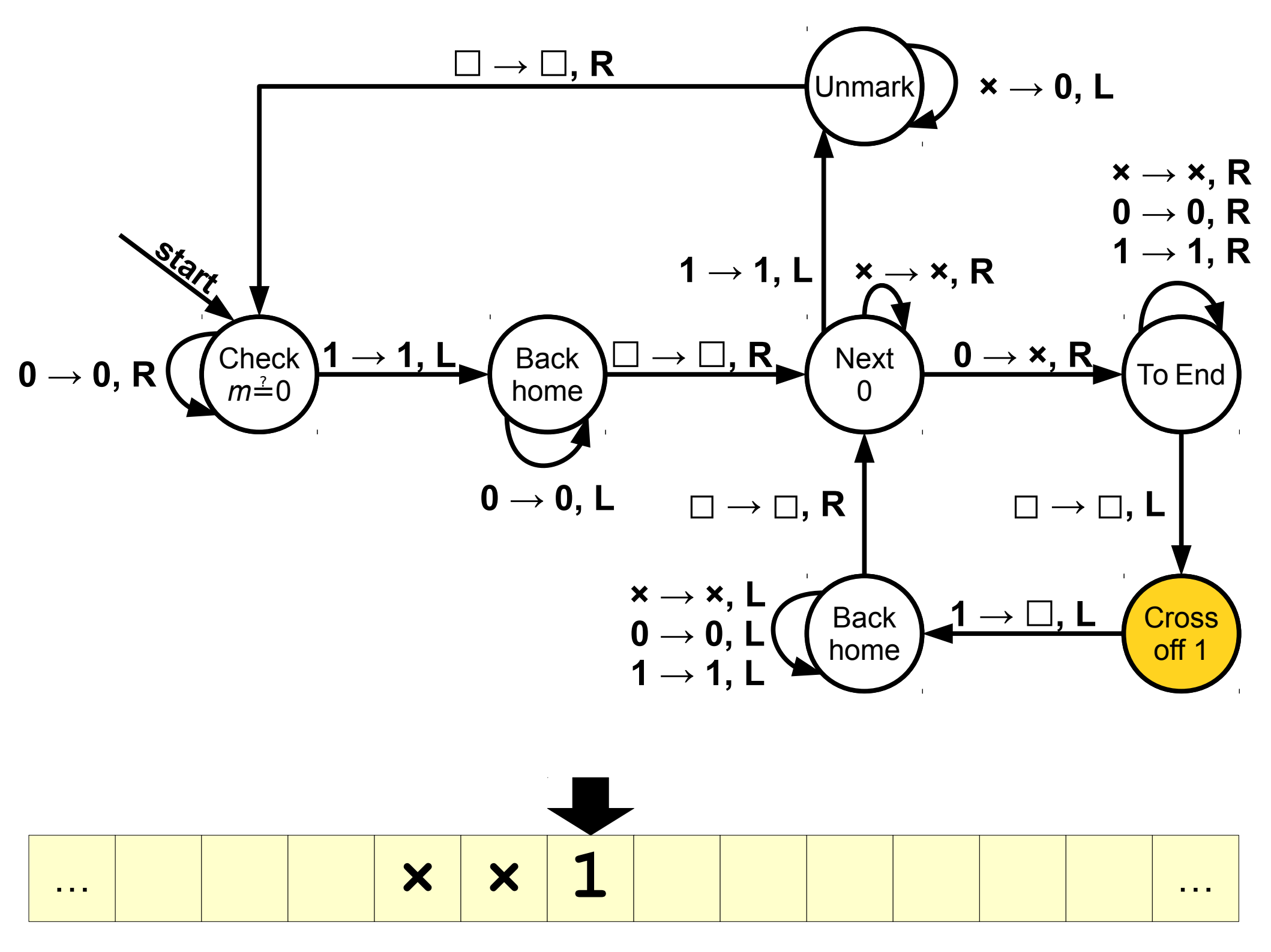

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

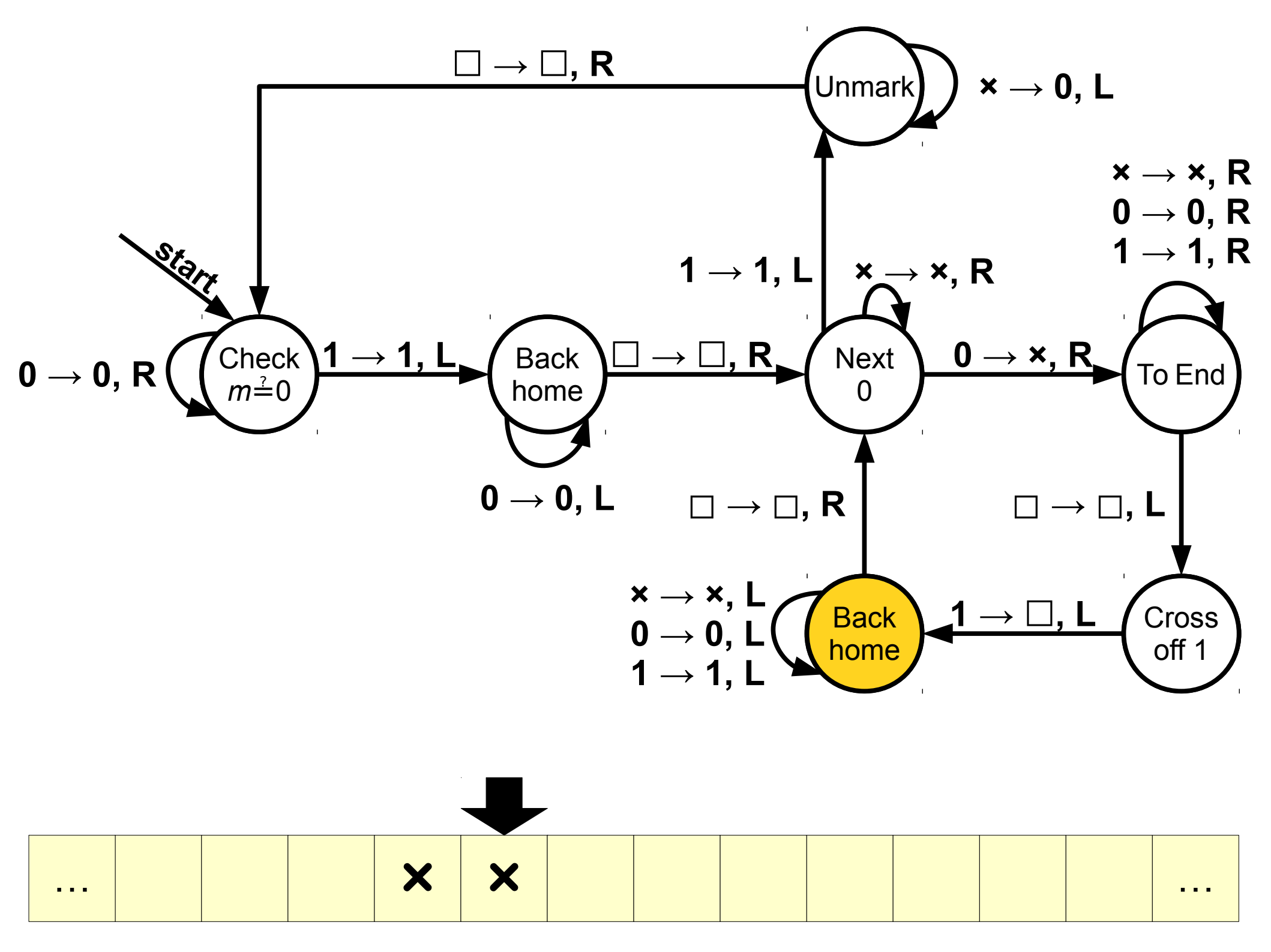

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

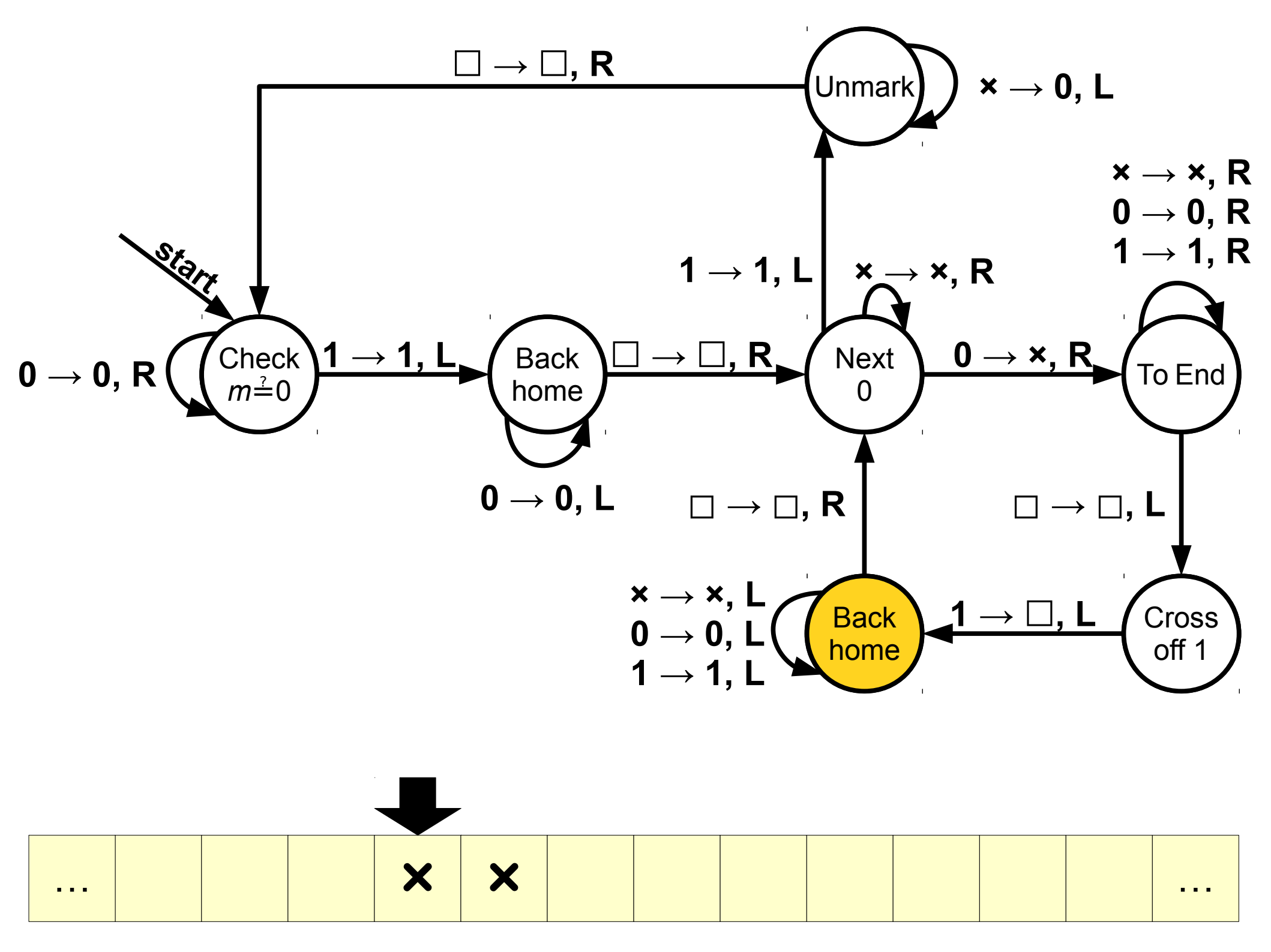

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

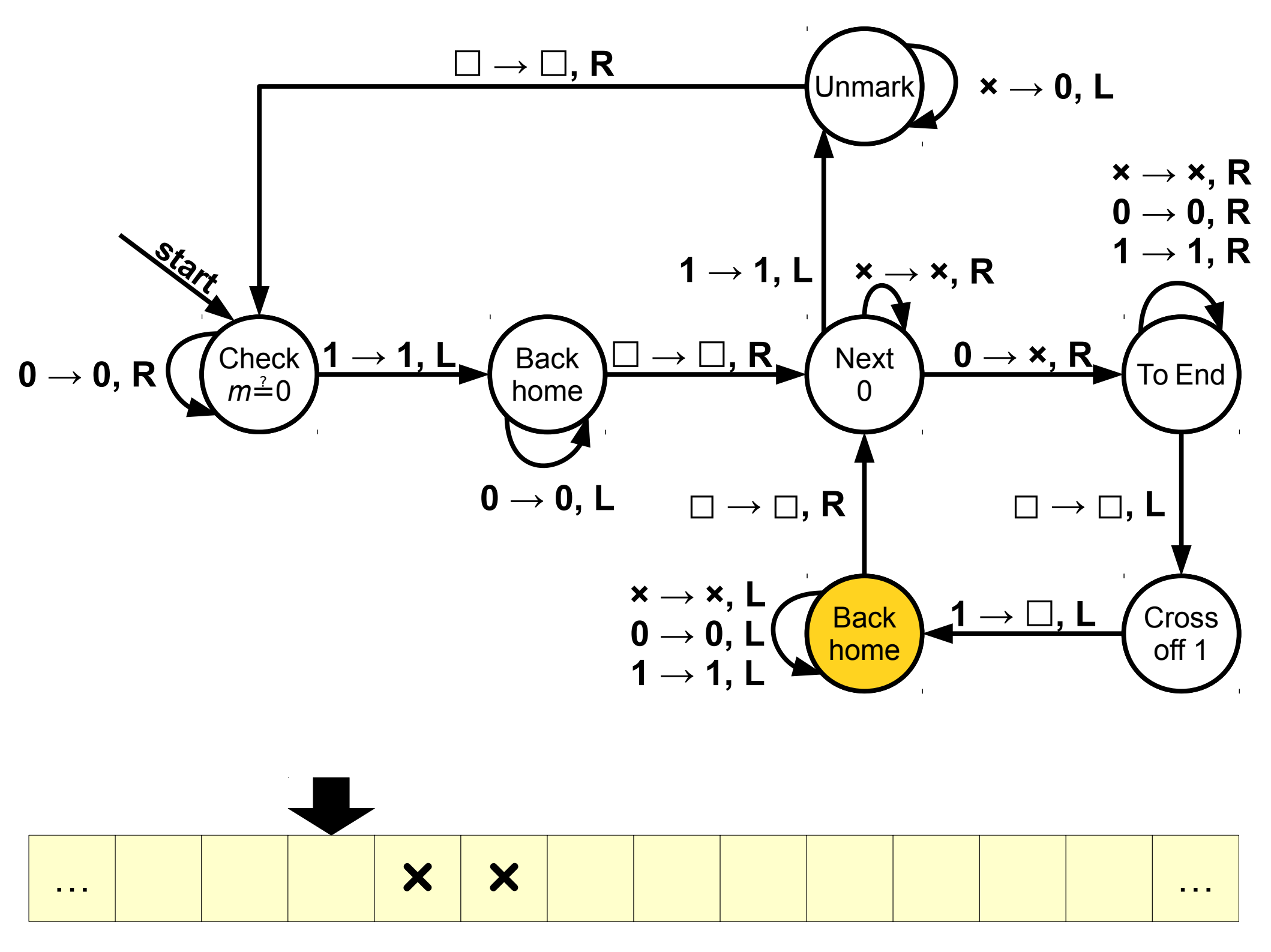

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
1
…

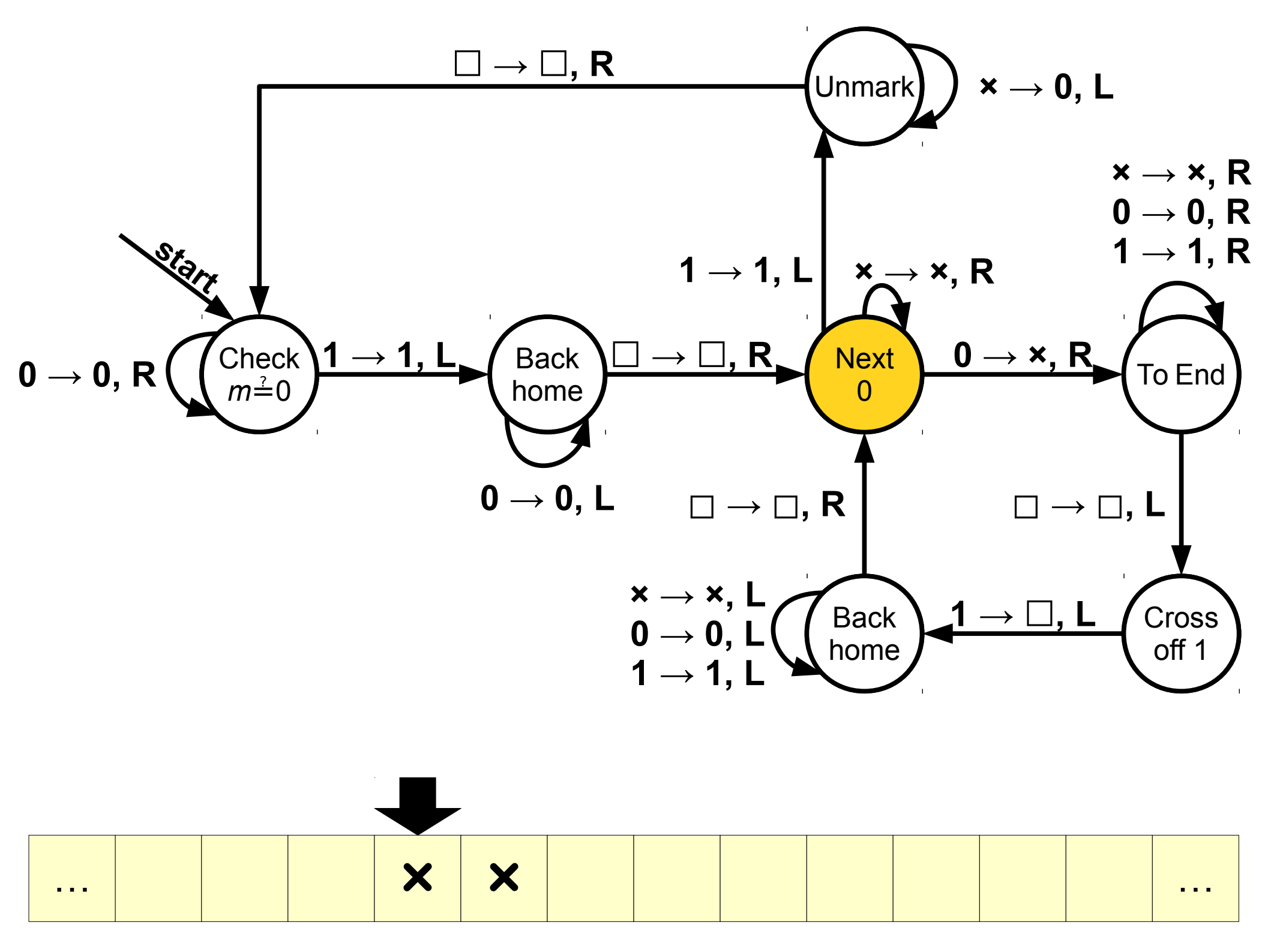

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × × 1 …

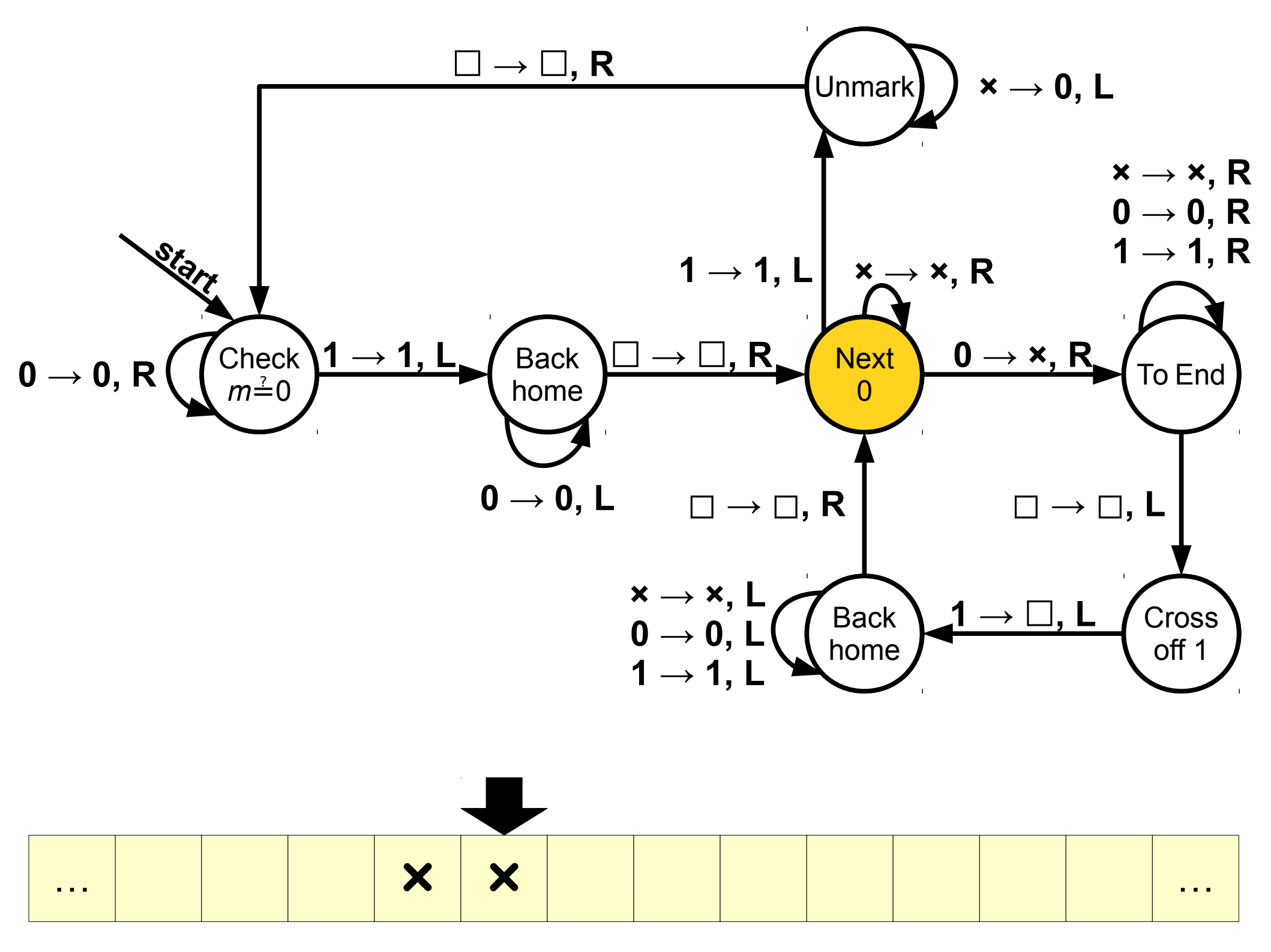

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
