

In class, I alluded to some other things that Turing machines can do such as working with base 10 numbers and doing simple arithmetic operations like addition.

Here's a few more TM constructions to show how to go and do that! Going through these slides is completely optional, but we thought we'd include them in case you're curious :)

TM Arithmetic

- Let's design a TM that, given a tape that looks like this:

...				1	3	7		4	2			...
-----	--	--	--	---	---	---	--	---	---	--	--	-----

ends up having the tape look like this:

...				1	7	9		0	0			...
-----	--	--	--	---	---	---	--	---	---	--	--	-----

- In other words, we want to build a TM that can add two numbers.

TM Arithmetic

- There are many ways we could in principle design this TM.
- We're going to take the following approach:
 - First, we'll build a TM that increments a number.
 - Next, we'll build a TM that decrements a number.
 - Then, we'll combine them together, repeatedly decrementing the second number and adding one to the first number.

TM Arithmetic

```
add(num1, num2) {  
    while (num2 > 0) {  
        increment(num1);  
        decrement(num2);  
    }  
}
```

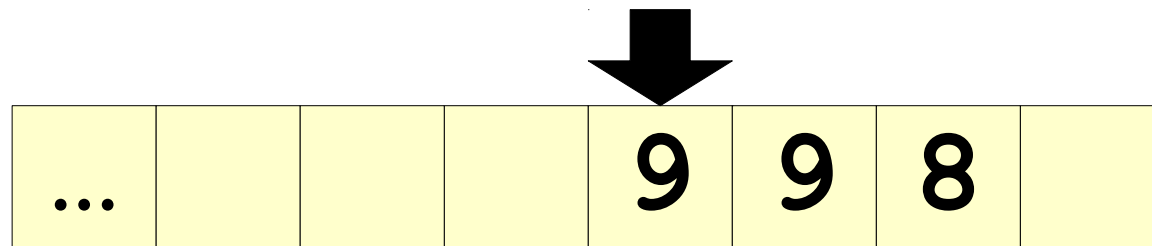
TM Arithmetic

```
add(num1, num2) {  
    while (num2 > 0) {  
        increment(num1);  
        decrement(num2);  
    }  
}
```

Let's write this helper method first!

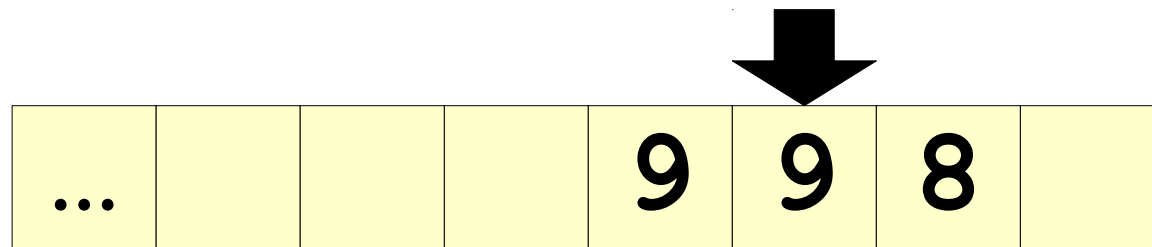
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



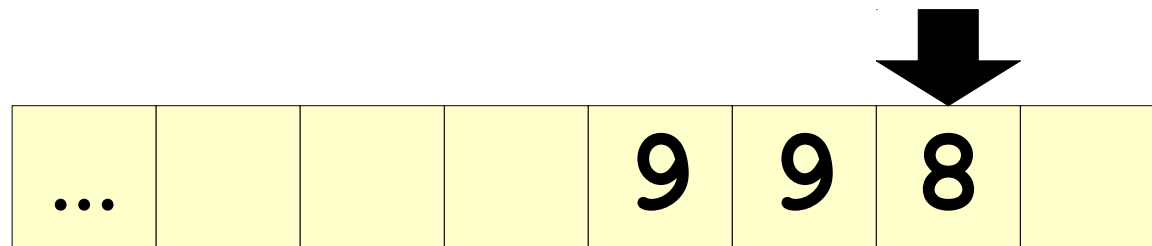
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



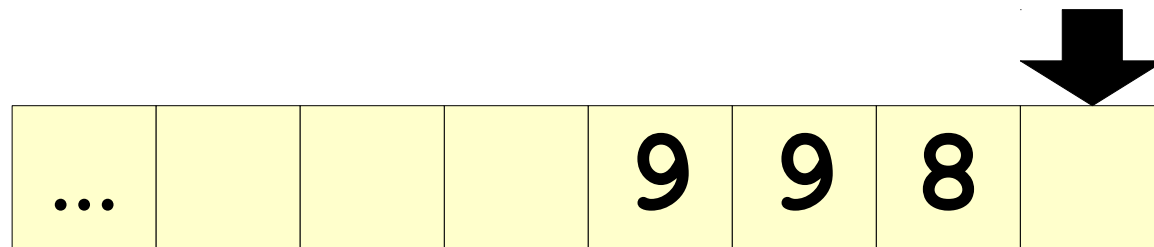
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



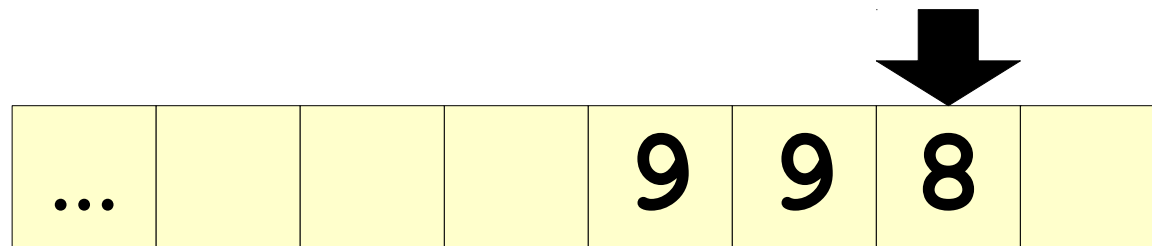
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



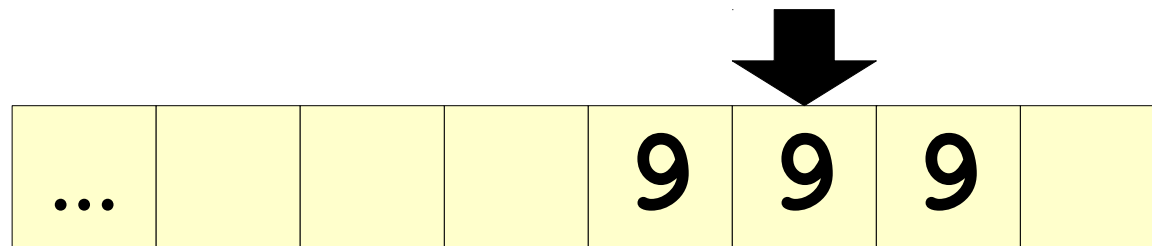
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



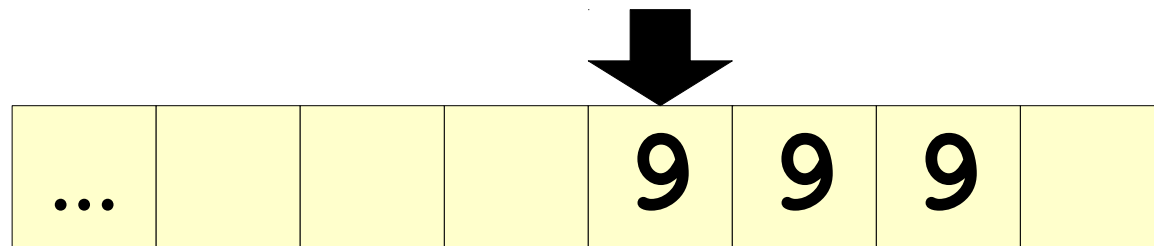
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



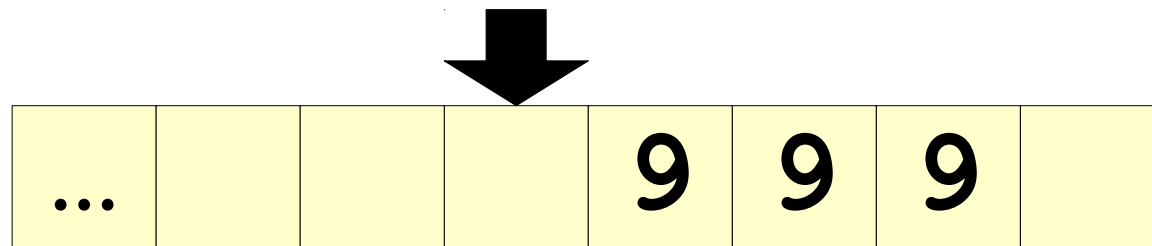
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