1
…

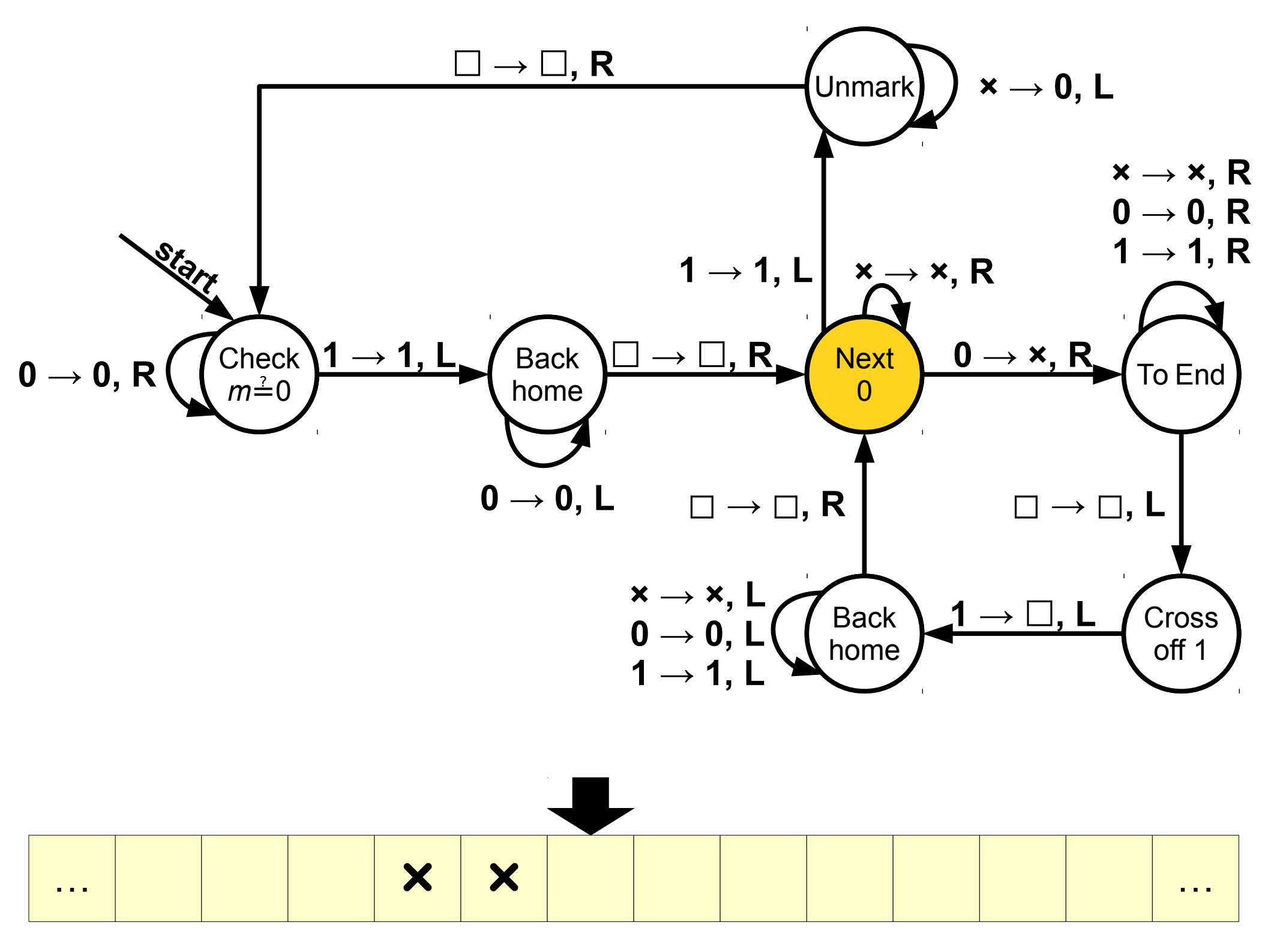

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

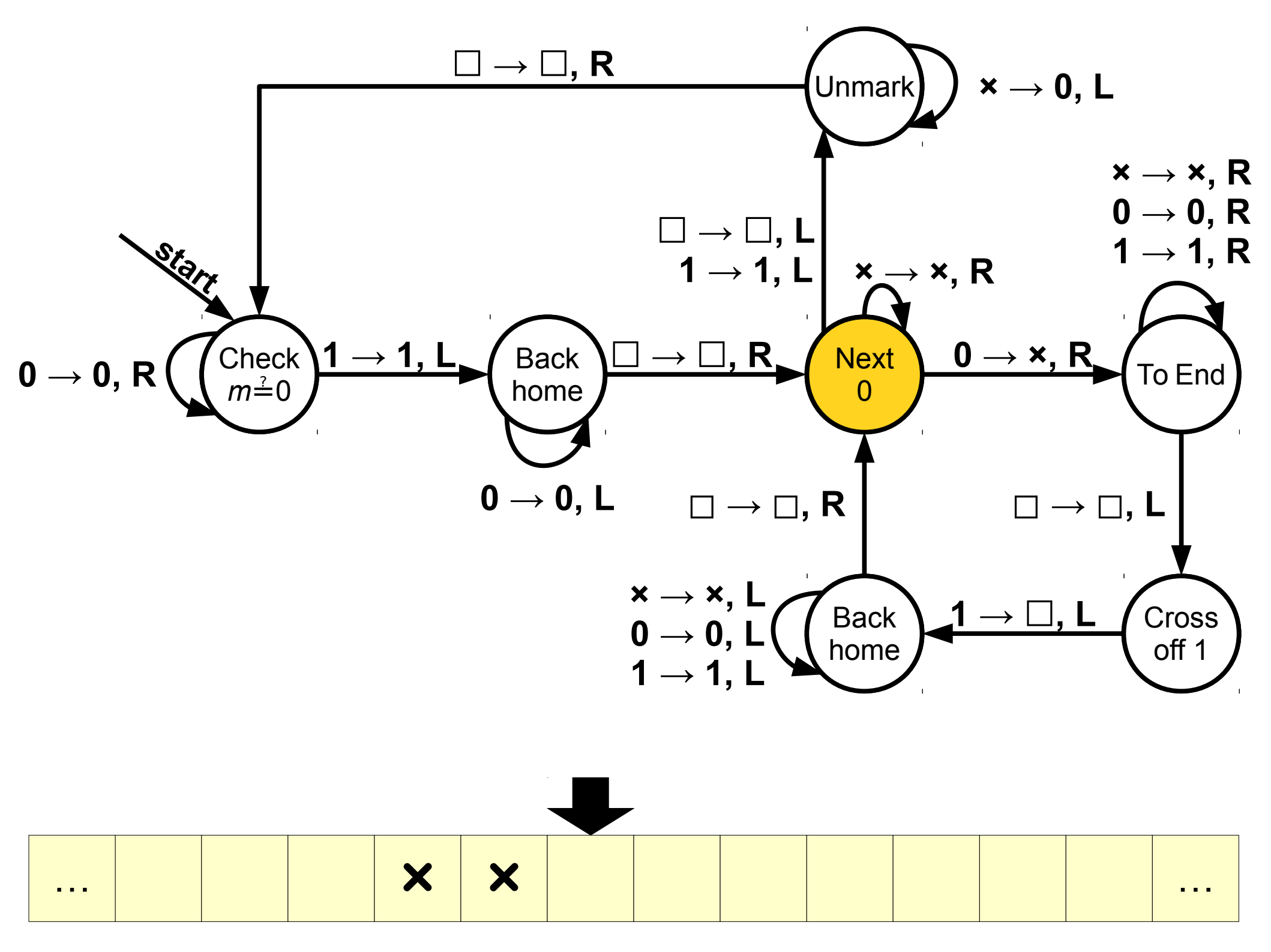

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

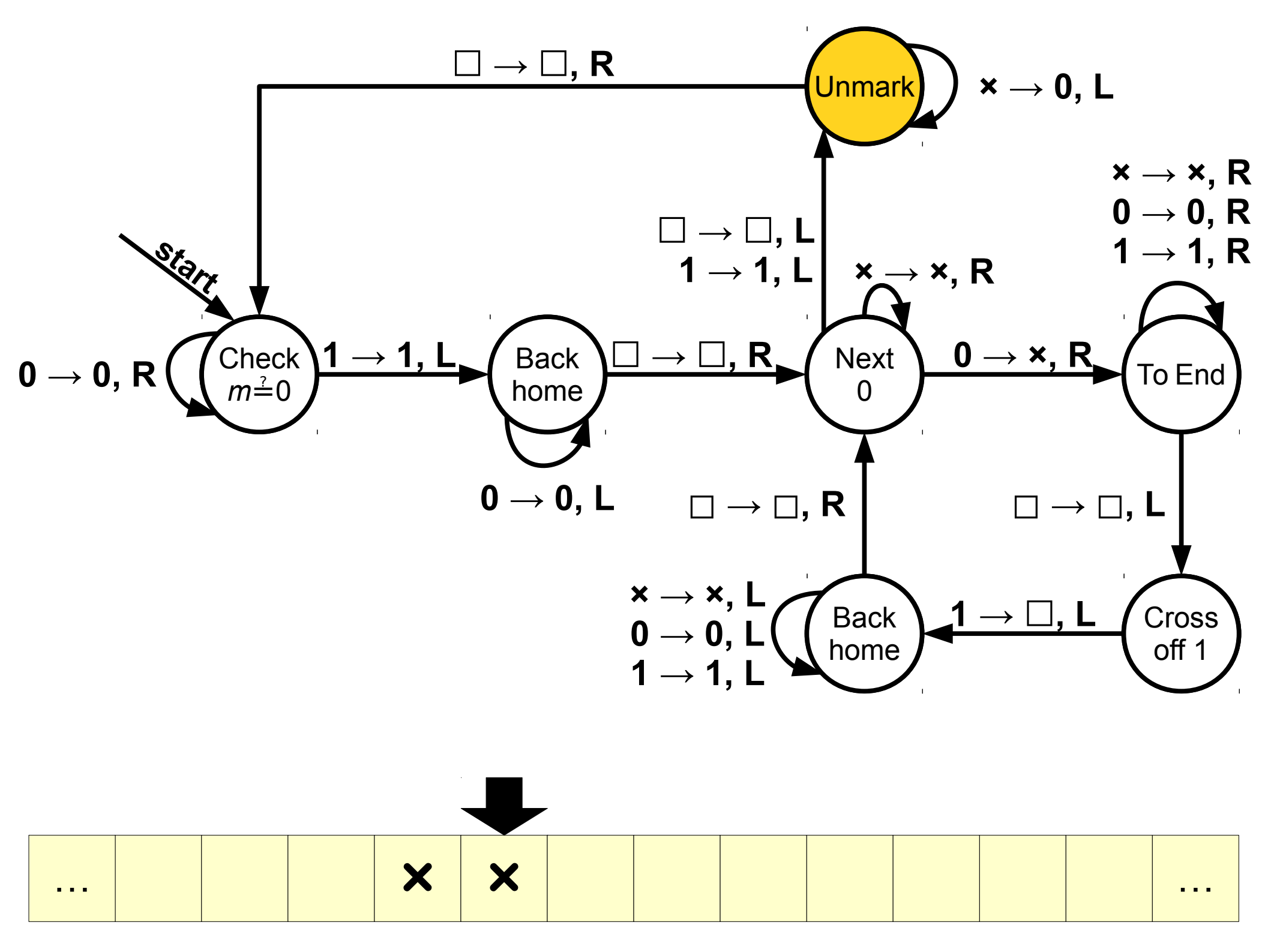

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

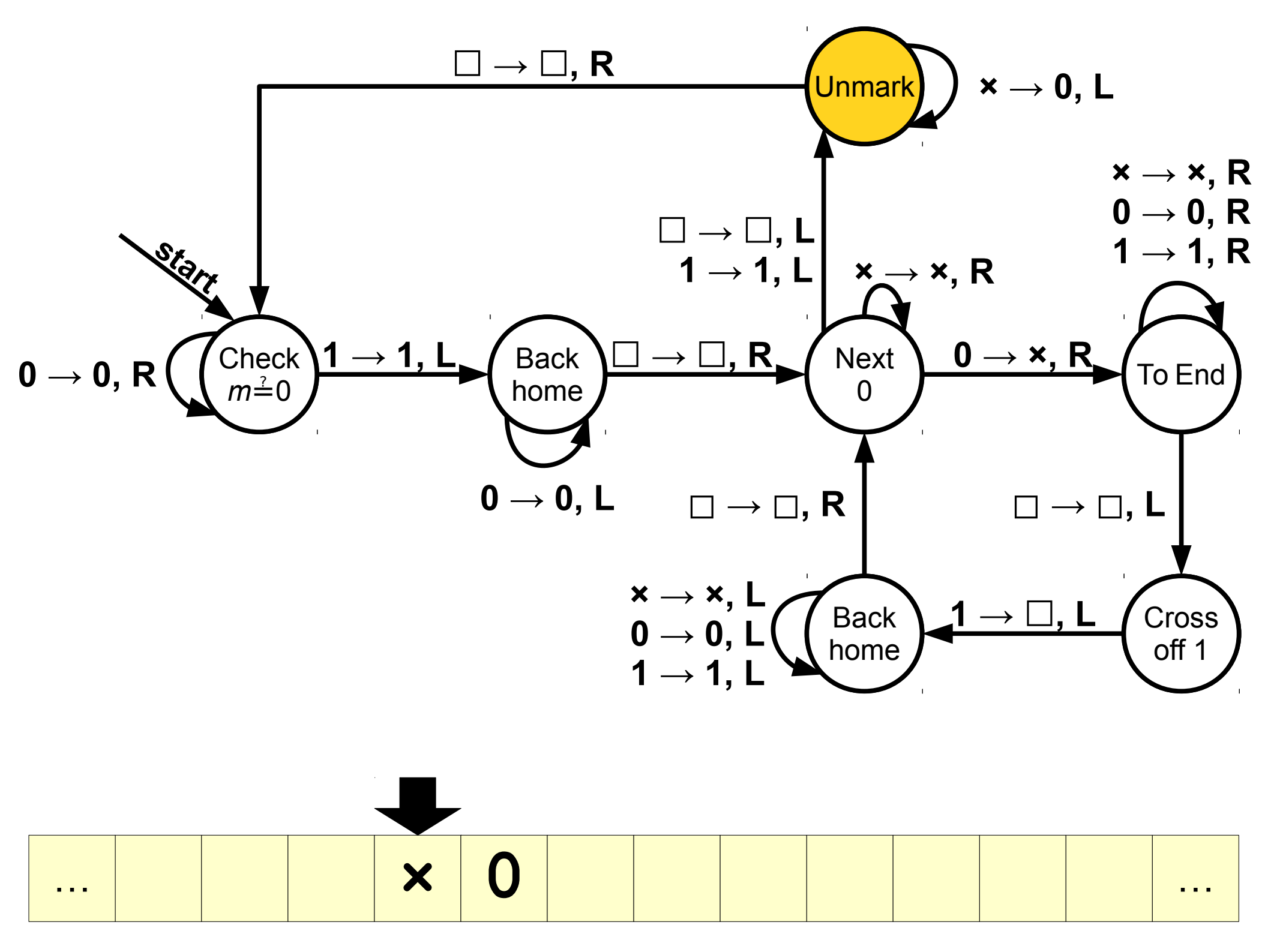

start
Check
m=0
Unmark
Back
home
Next
0
To End
Back
home
Cross
off 1
0 → 0, R
1 → 1, L
□ → □, R
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
0 → ×, R
0 → 0, L
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
…
…

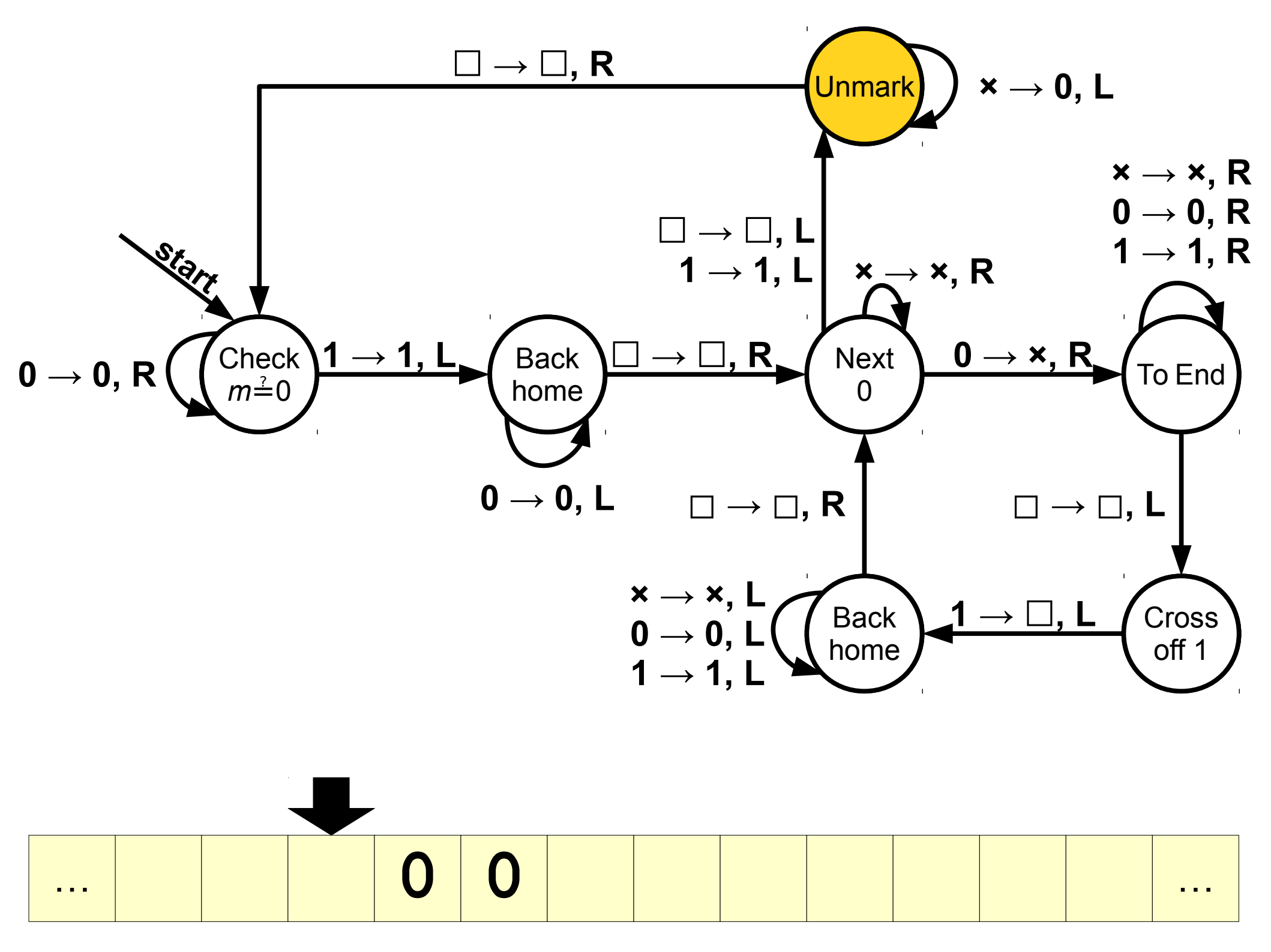

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m ≟ 0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

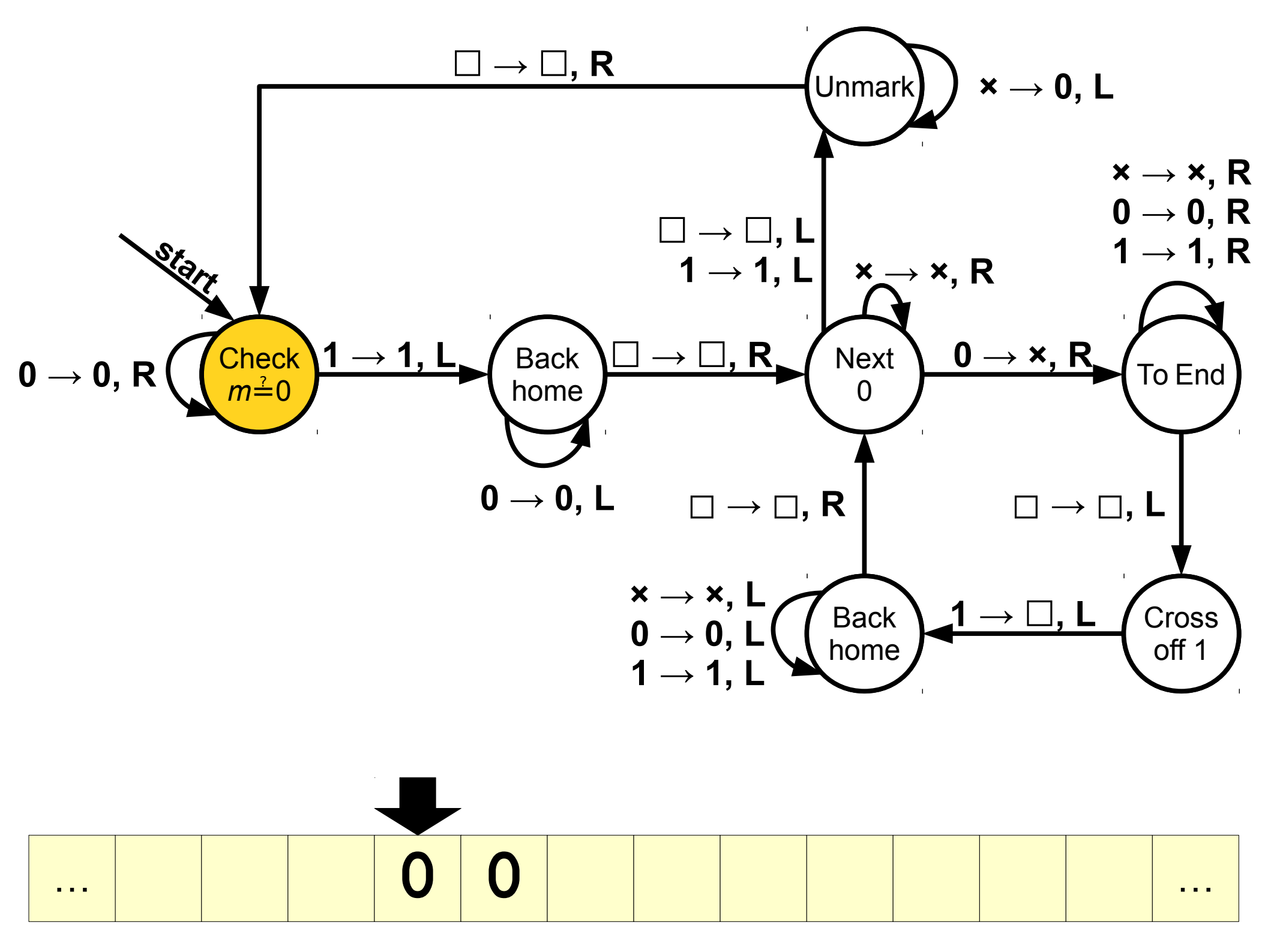

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
start
1 → 1, L
× → ×, R
Check
m=?0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, R
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