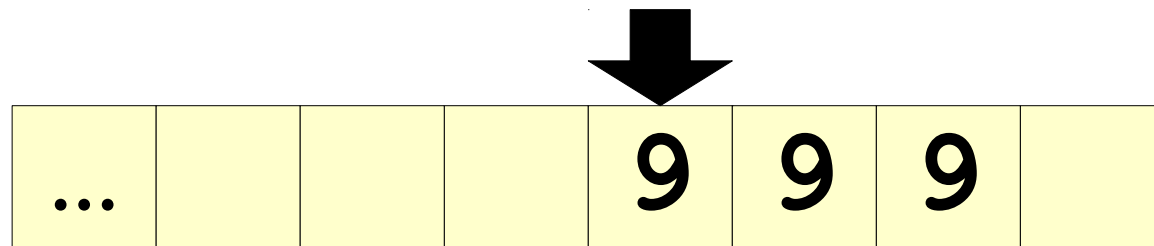
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



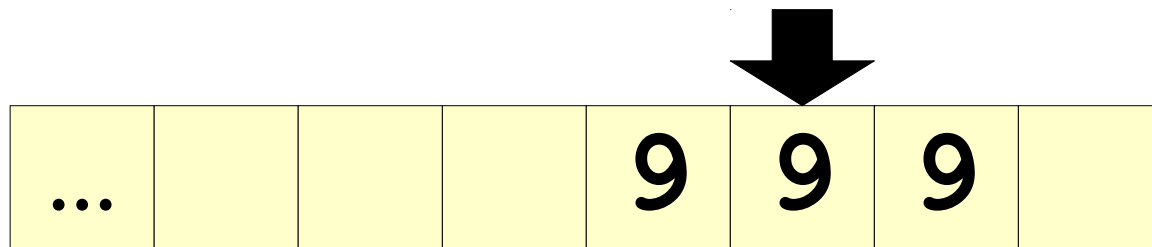
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



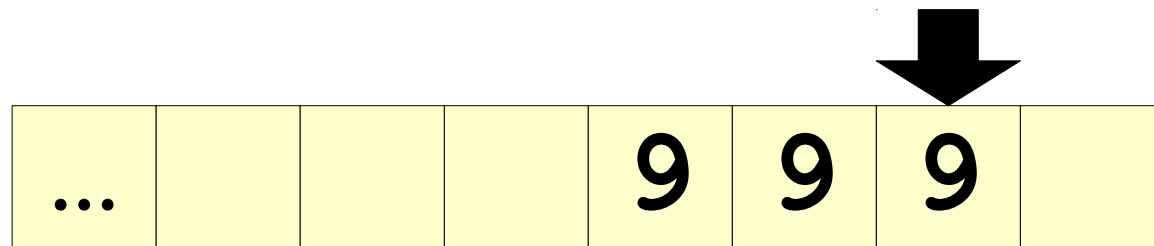
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



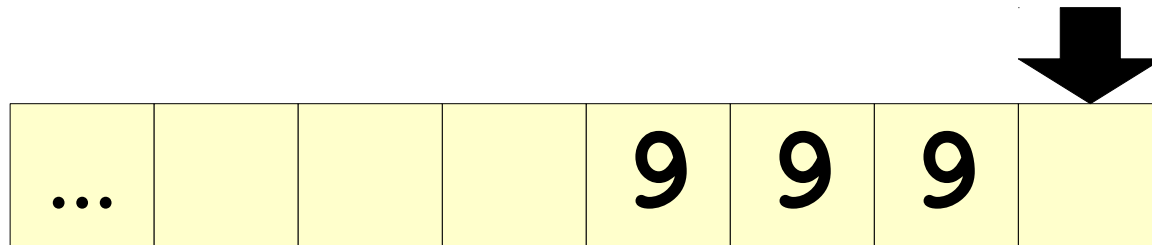
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



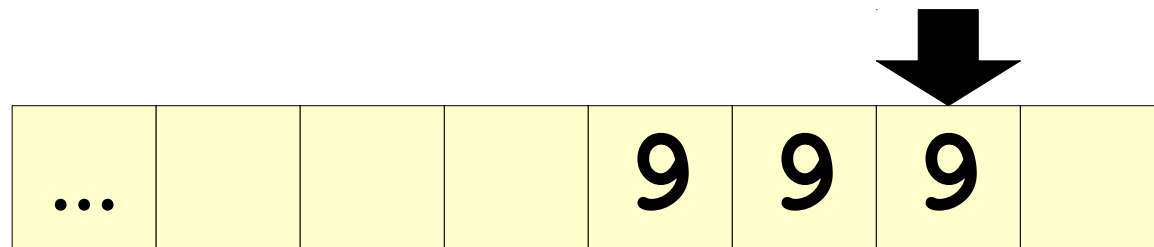
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



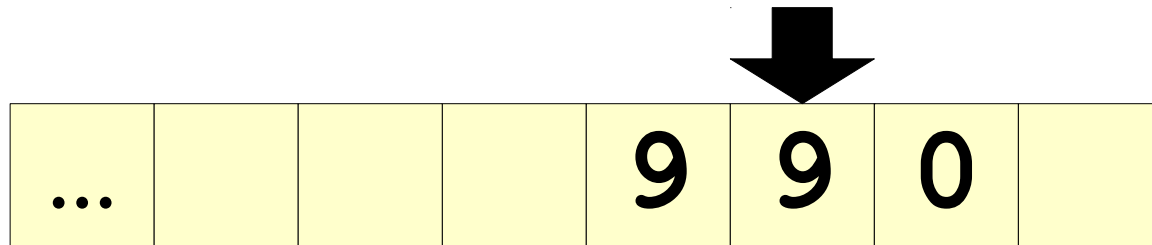
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



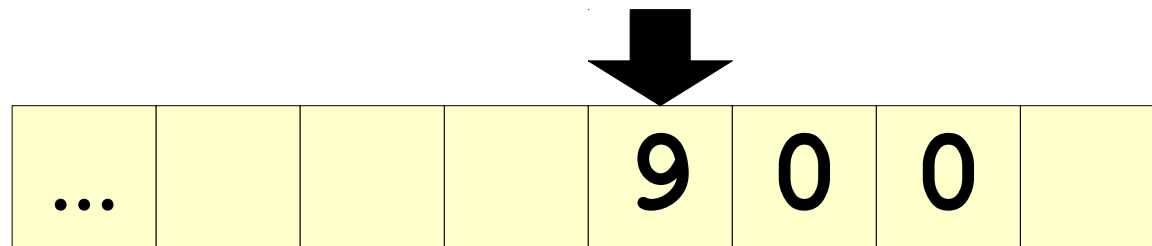
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



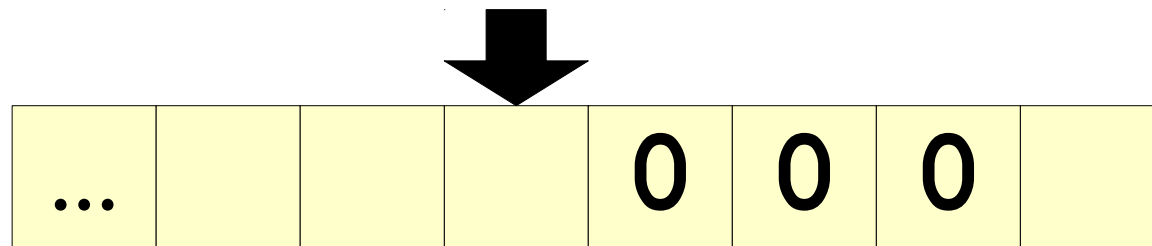
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



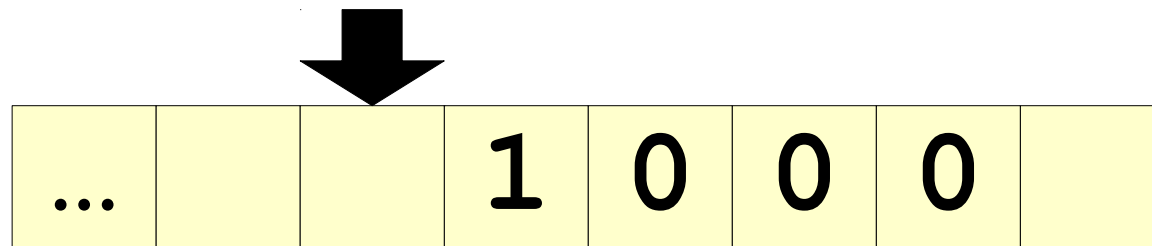
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



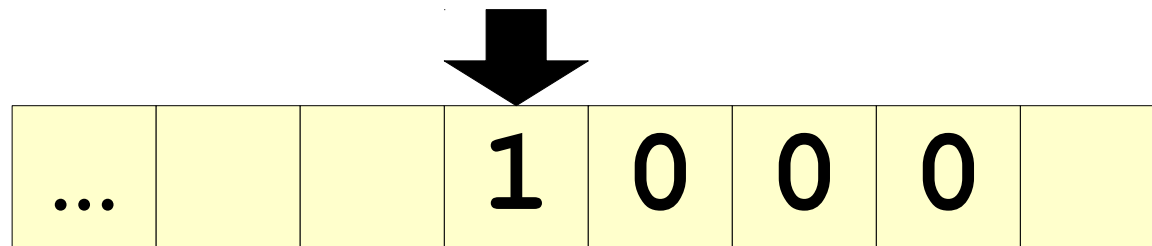
Incrementing Numbers

- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.



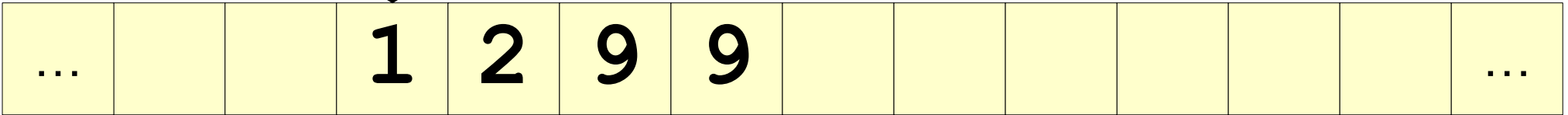
Incrementing Numbers

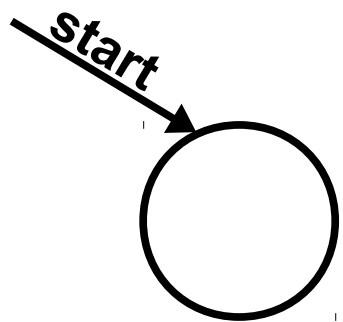
- Let's begin by building a TM that increments a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is are at least two blanks to the left of the number, and
 - that there's at least one blank at the end of the number.
- The tape head will end at the start of the number after incrementing it.

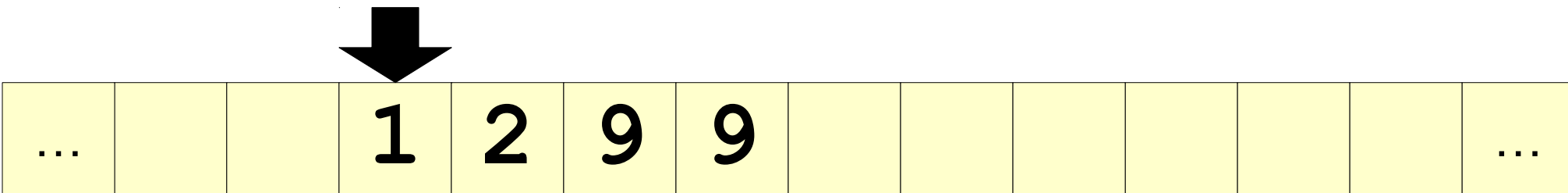
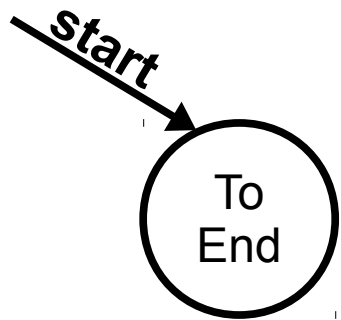


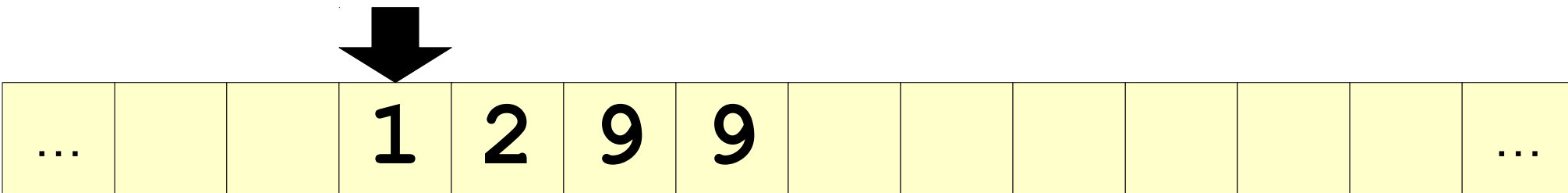
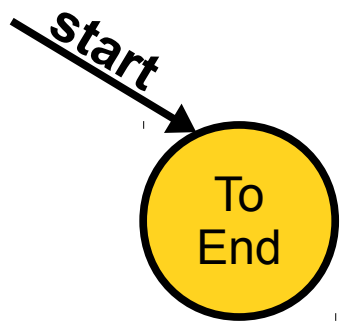
Incrementing Numbers

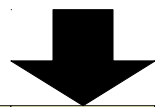
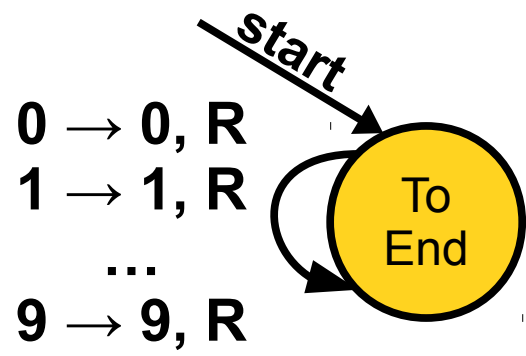
```
increment(num) {  
    go to the end of the number;  
    while (the current digit is 9) {  
        set the current digit to 0;  
        back up one digit;  
    }  
    increment the current digit;  
    go to the start of the number;  
}
```

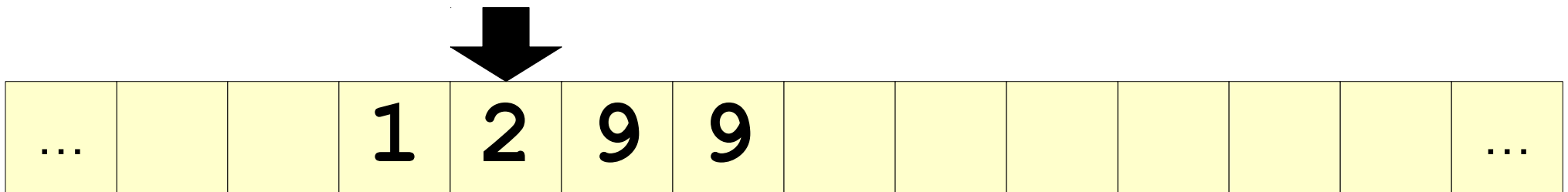
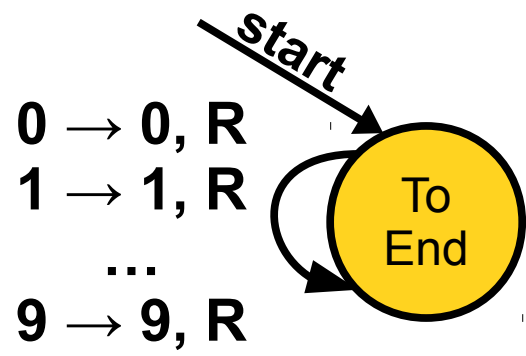



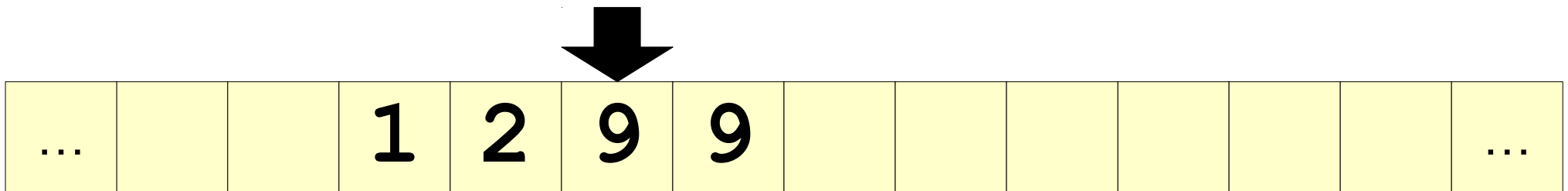
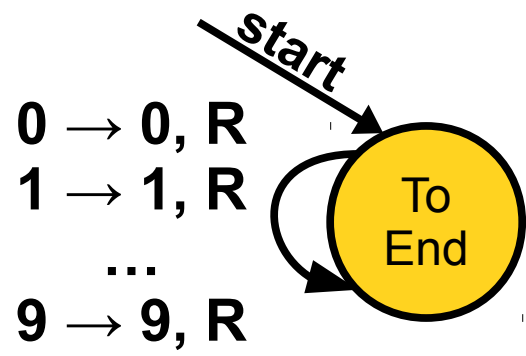