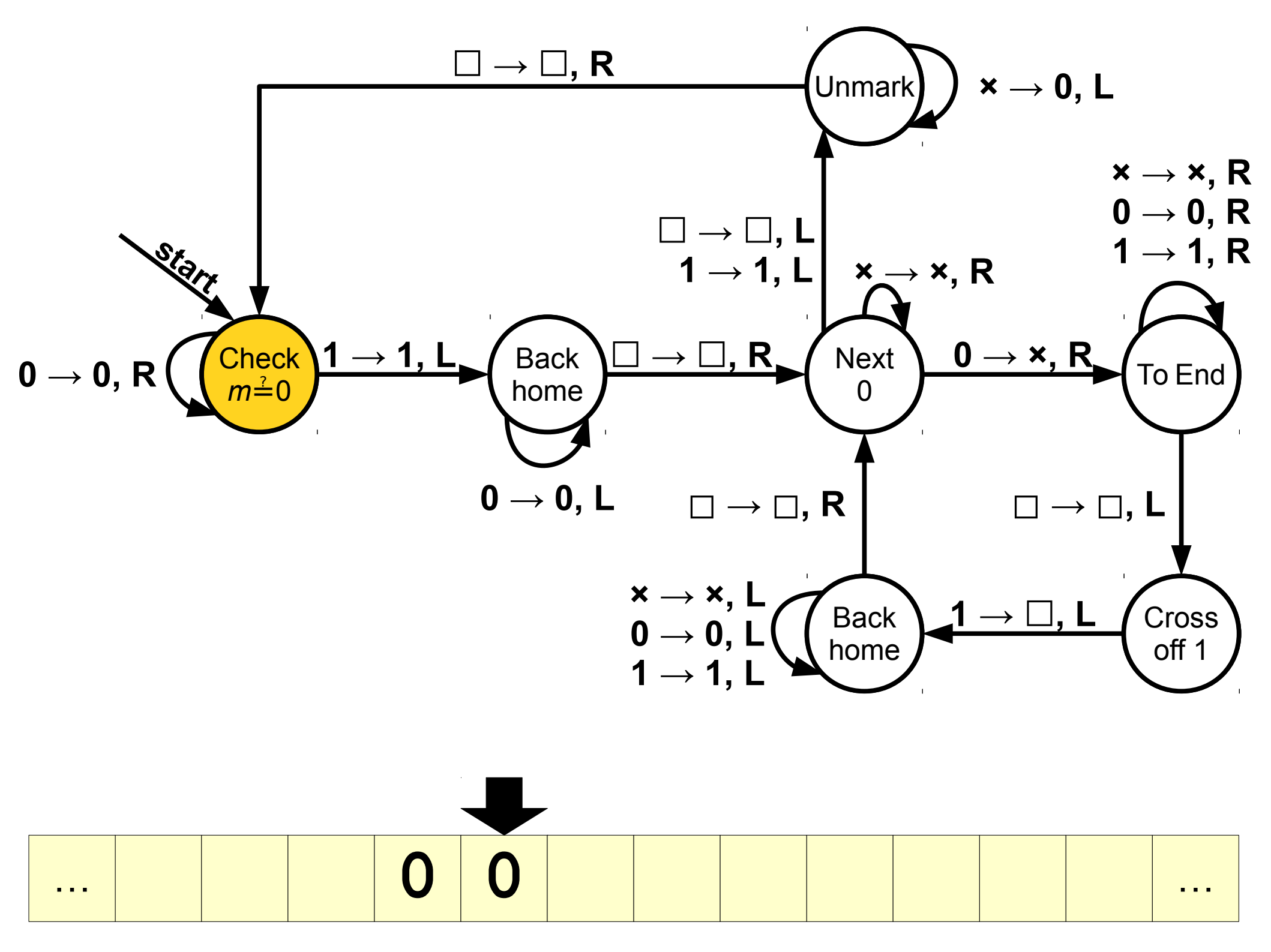

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

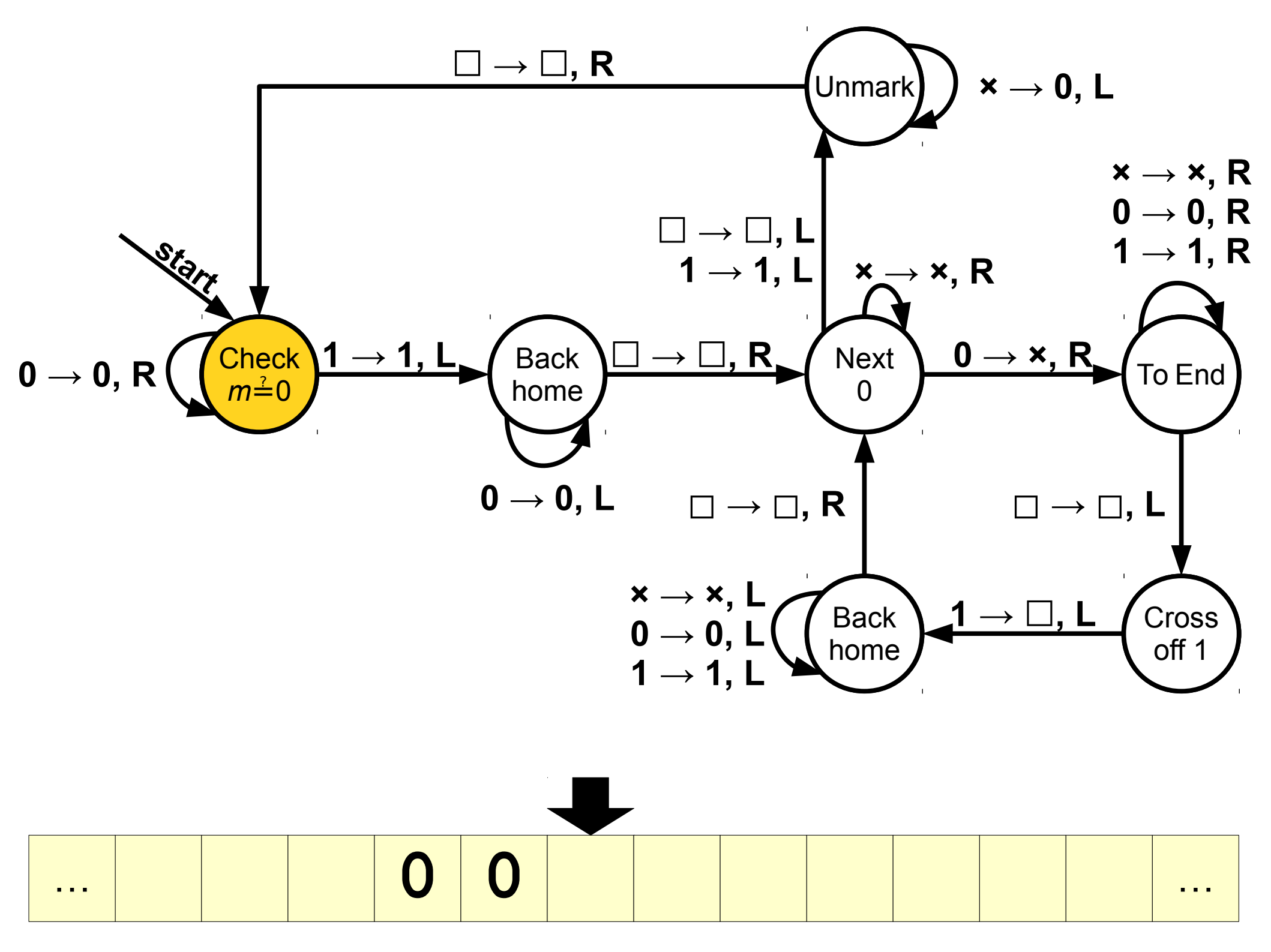

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…

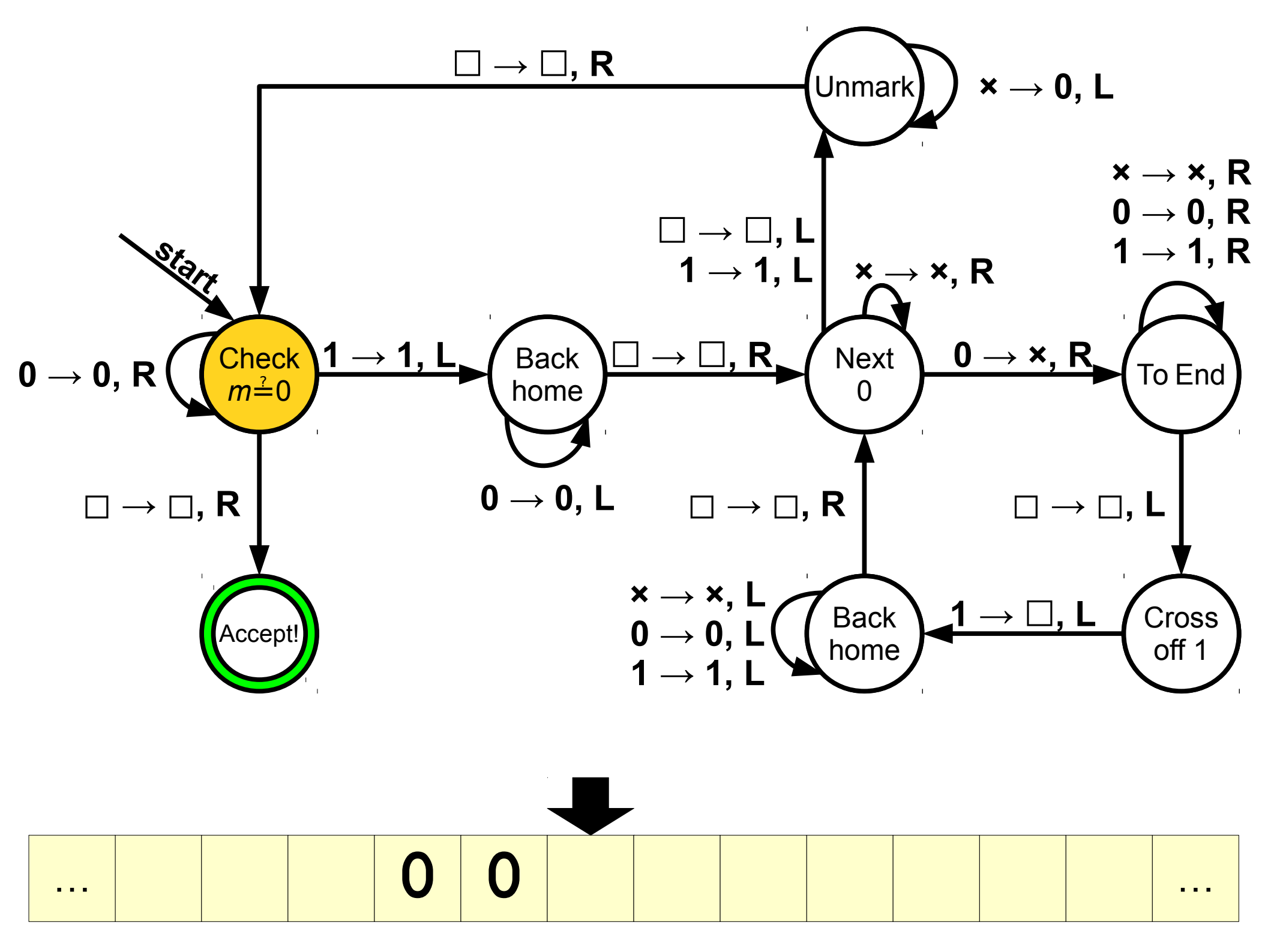

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
…

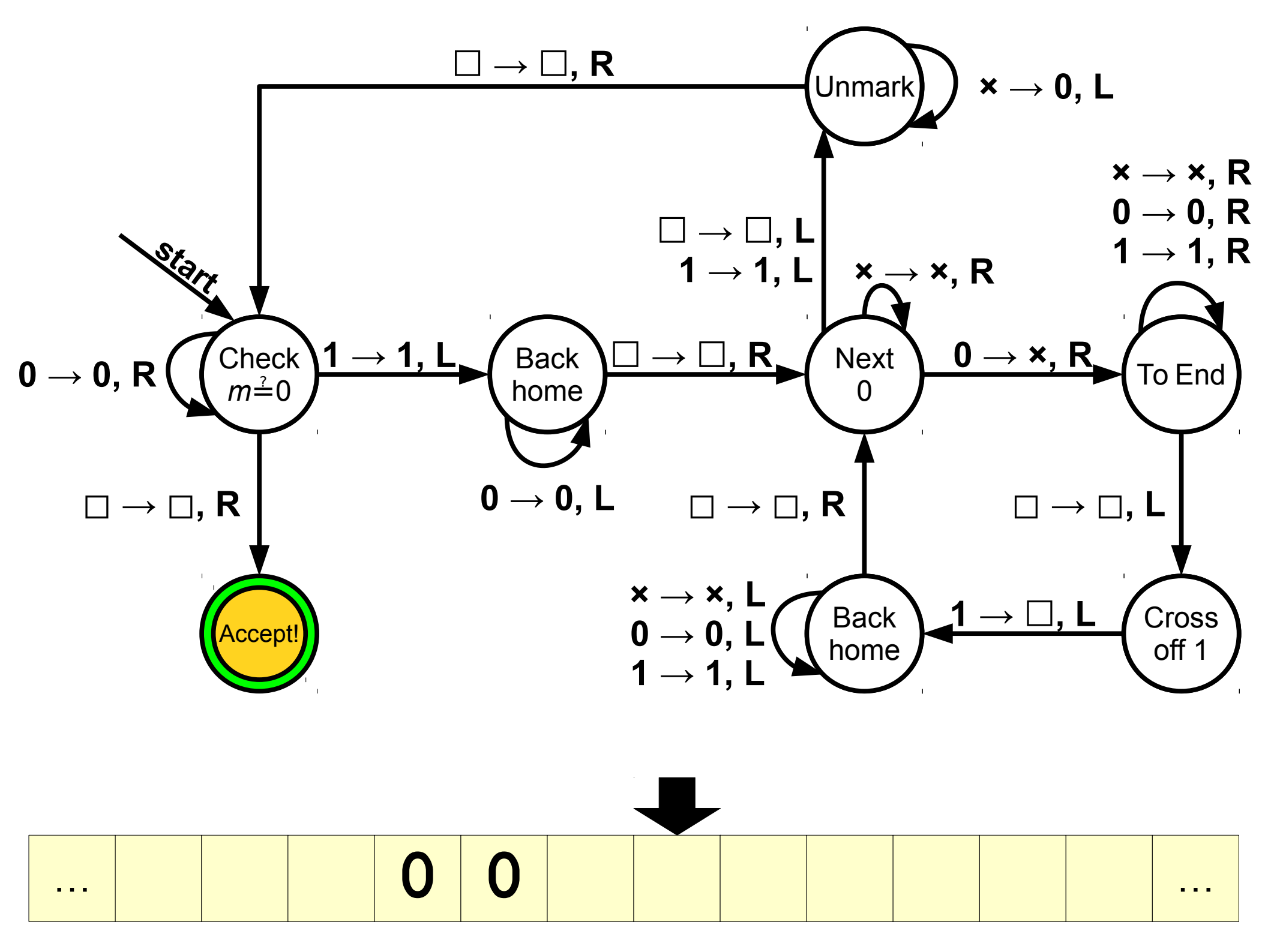

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
0 → 0, R
Check
m=?0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

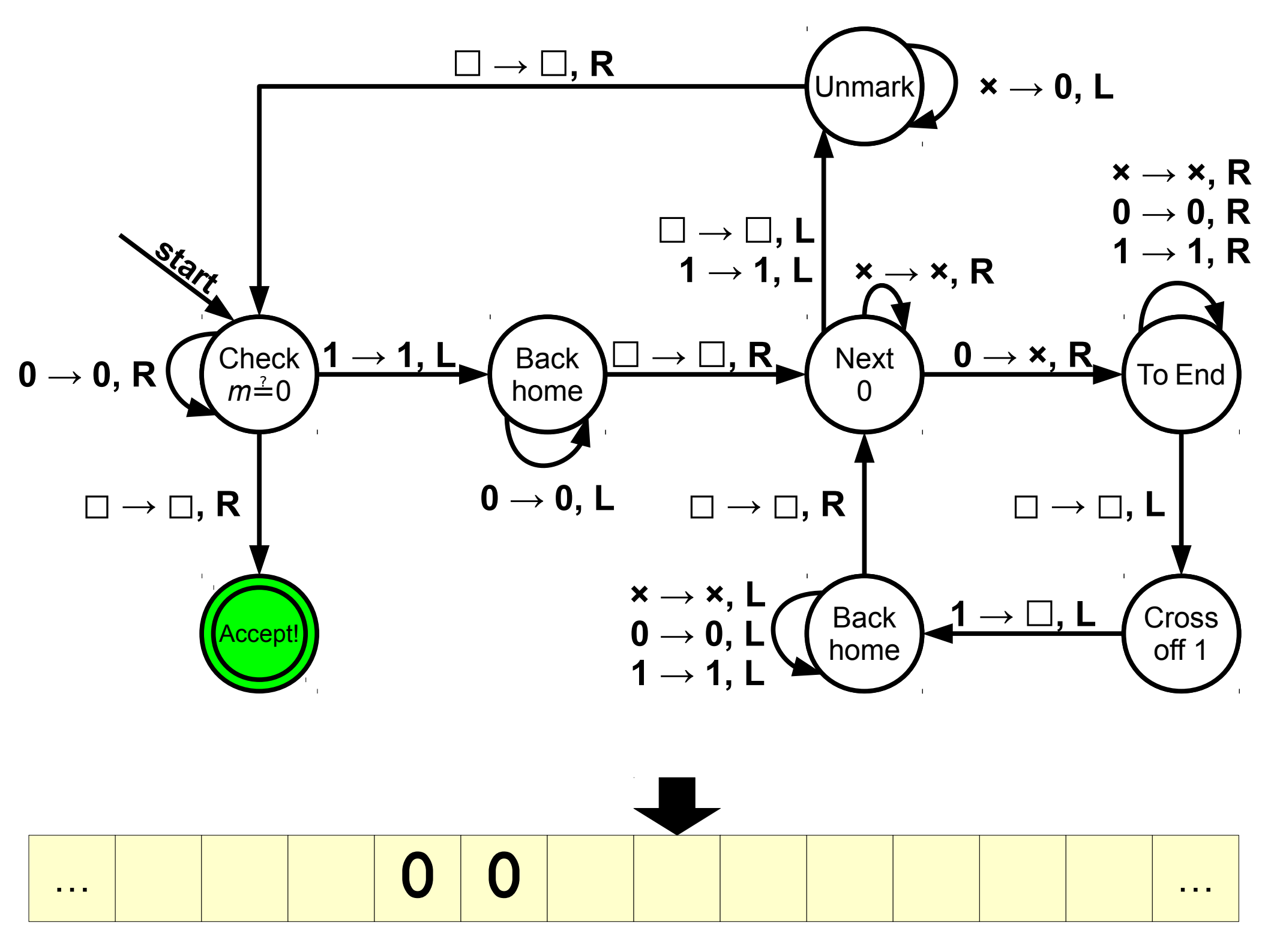

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

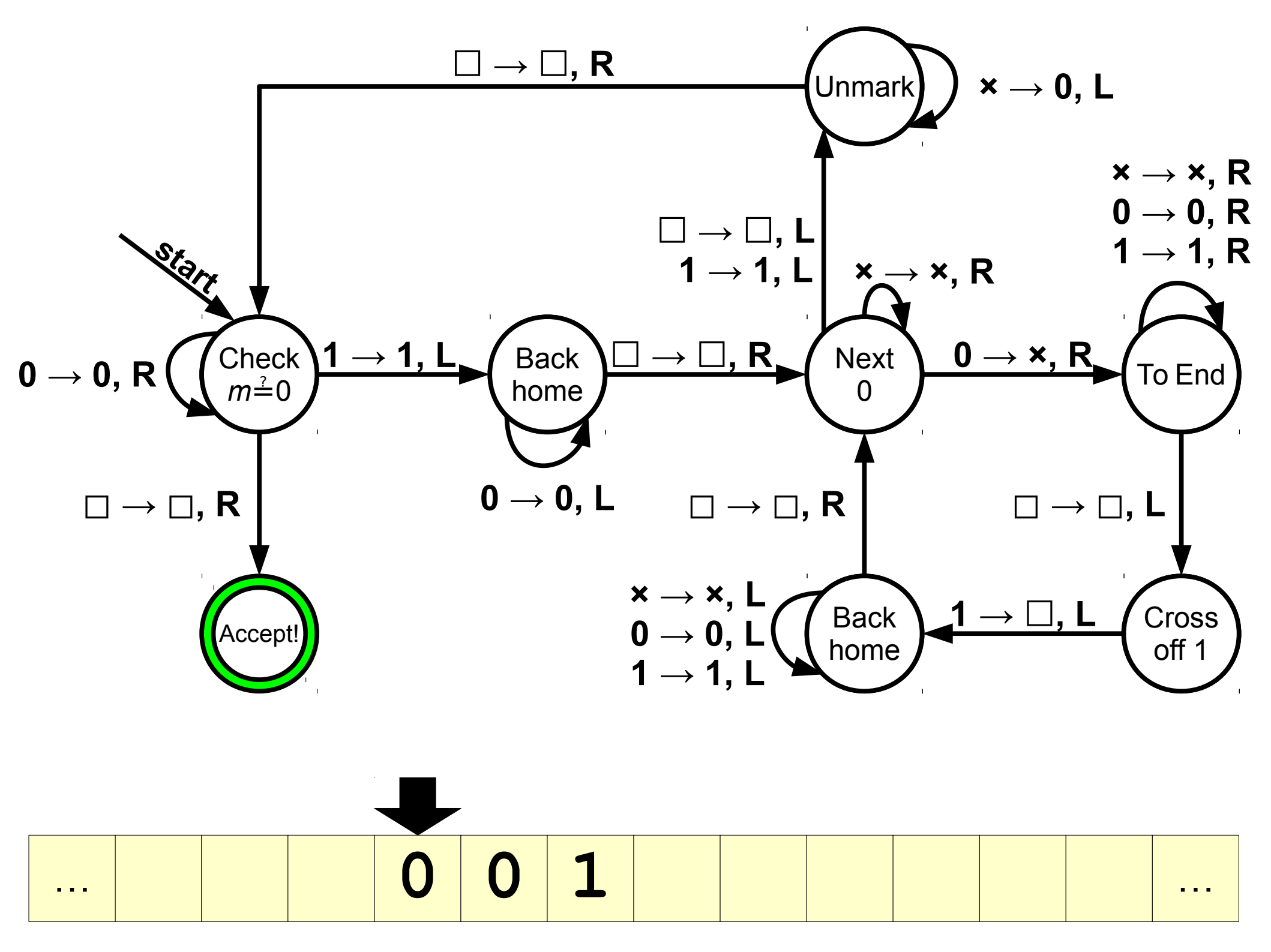

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
0 → 0, R
Check
m=0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

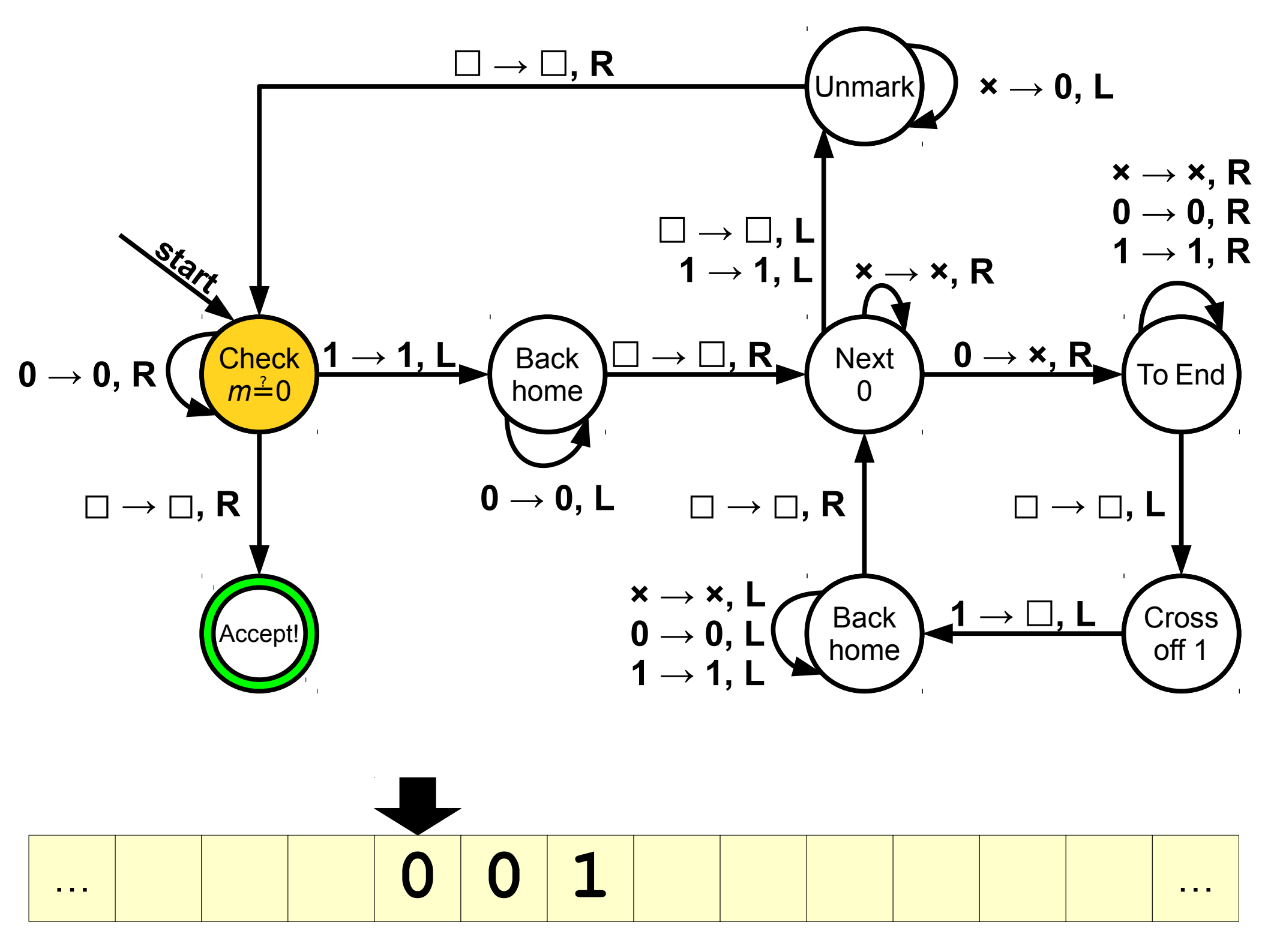

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
0 → 0, R
Check
m≟0
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