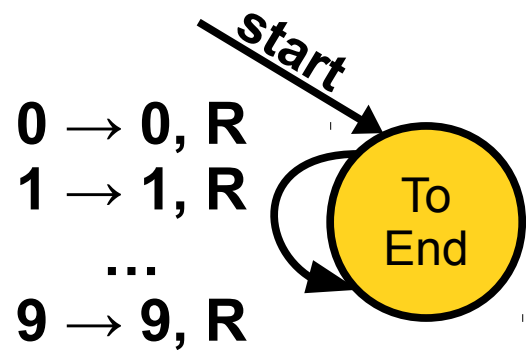


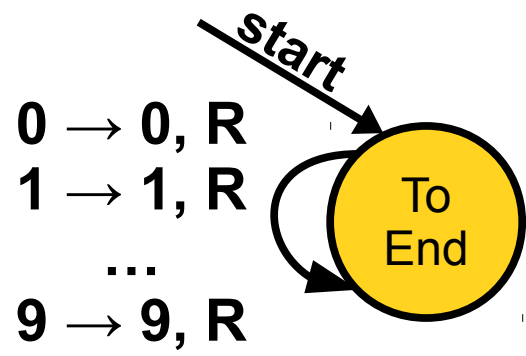


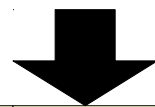
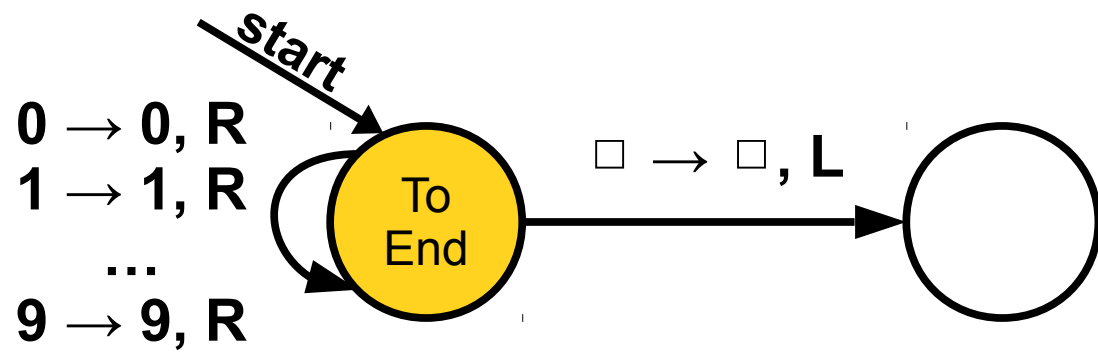


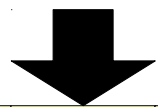
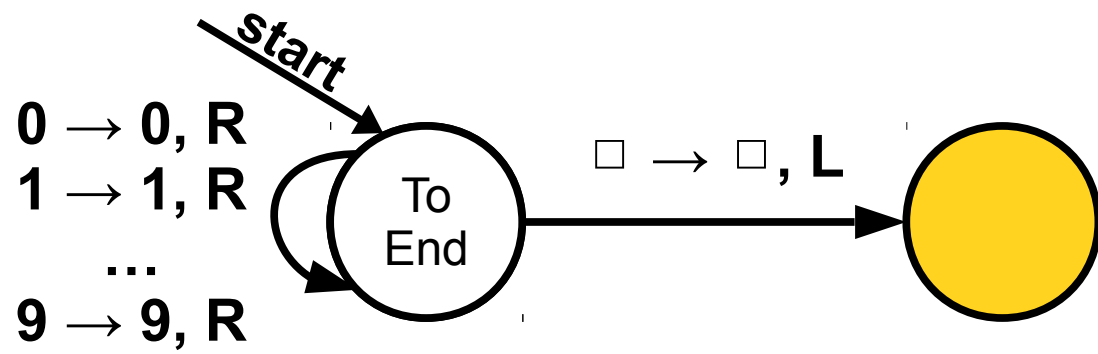


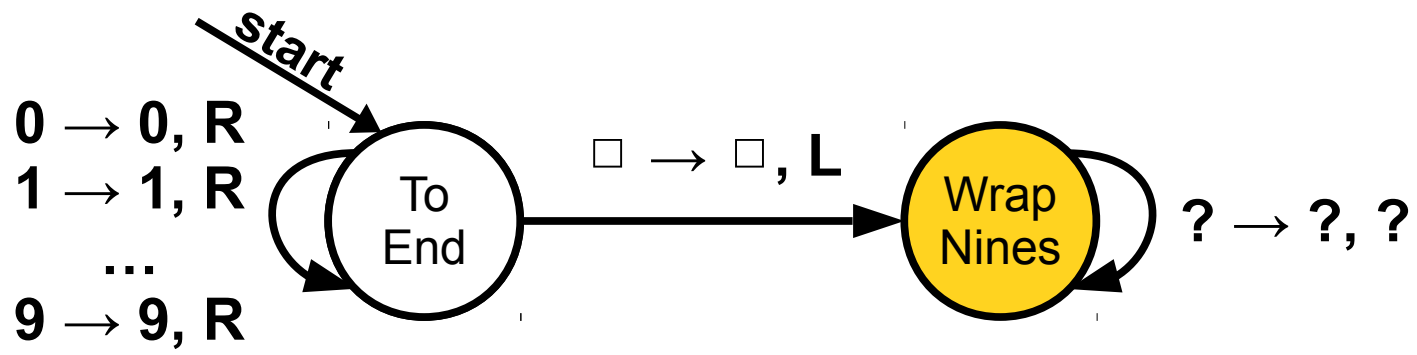


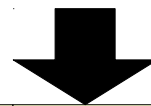
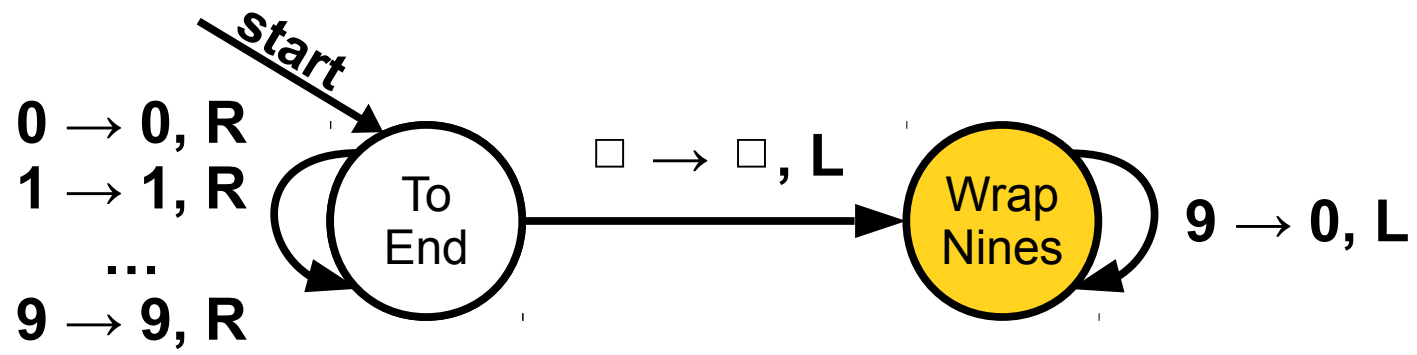


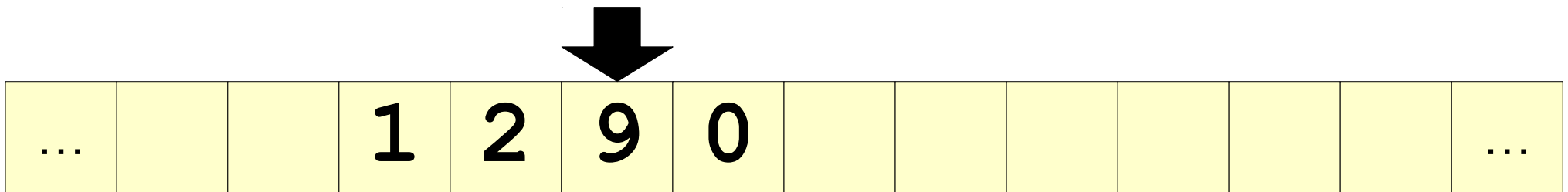
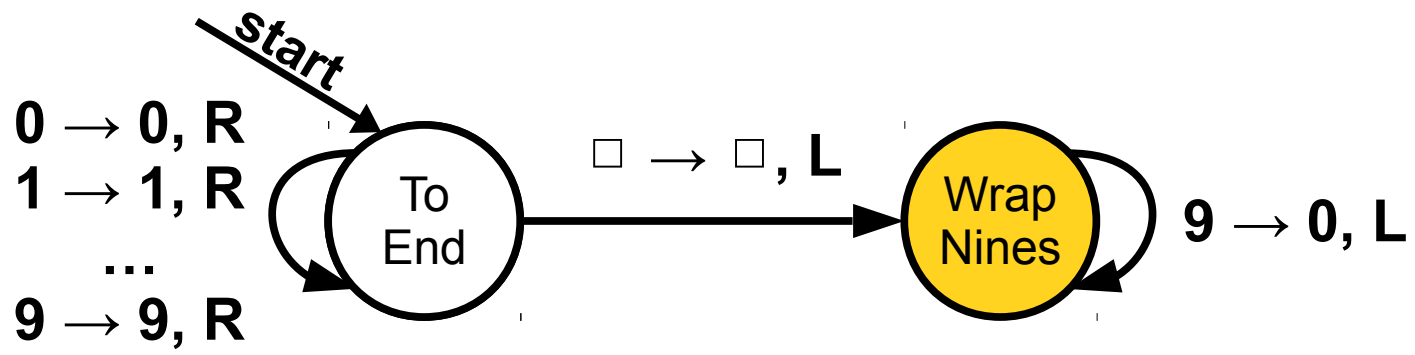


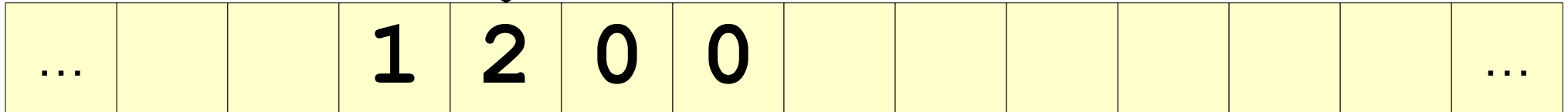
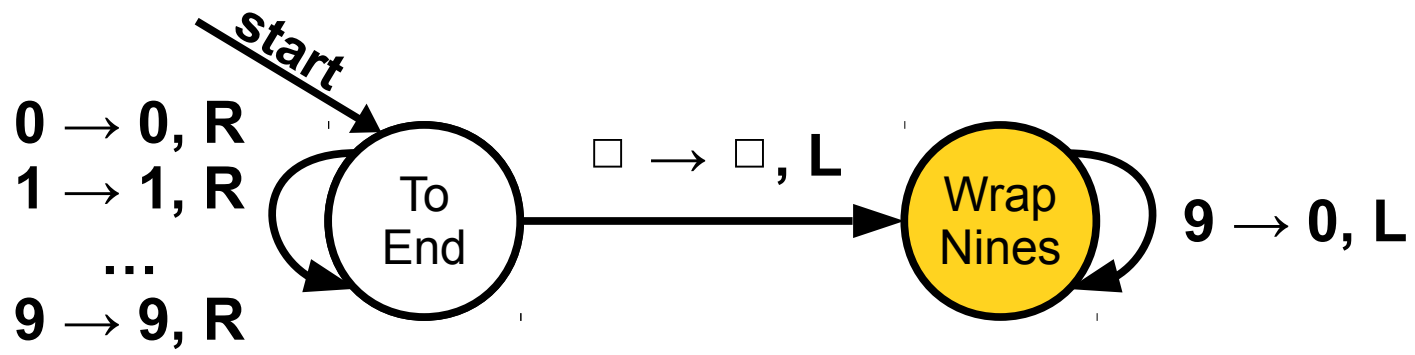


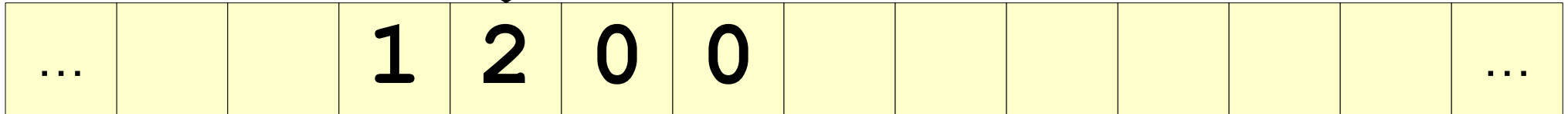
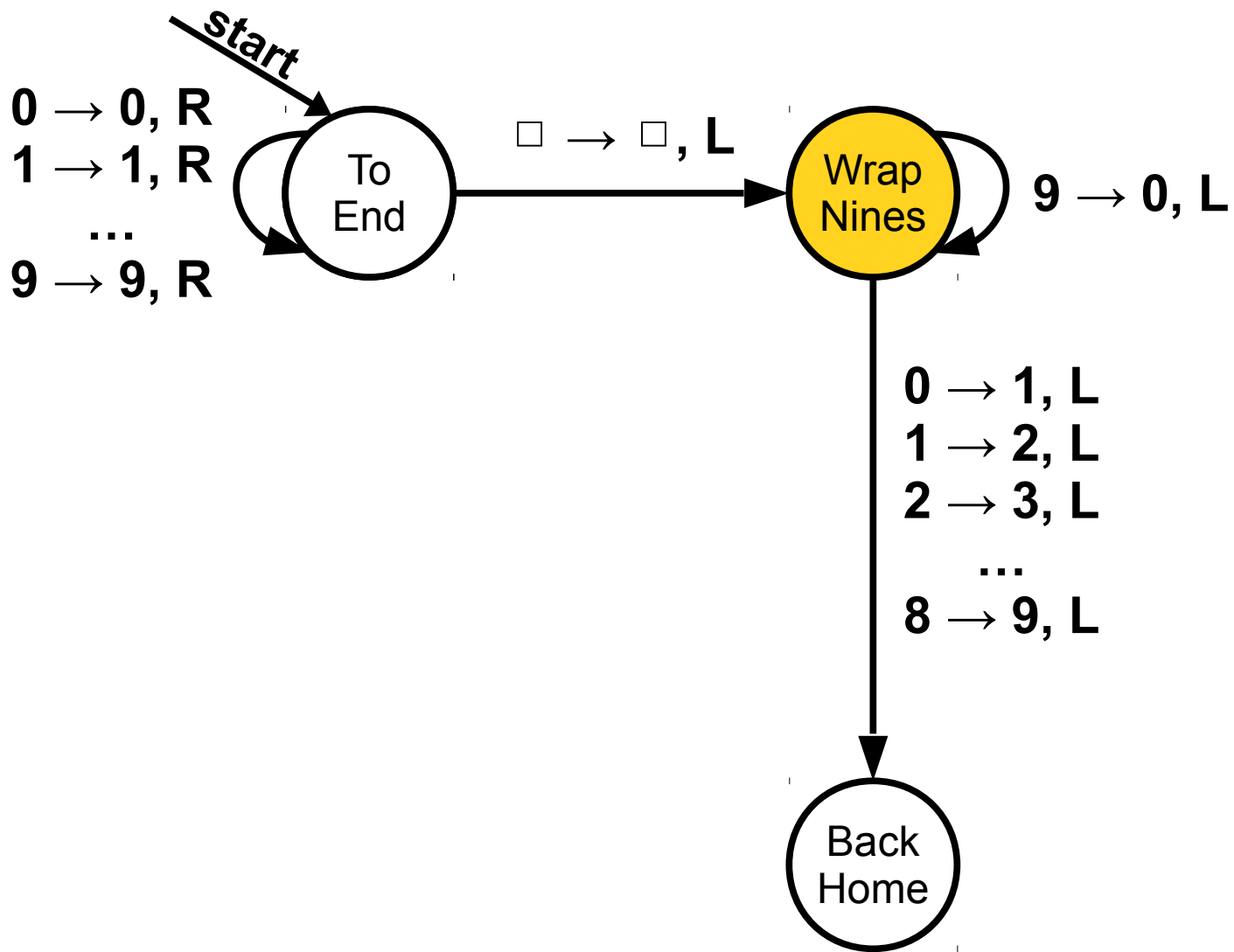