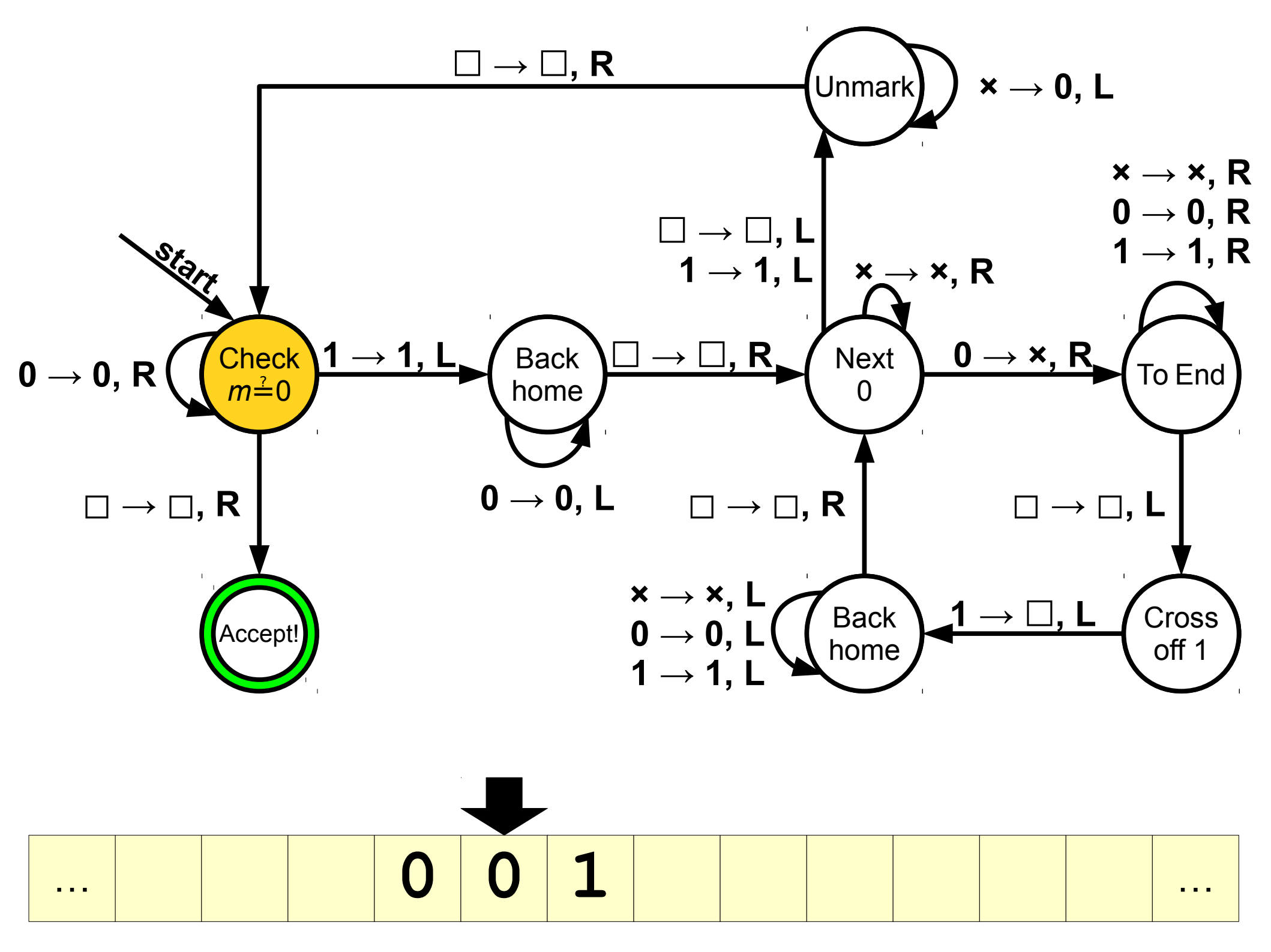

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

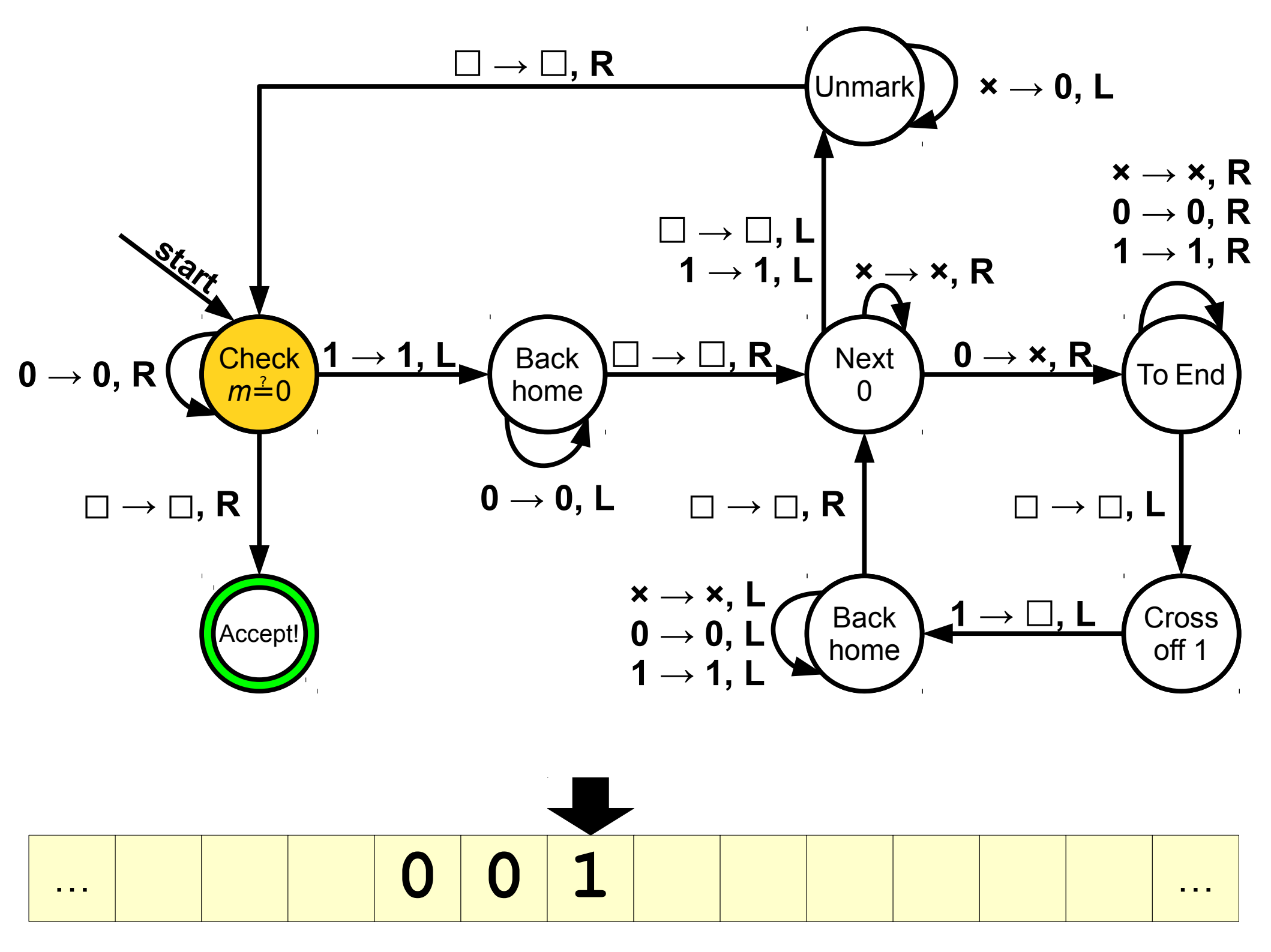

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

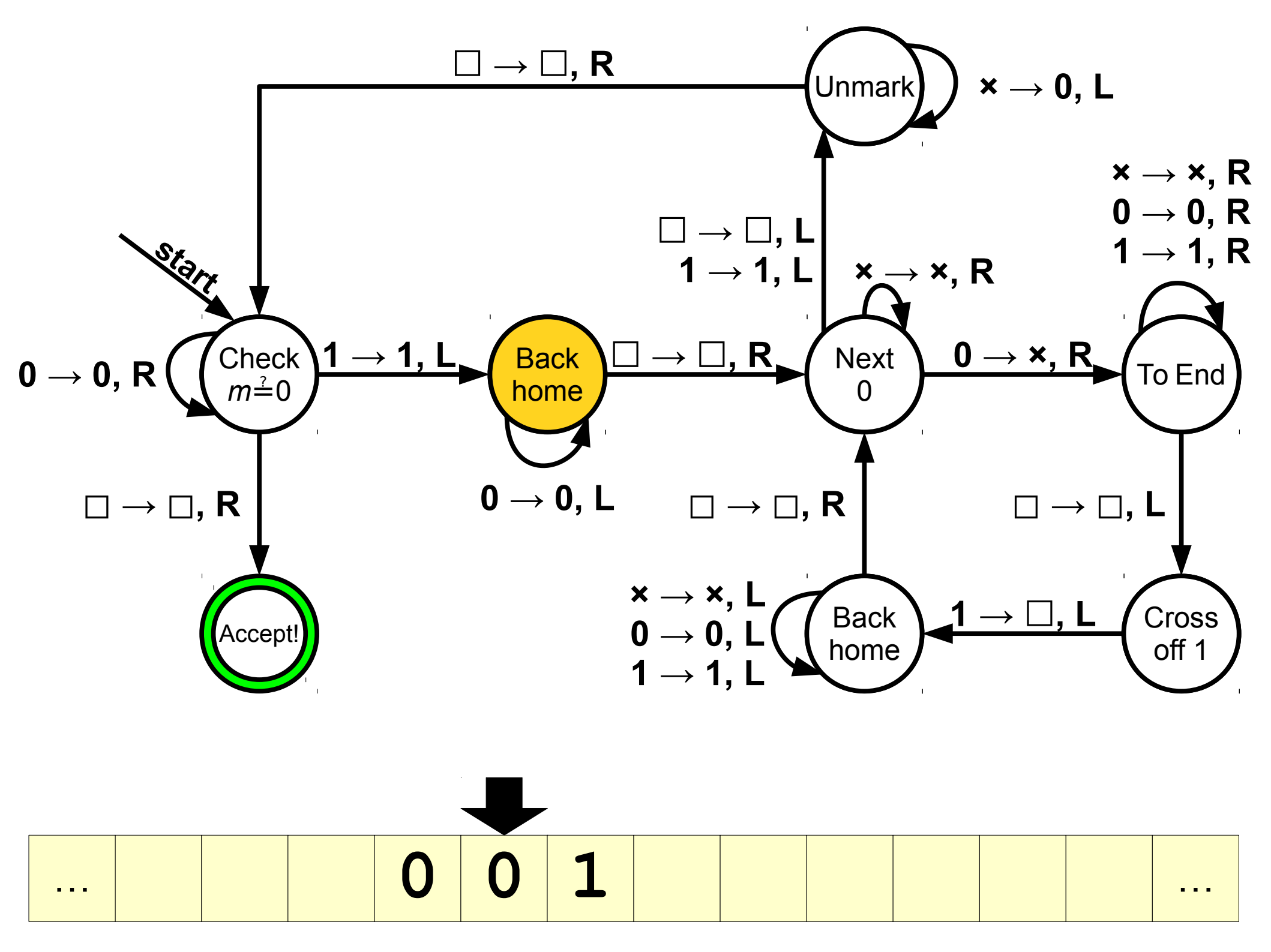

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
…

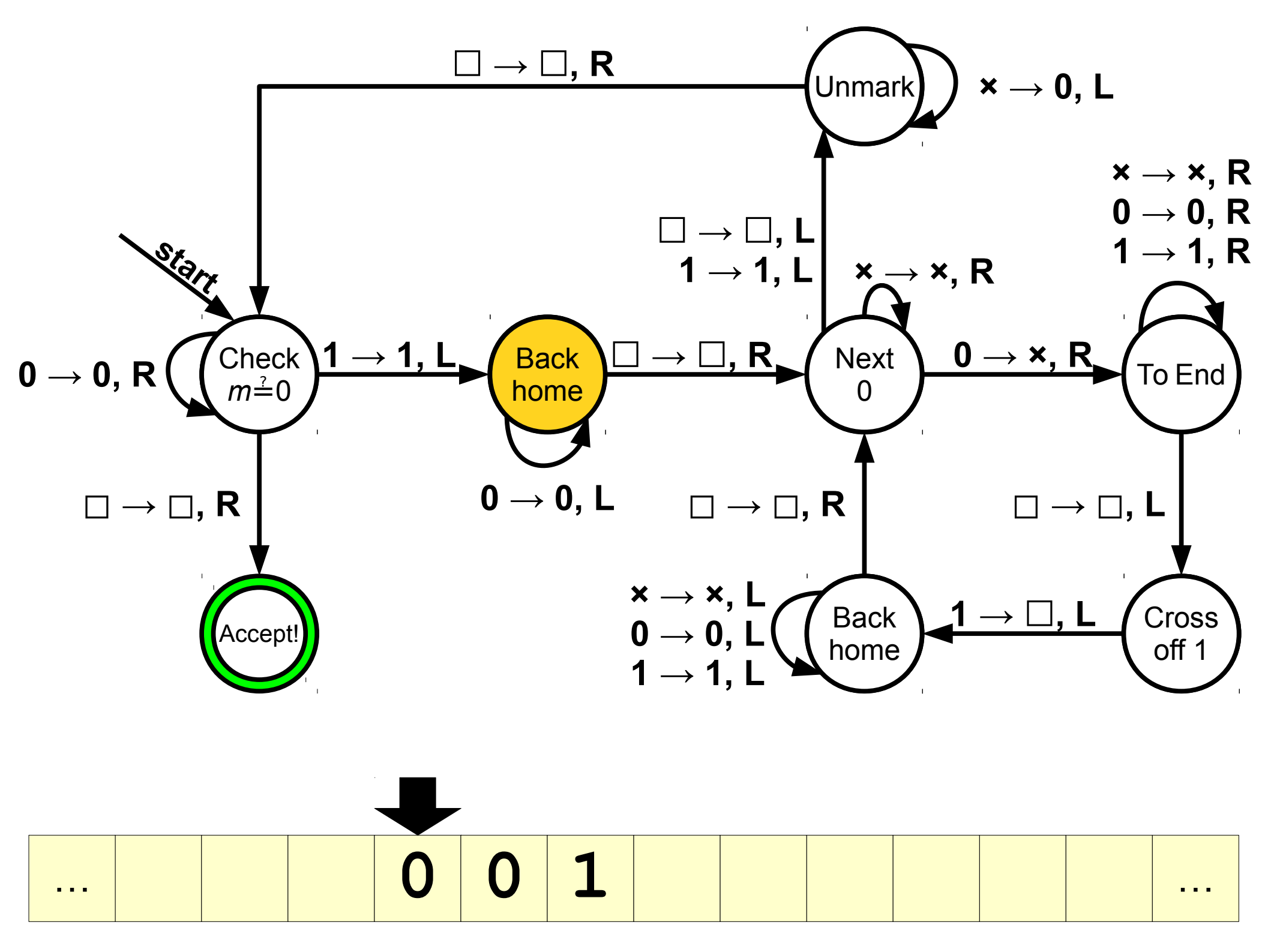

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

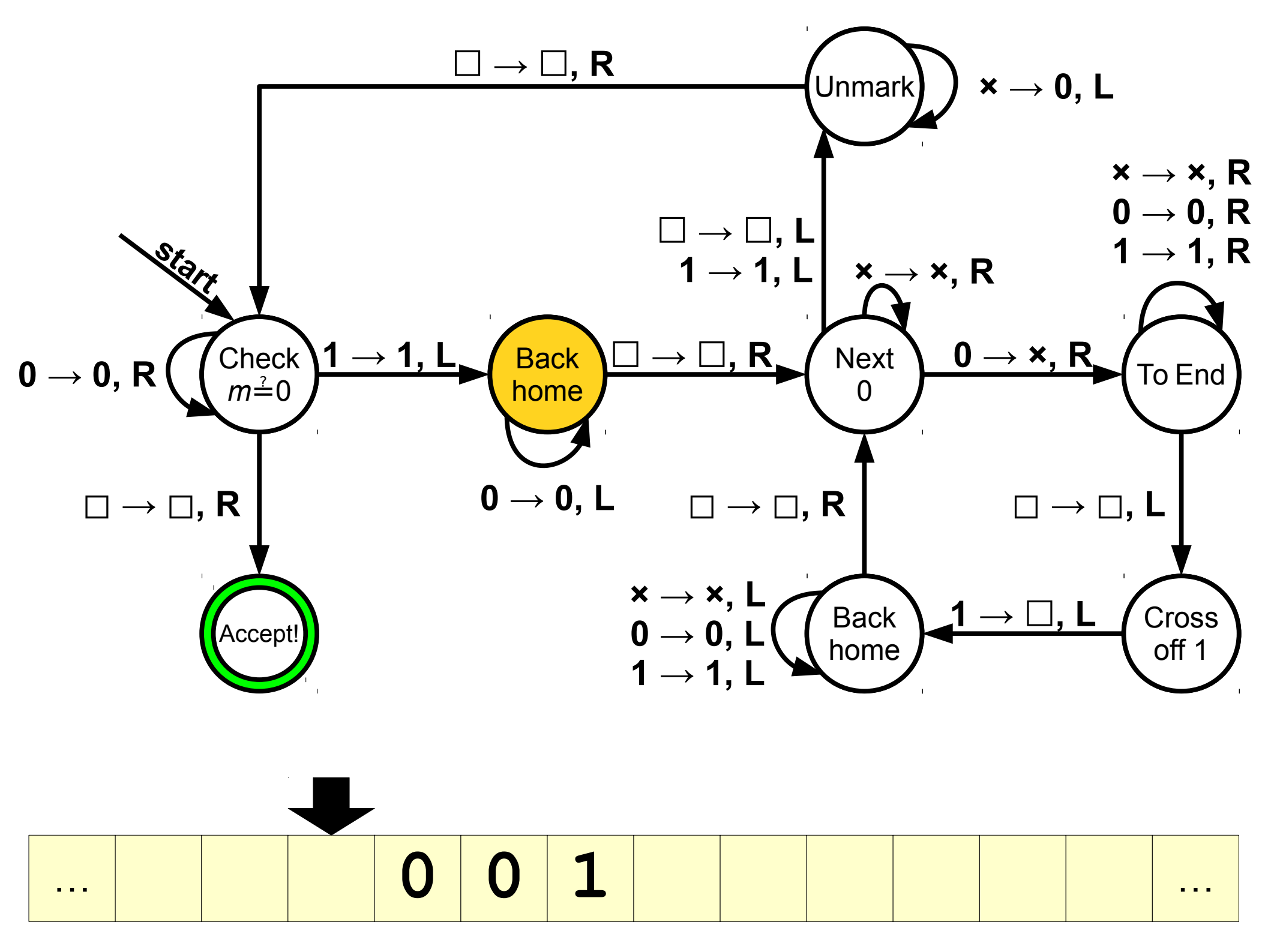

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

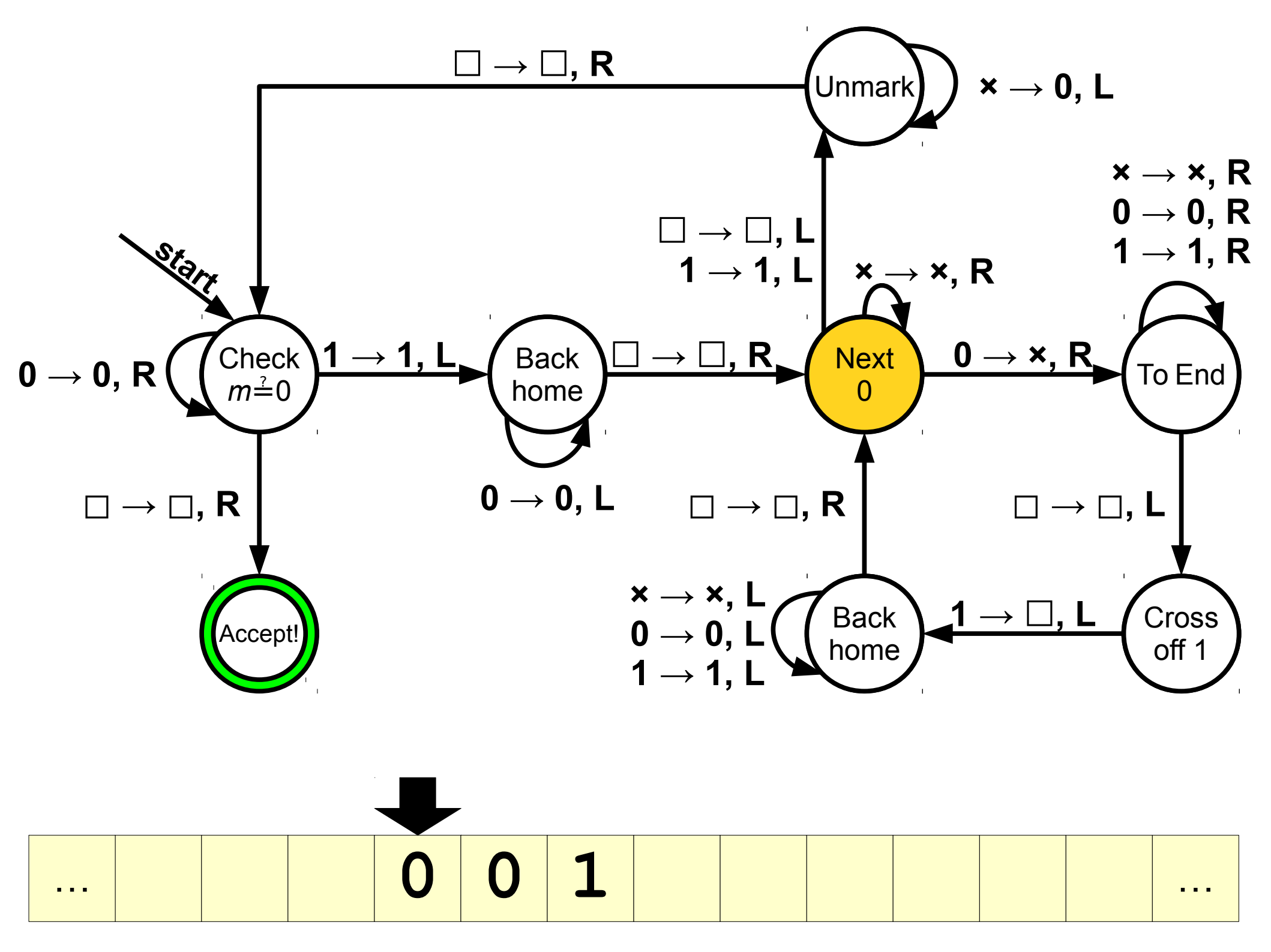

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

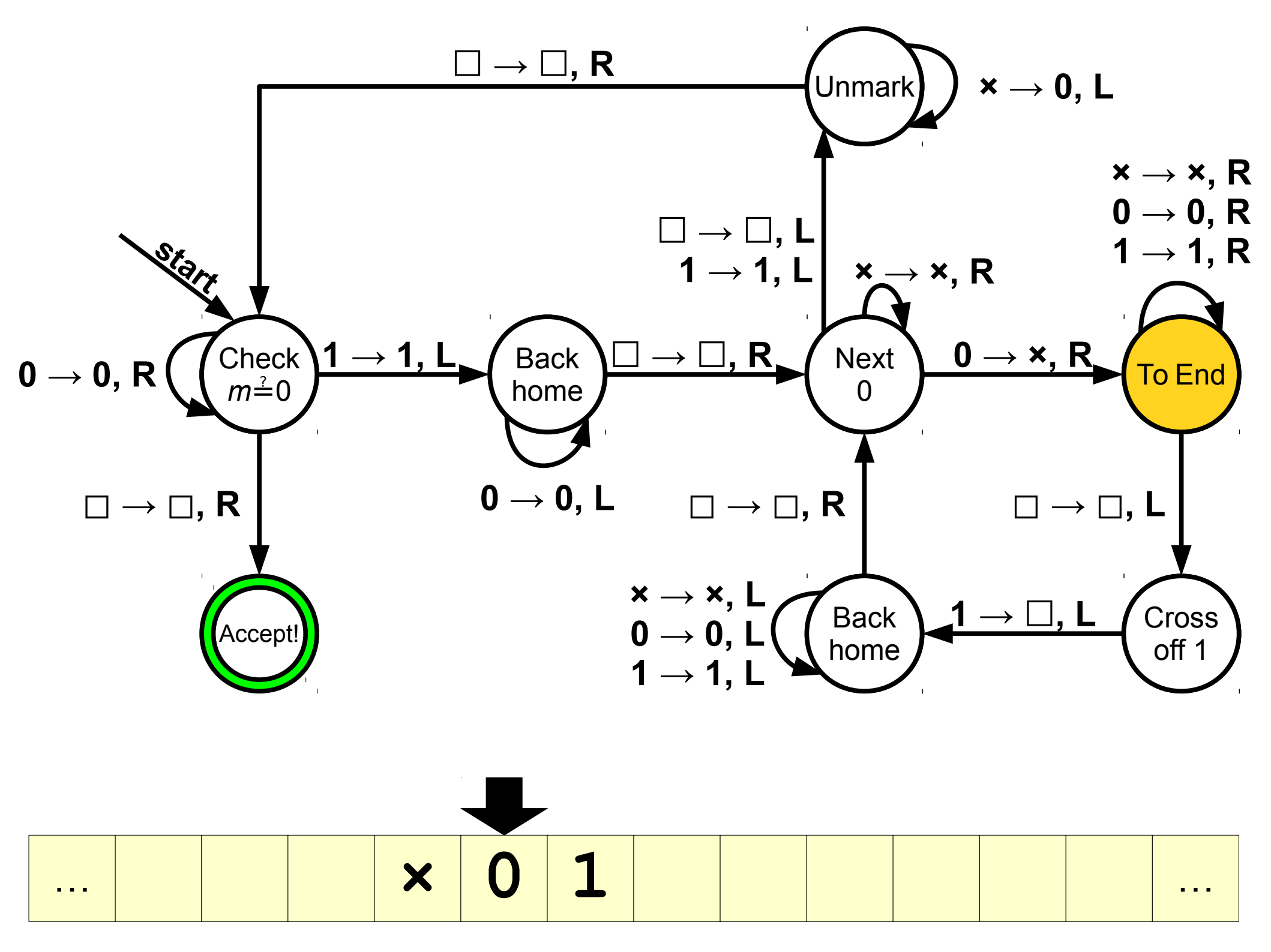

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… 0 0 1 …

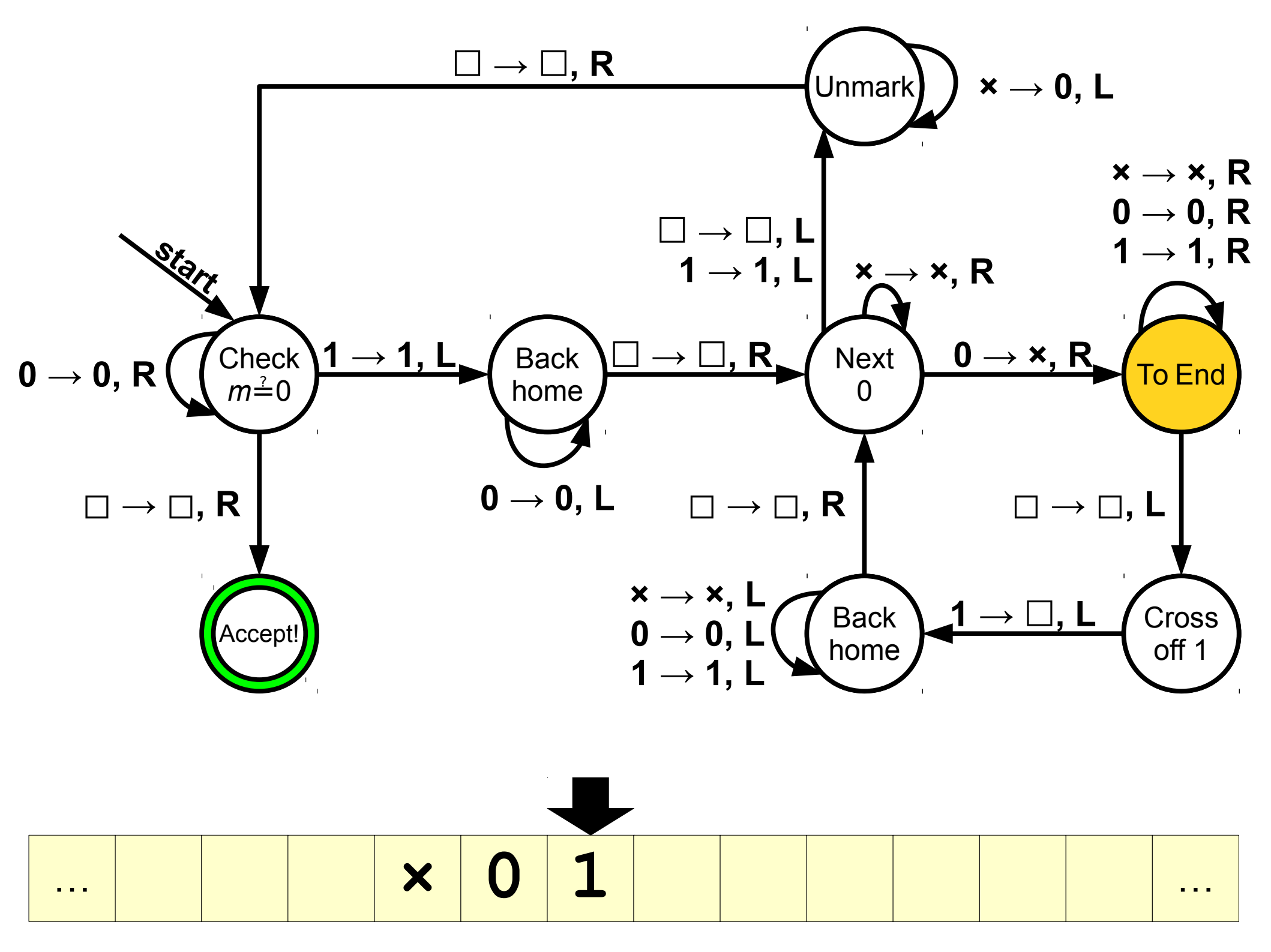

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

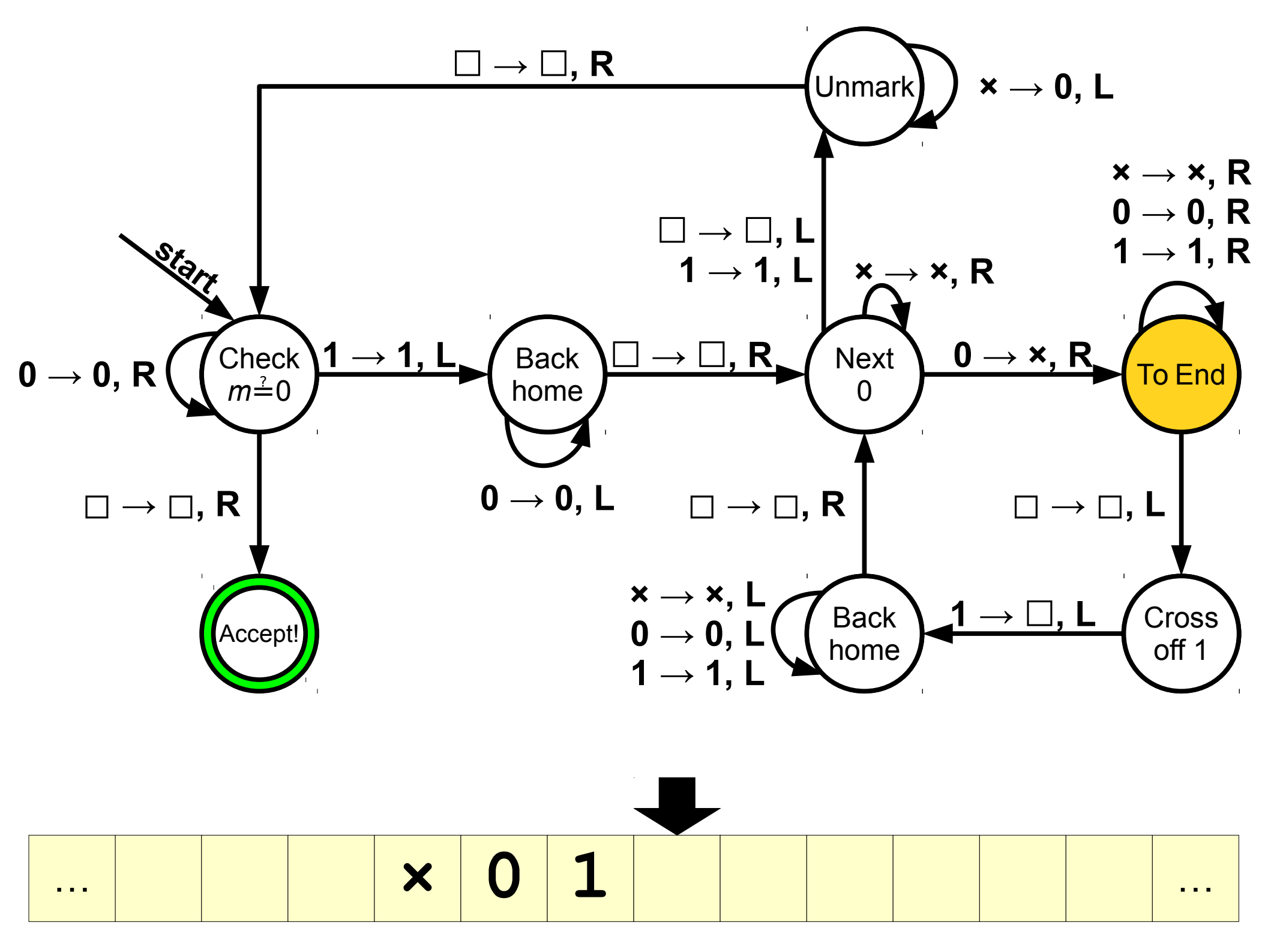

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

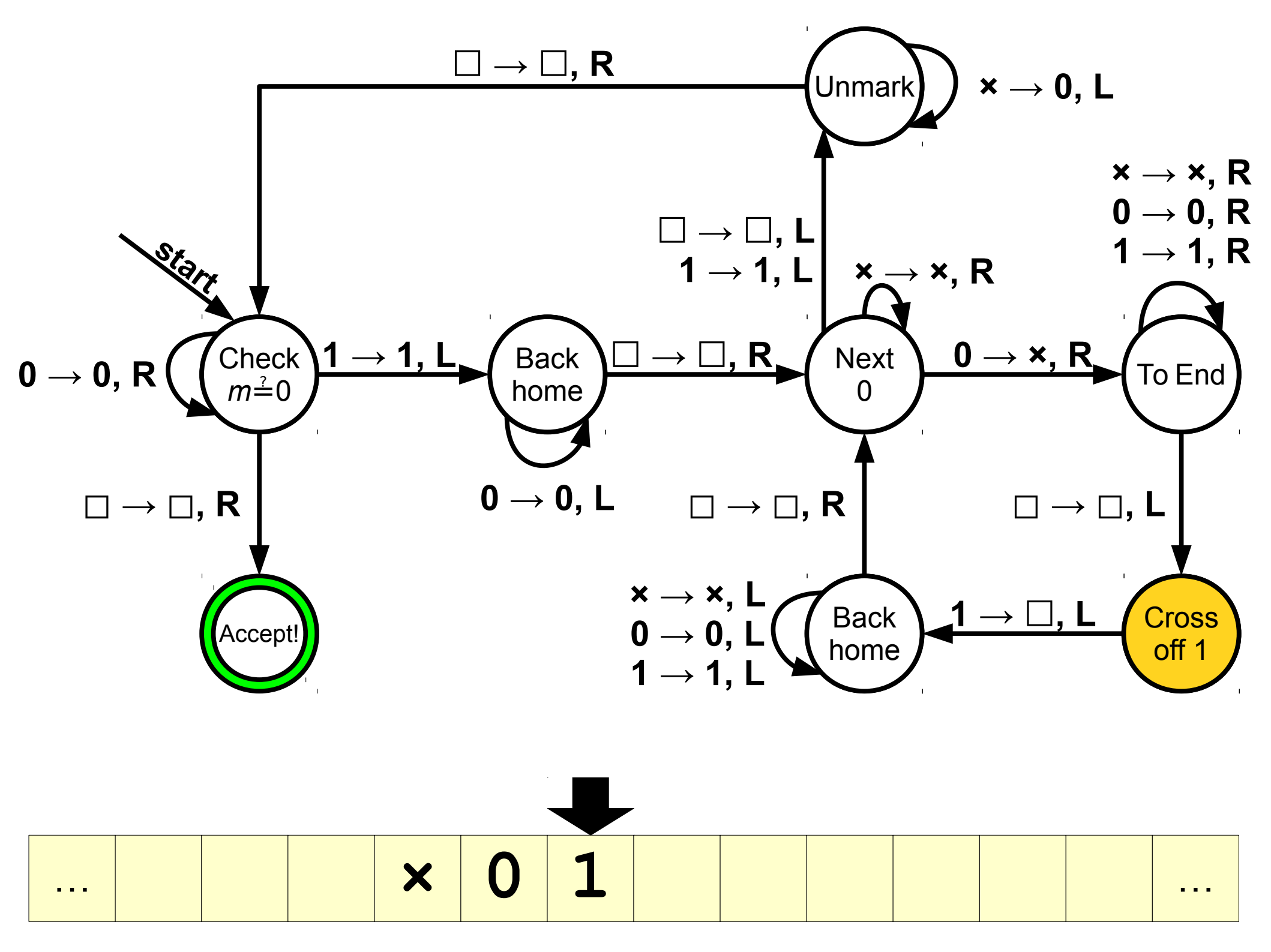

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

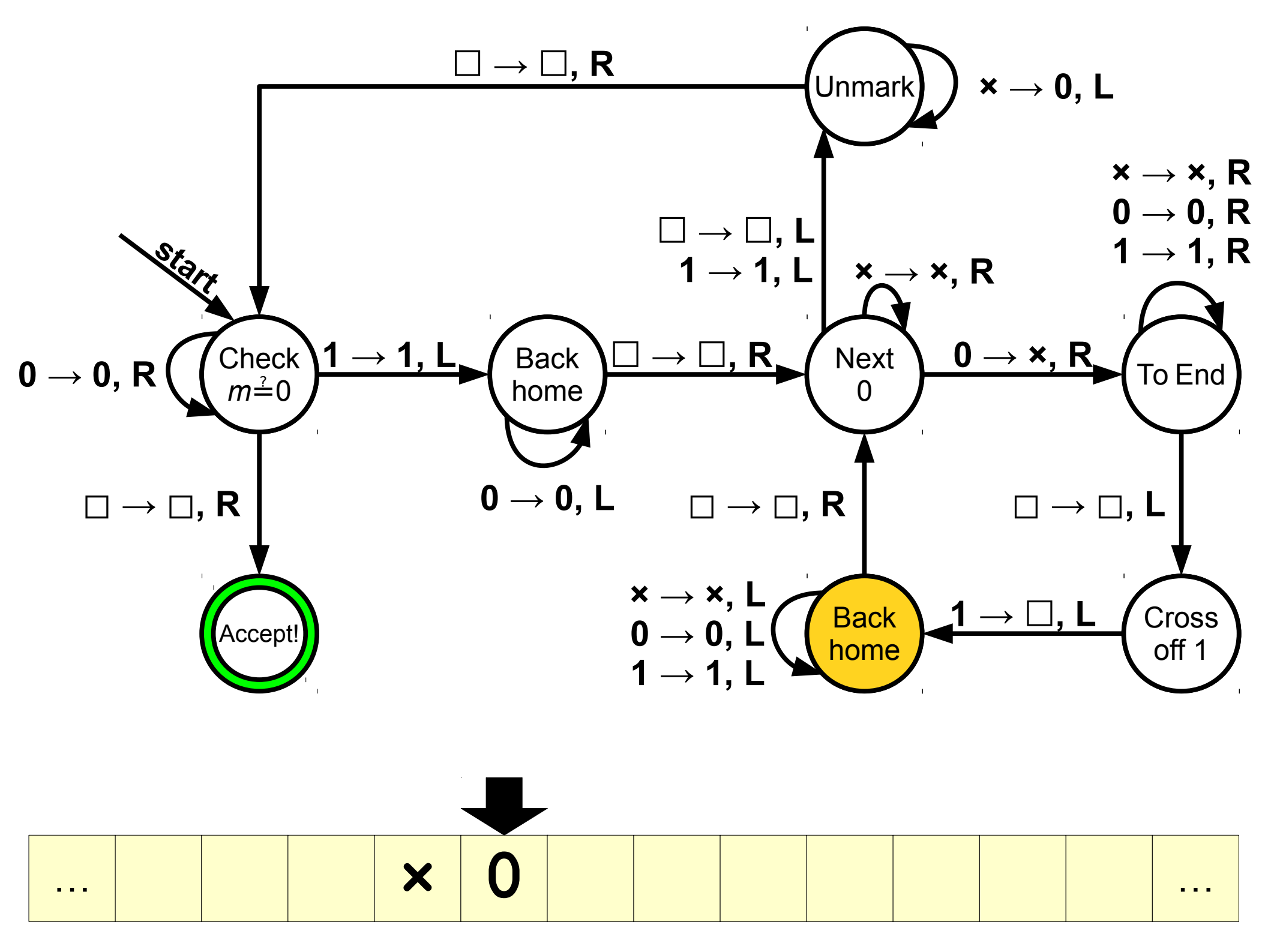

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m≟0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
0
0
1
…