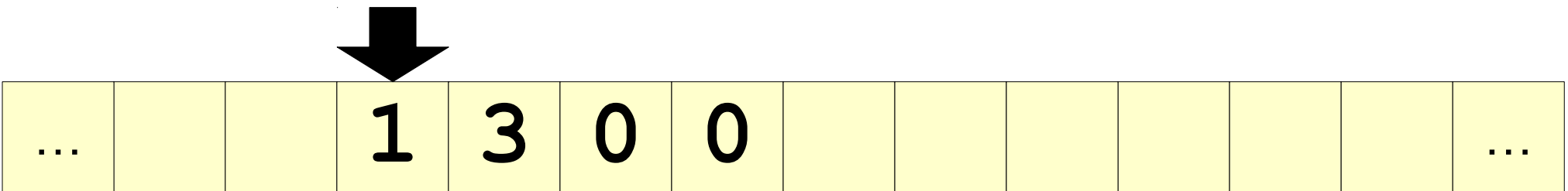
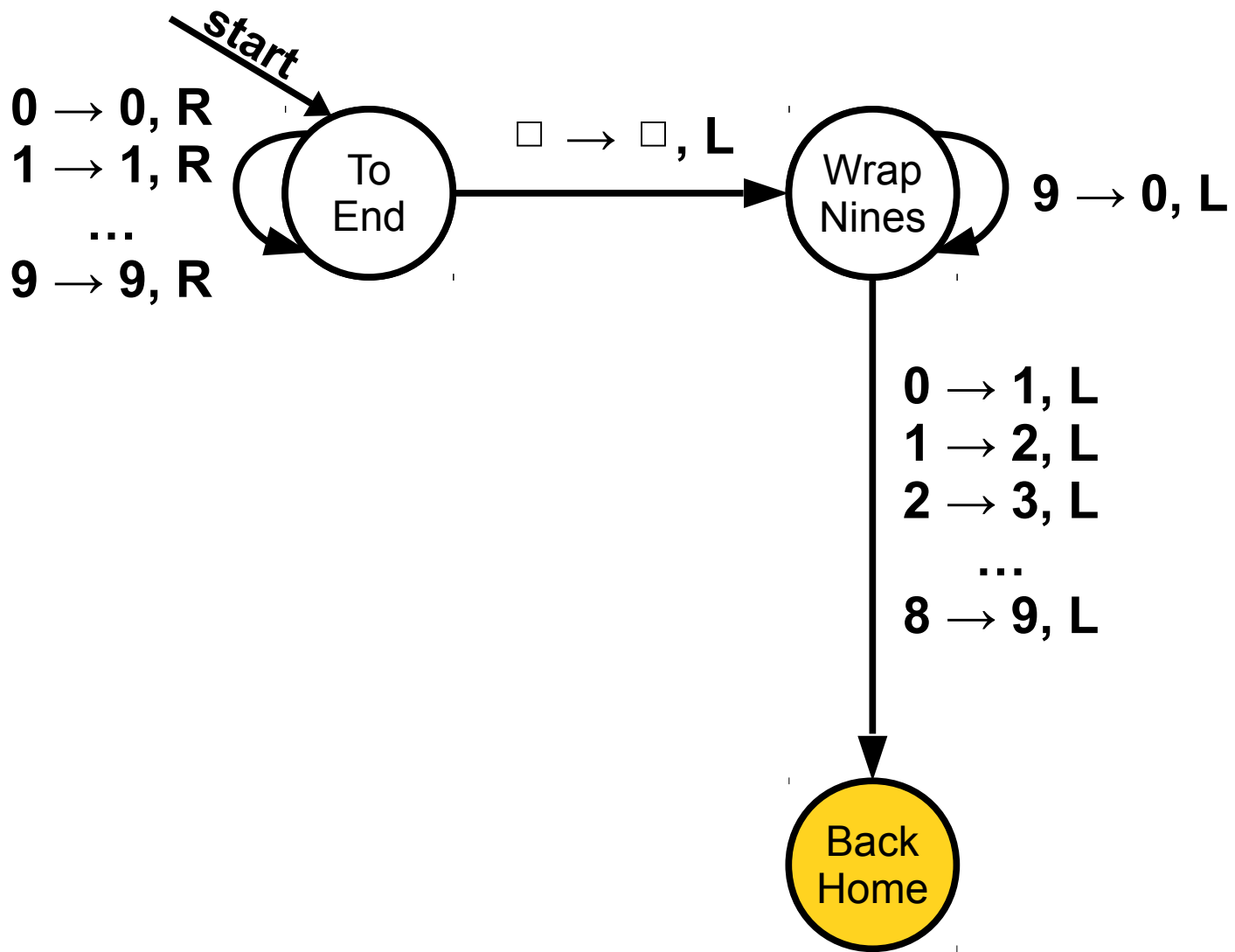


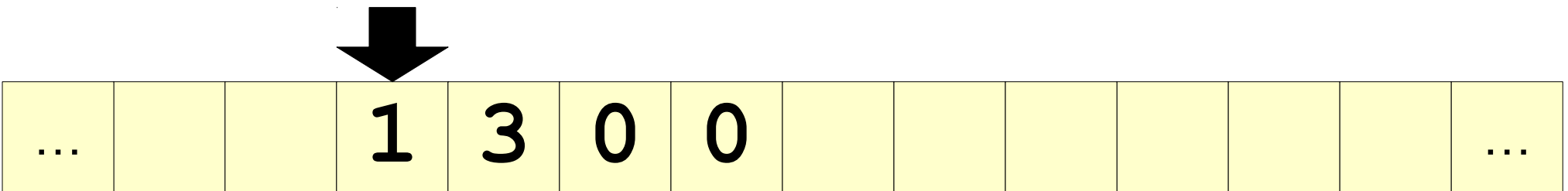
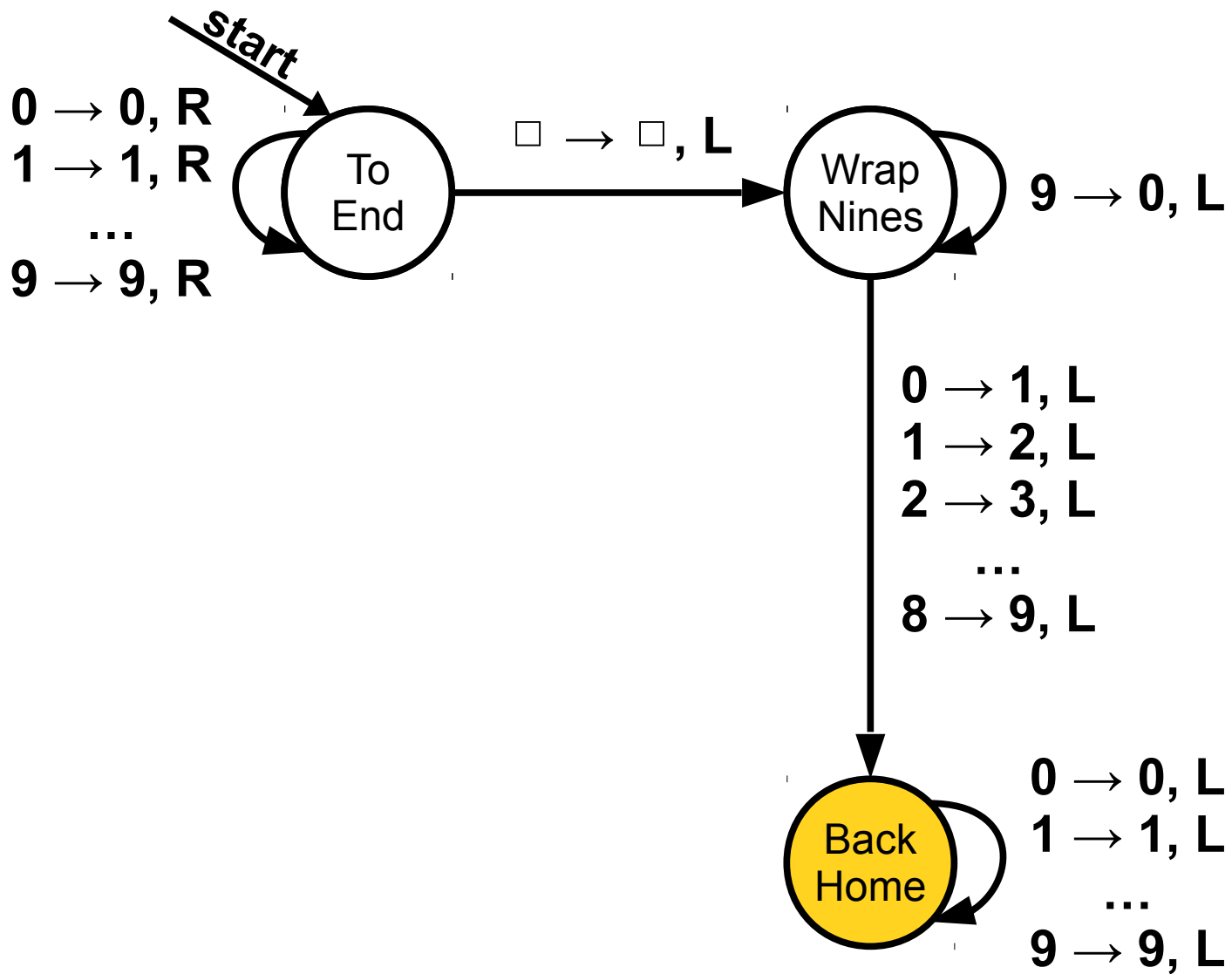