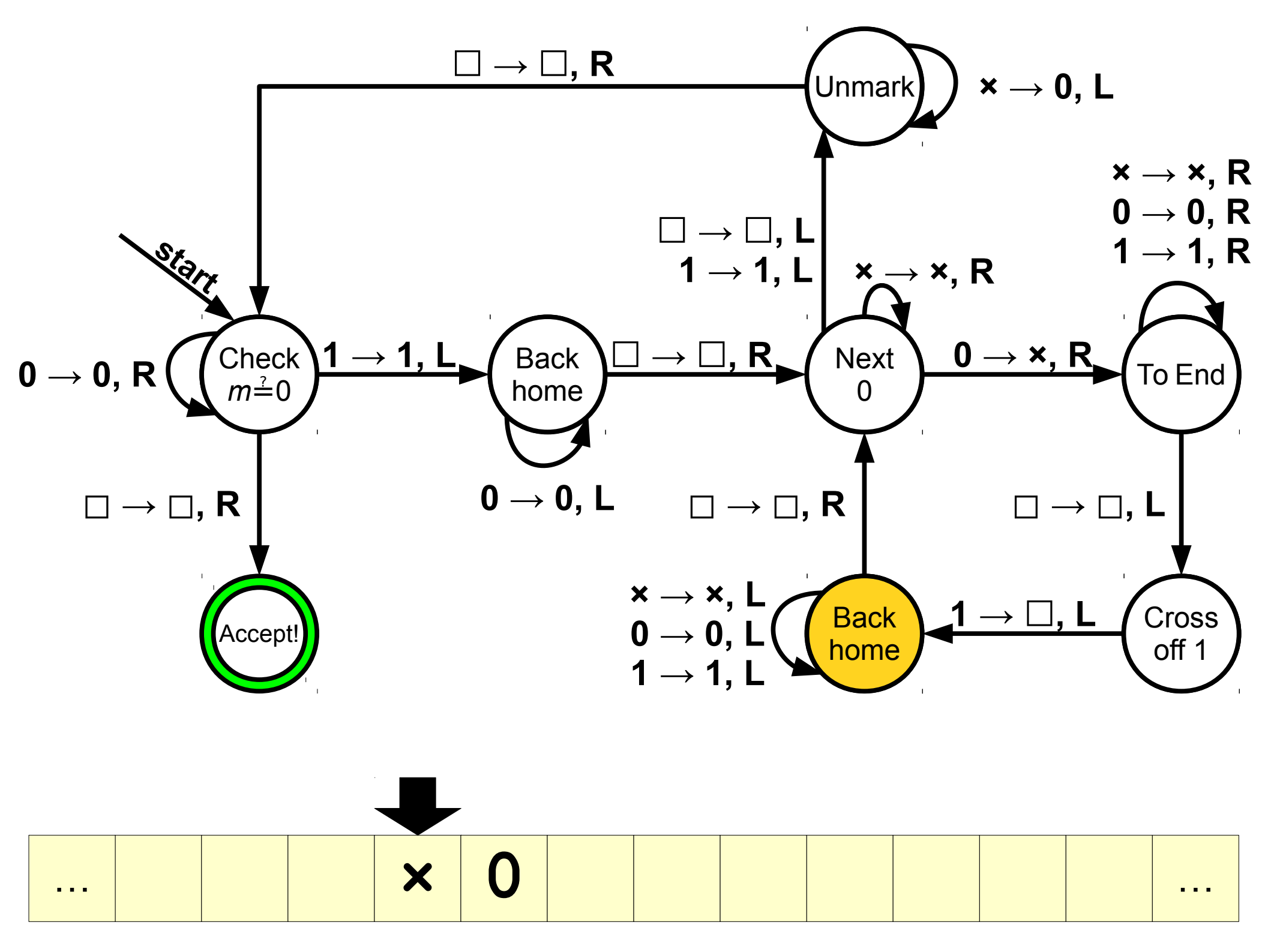

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

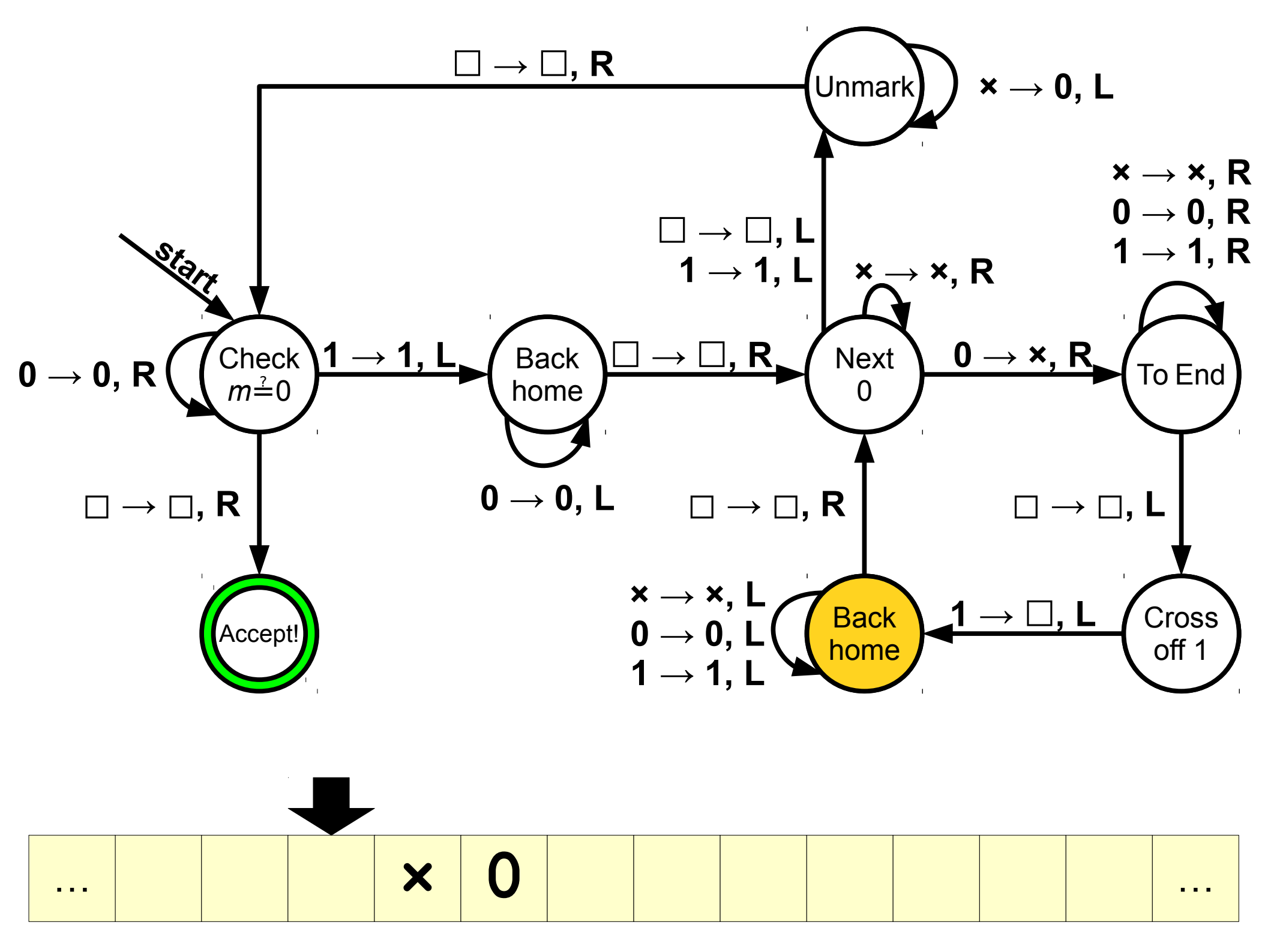

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

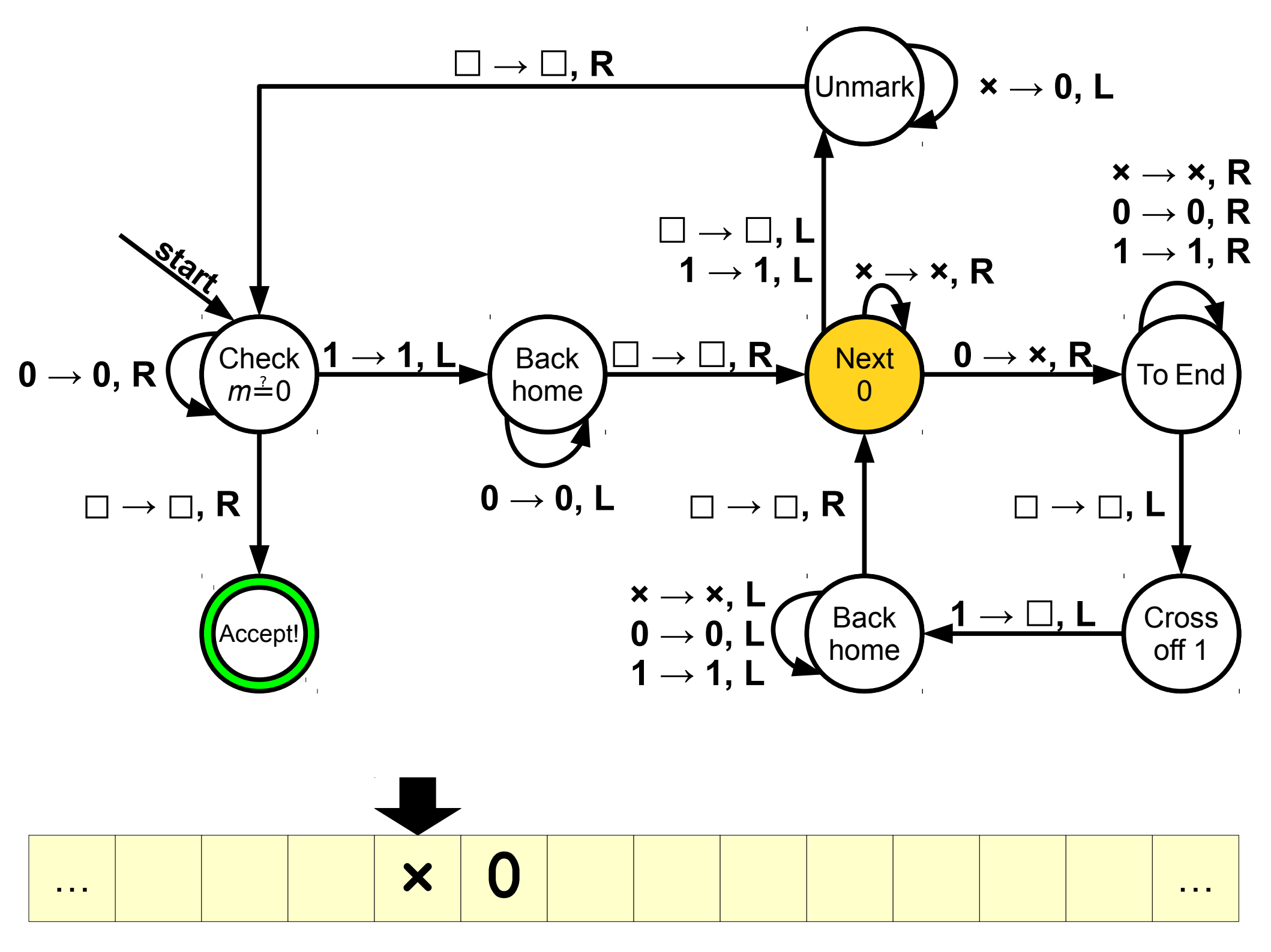

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
… × 0 1 …

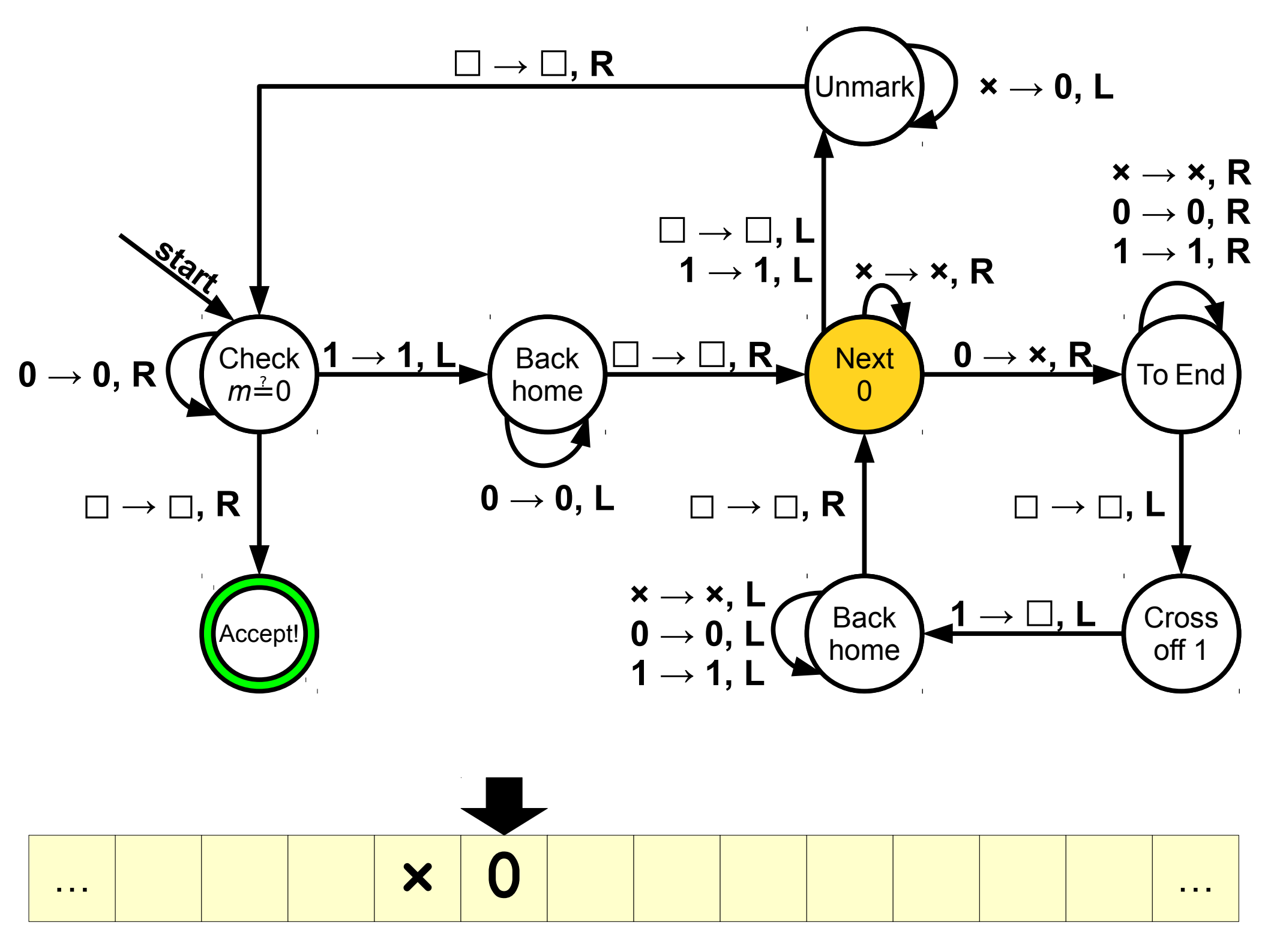

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
1
…

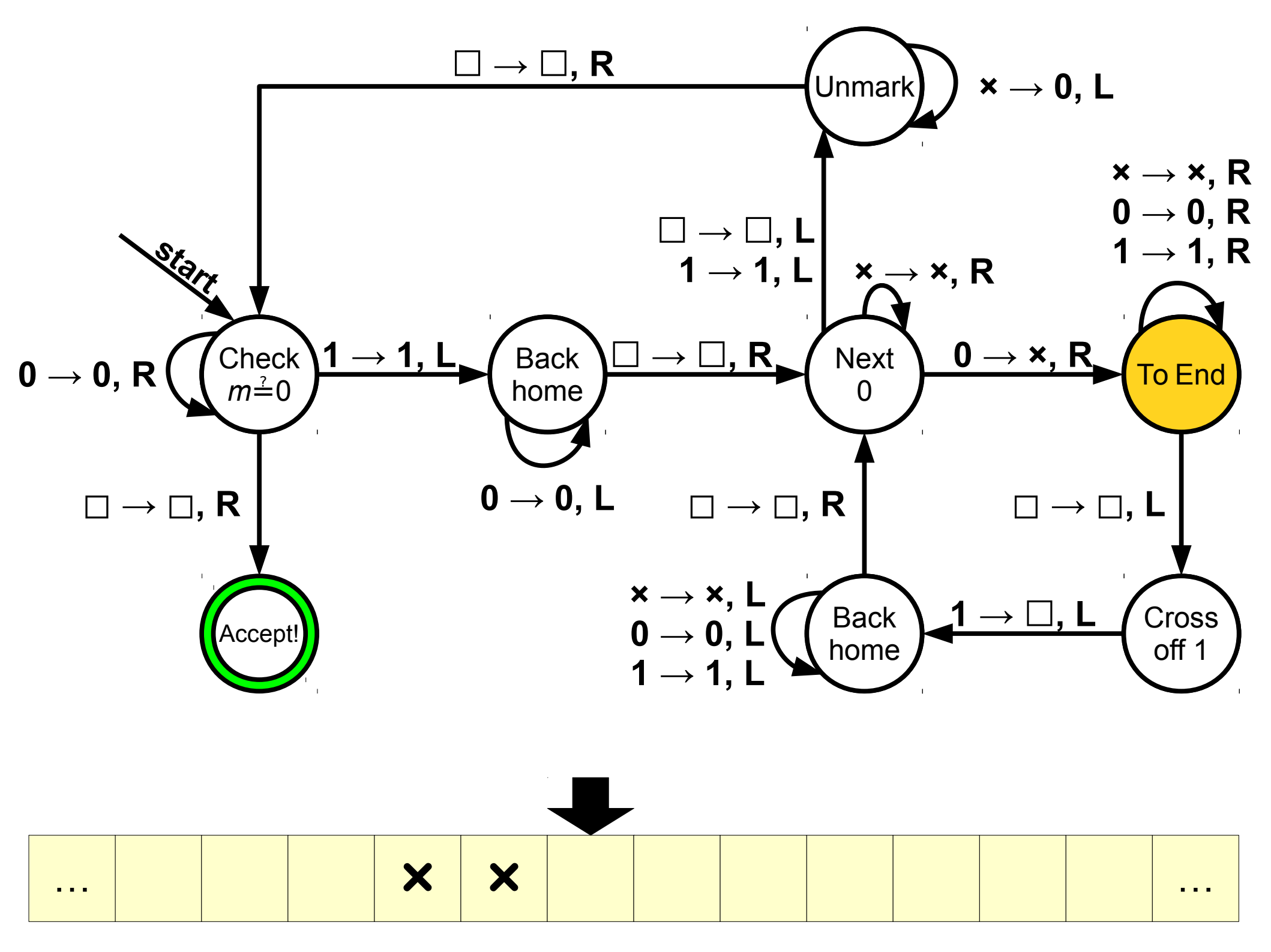

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
…

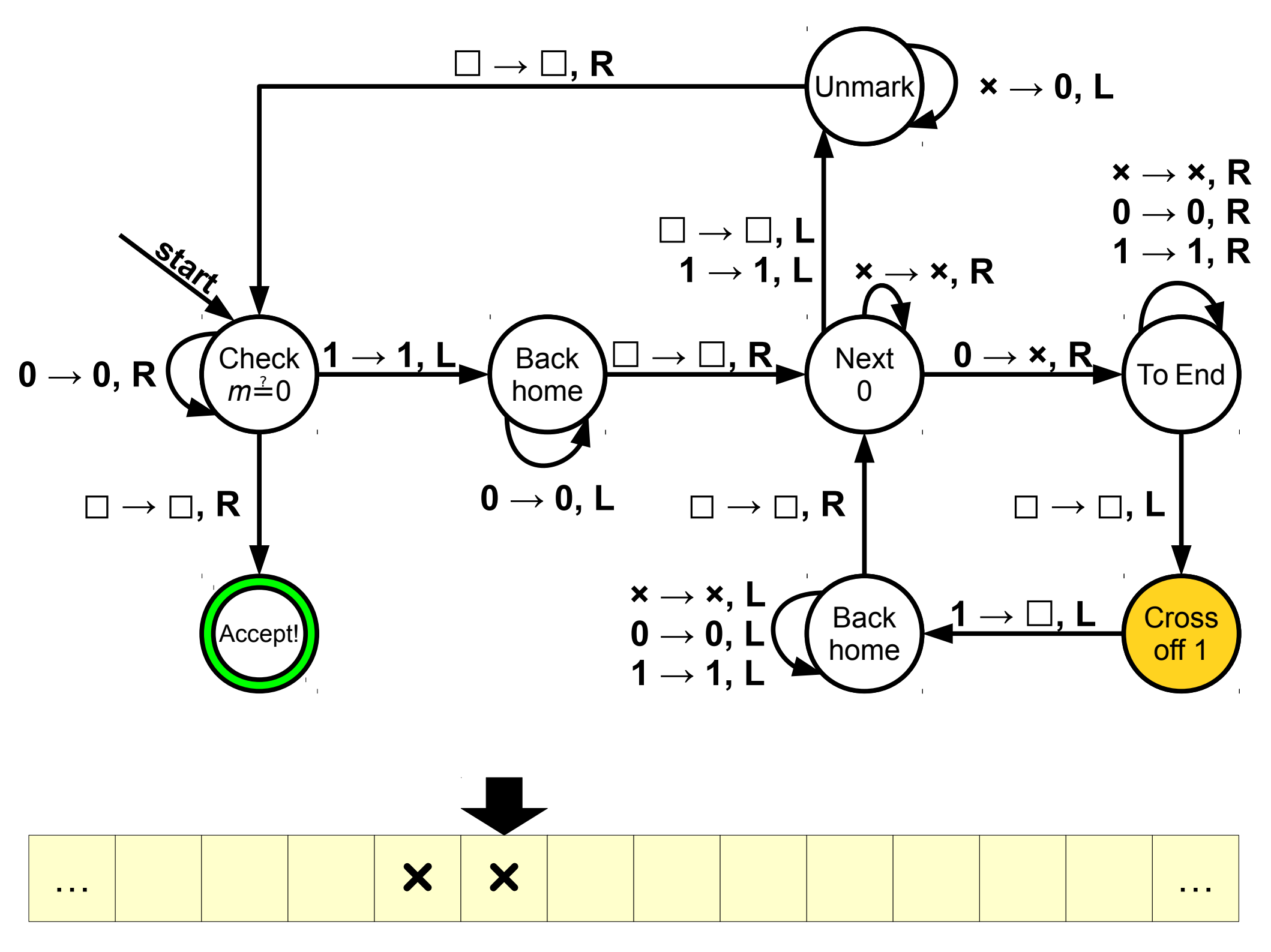

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
…

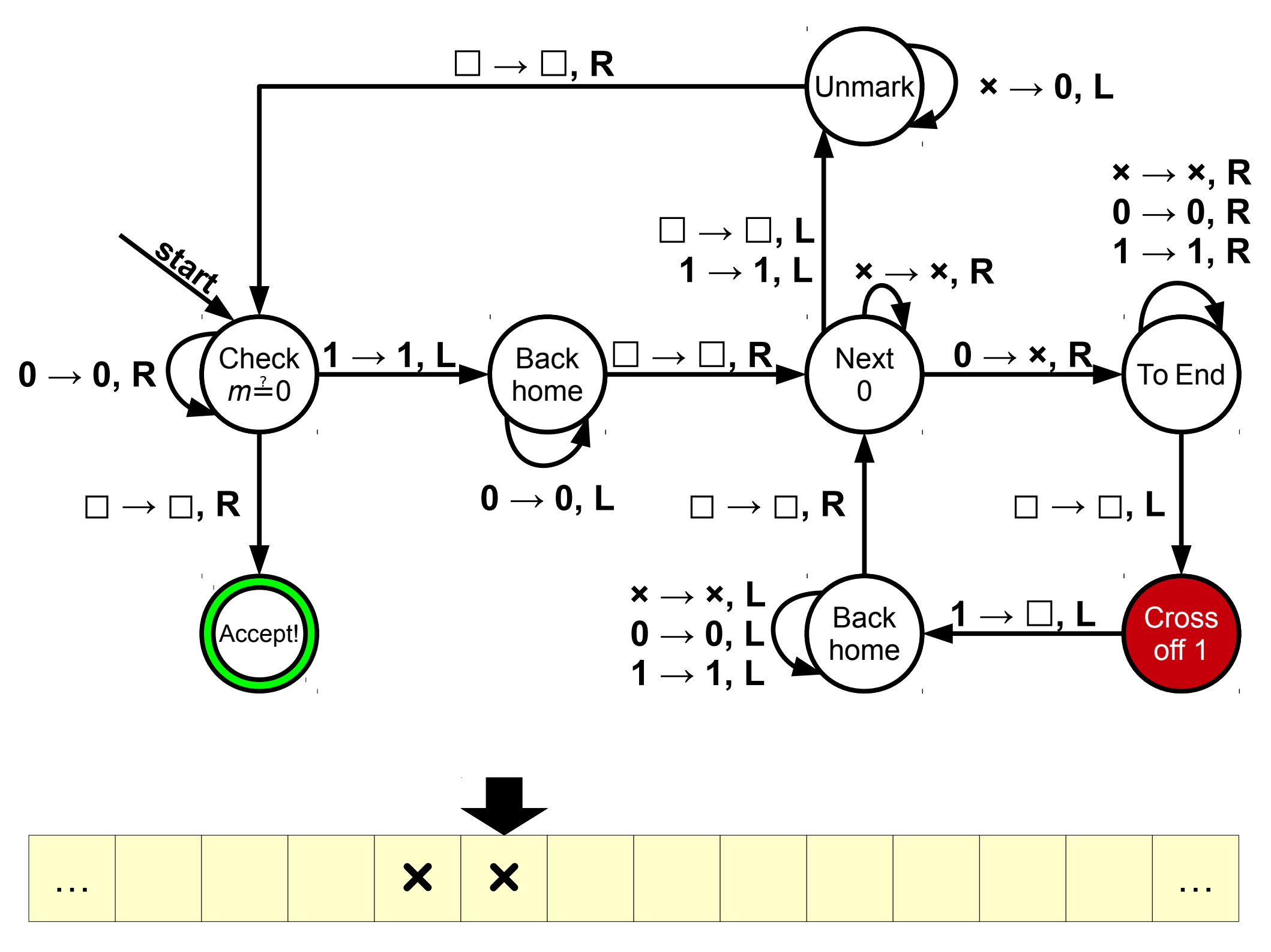

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
…

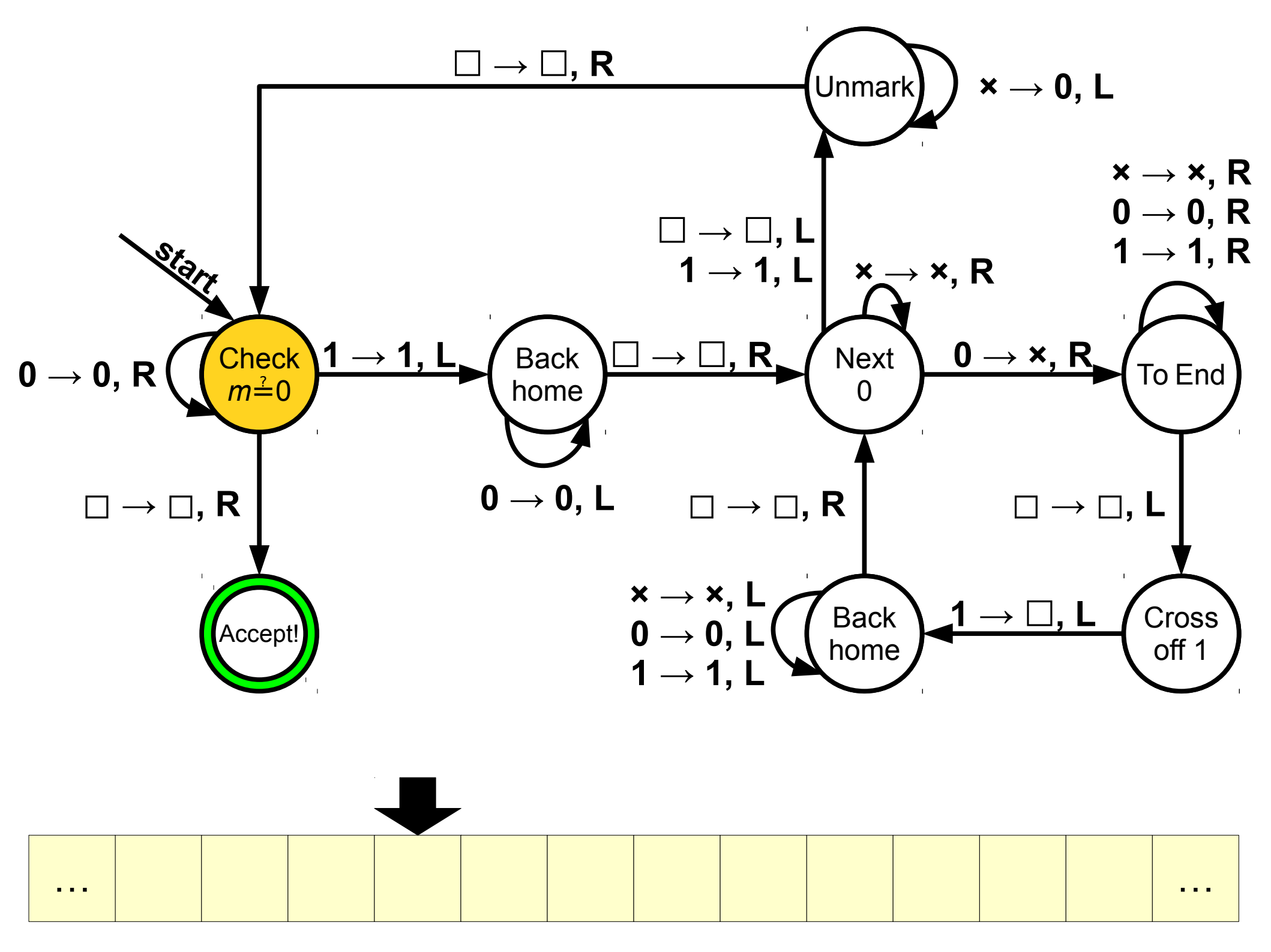

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
…

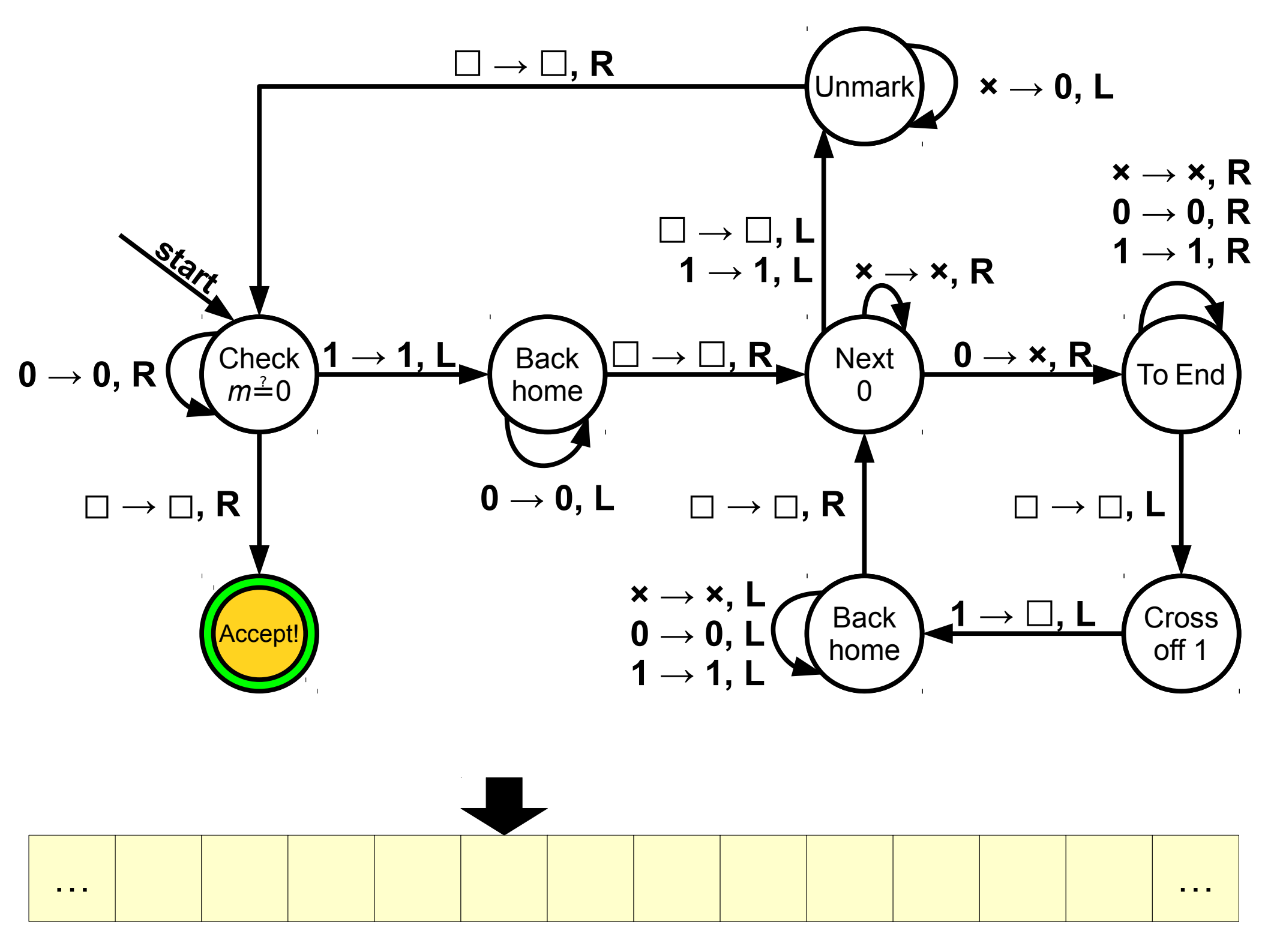

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
0
…

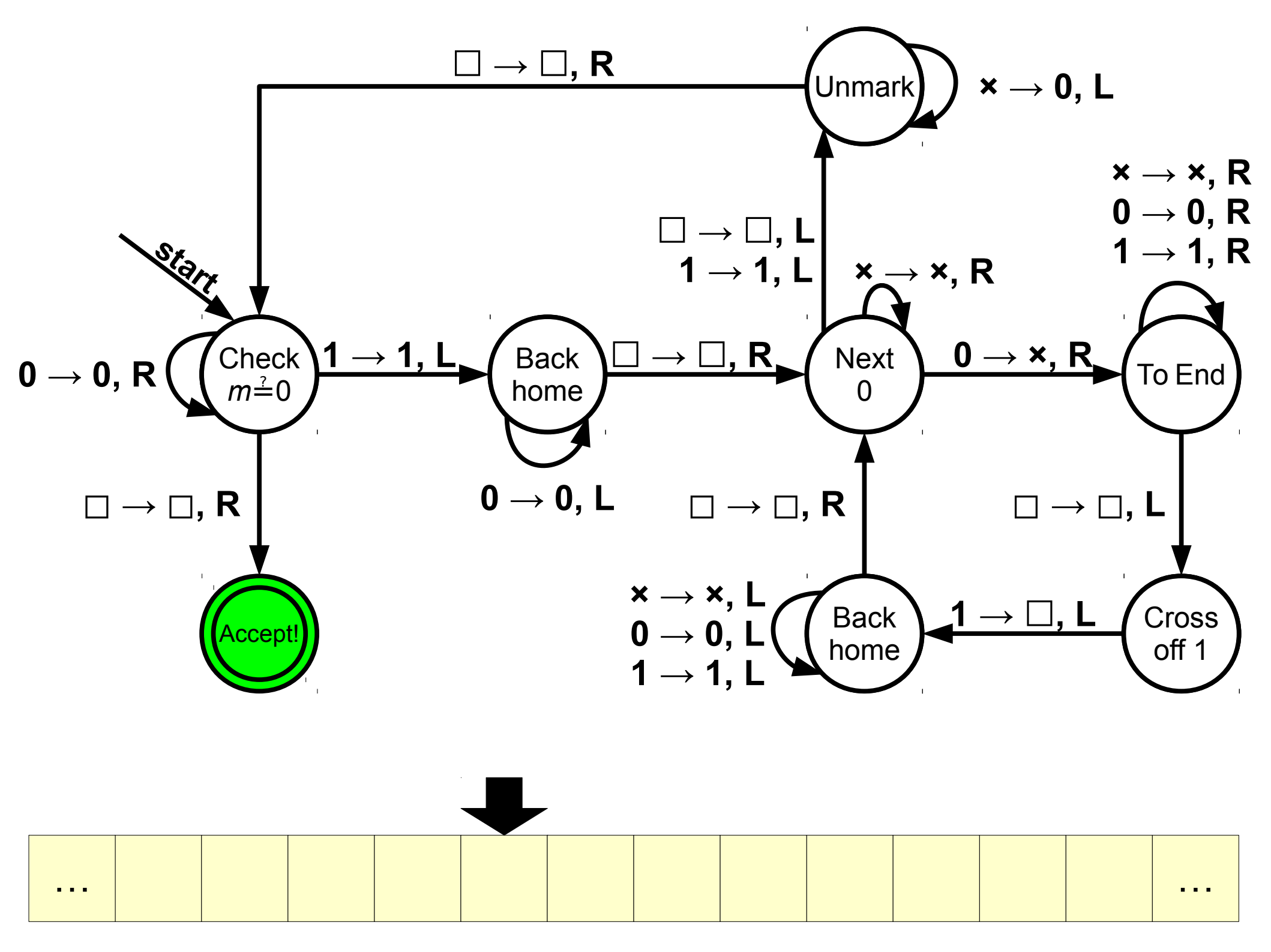

□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
×
×
…


□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
…


□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
…


□ → □, R
Unmark
× → 0, L
× → ×, R
0 → 0, R
1 → 1, R
□ → □, L
1 → 1, L
× → ×, R
start
Check
m ≟ 0
0 → 0, R
1 → 1, L
Back
home
□ → □, R
Next
0
0 → ×, R
To End
0 → 0, L
□ → □, R
□ → □, R
□ → □, L
Accept!
× → ×, L
0 → 0, L
1 → 1, L
Back
home
1 → □, L
Cross
off 1
…
…


start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Unmark
Back
home
Cross
off 1
Accept!
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
□ → □, L
1 → 1, L
× → ×, R
0 → ×, R
× → 0, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
× → ×, R
0 → 0, R
1 → 1, R
…
…


start
Check
$m \stackrel{?}{=} 0$
Back
home
Next
0
To End
Unmark
Back
home
Cross
off 1
Accept!
0 → 0, R
1 → 1, L
□ → □, R
0 → 0, L
□ → □, L
1 → 1, L
× → ×, R
0 → ×, R
× → 0, L
× → ×, L
0 → 0, L
1 → 1, L
1 → □, L
0 → 0, R
1 → 1, R
…
…