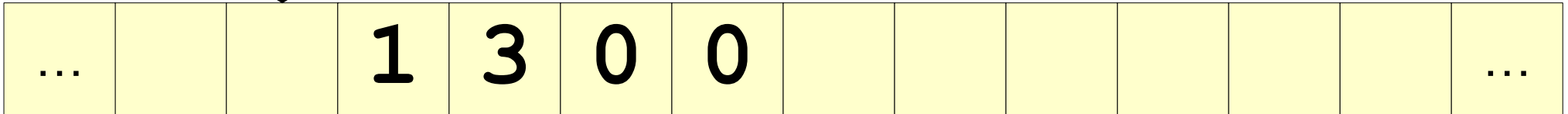
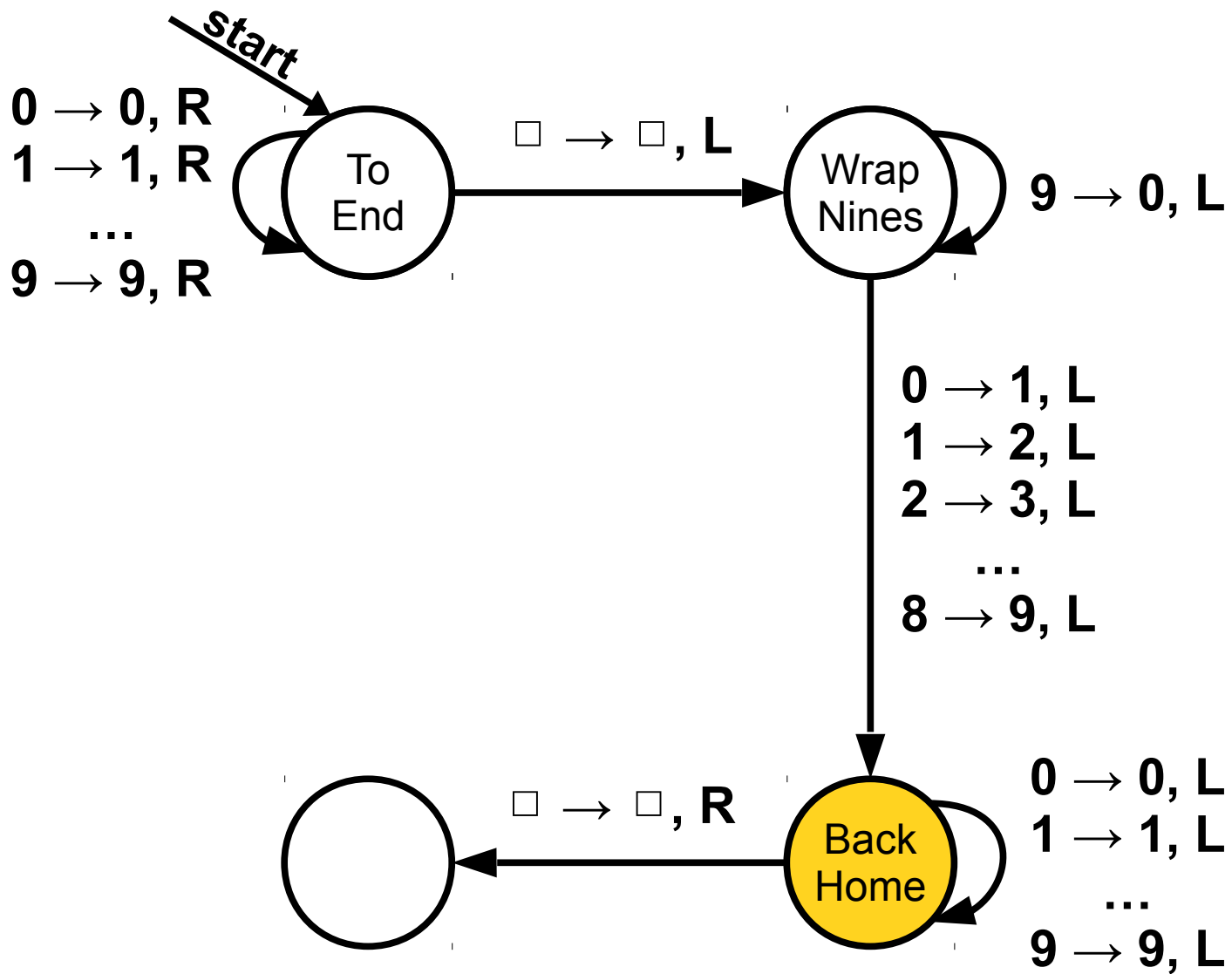


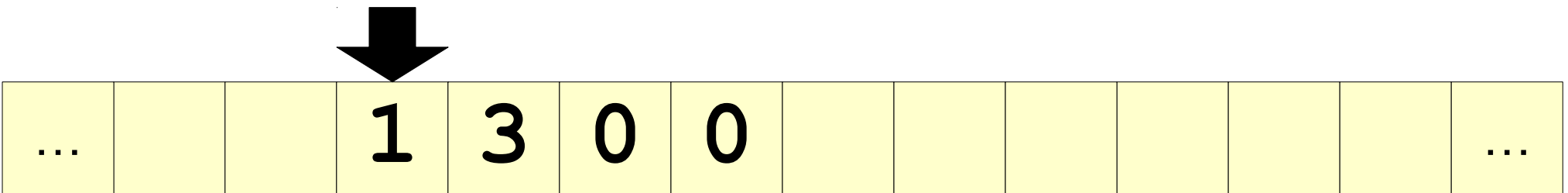
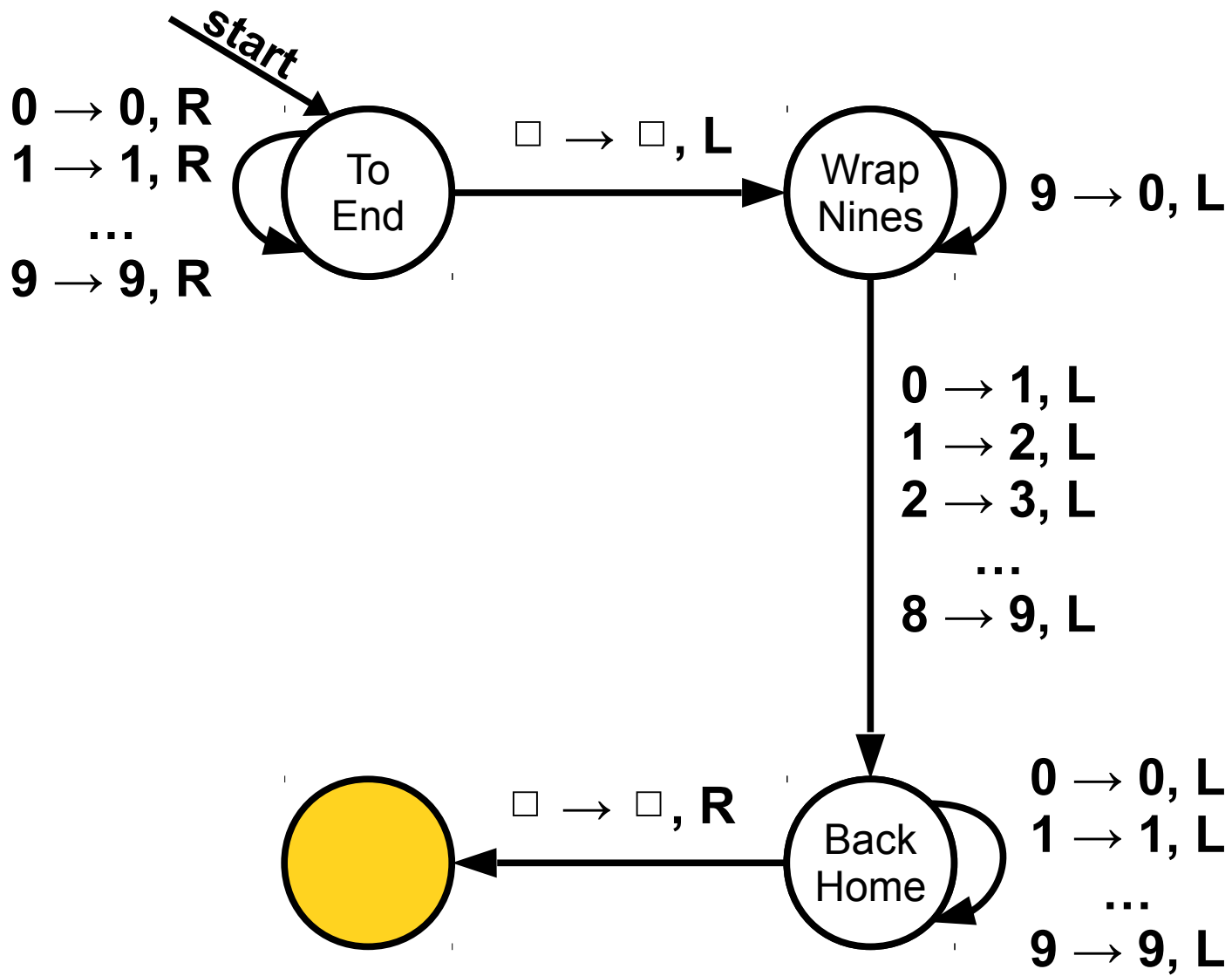


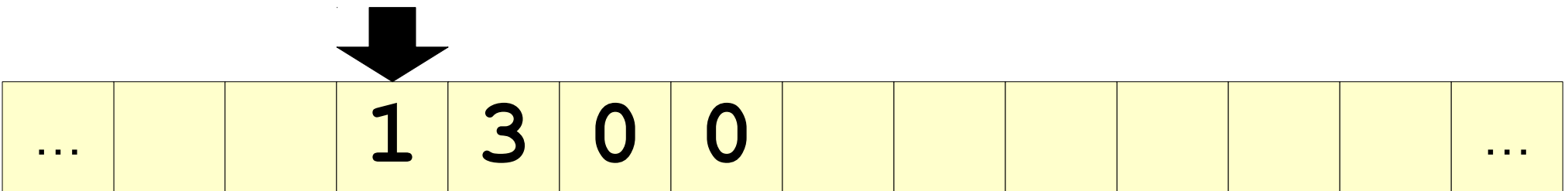
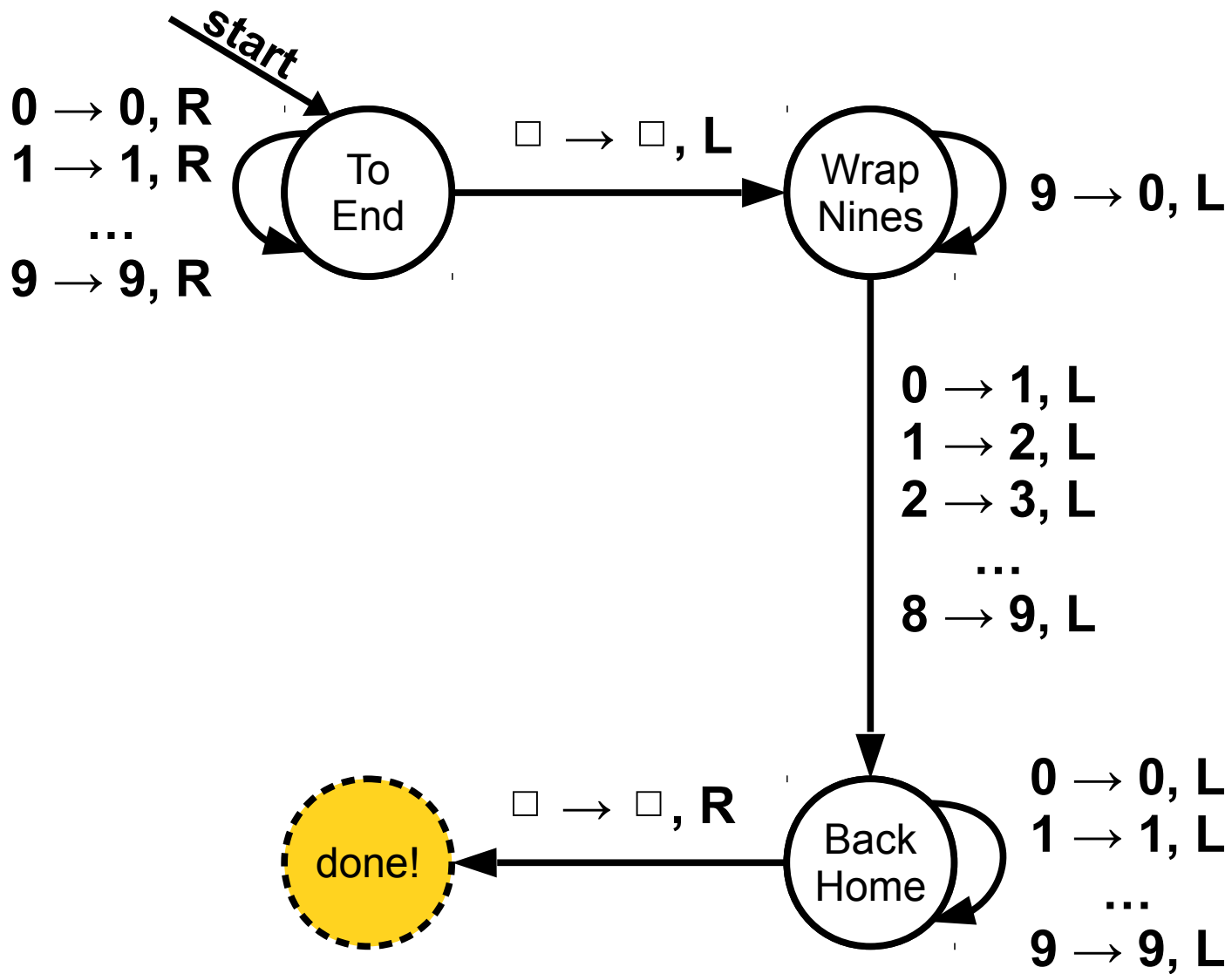


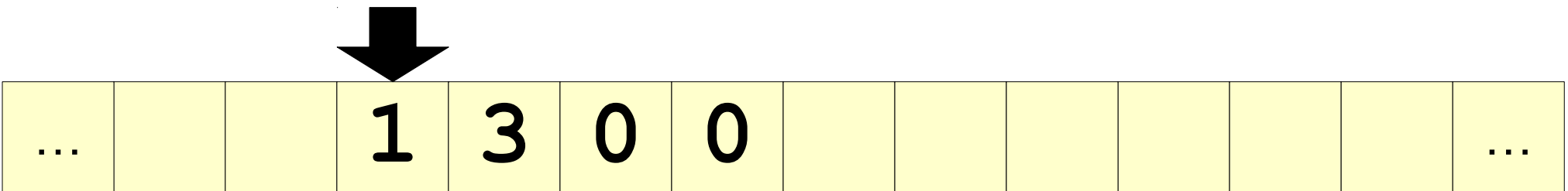
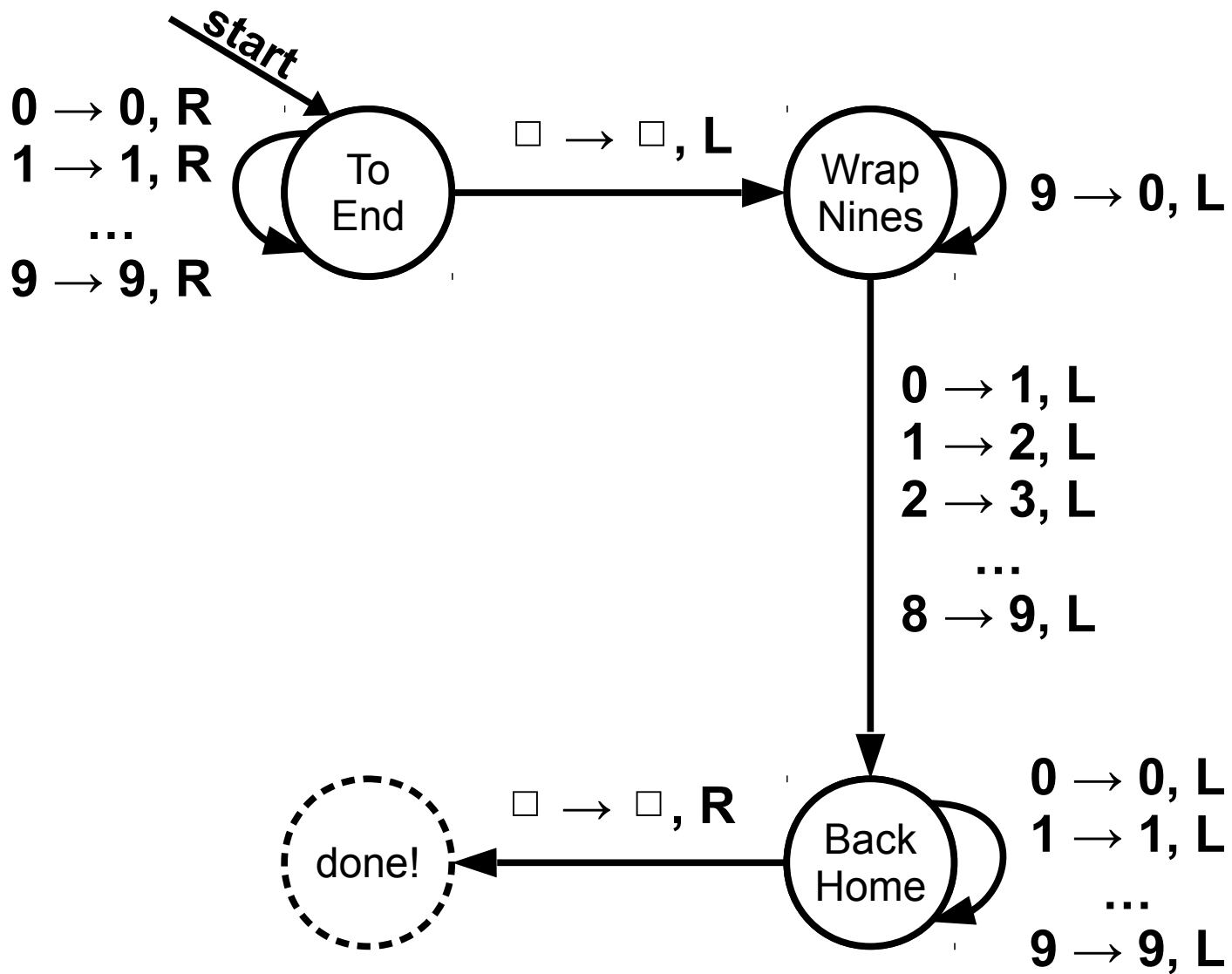


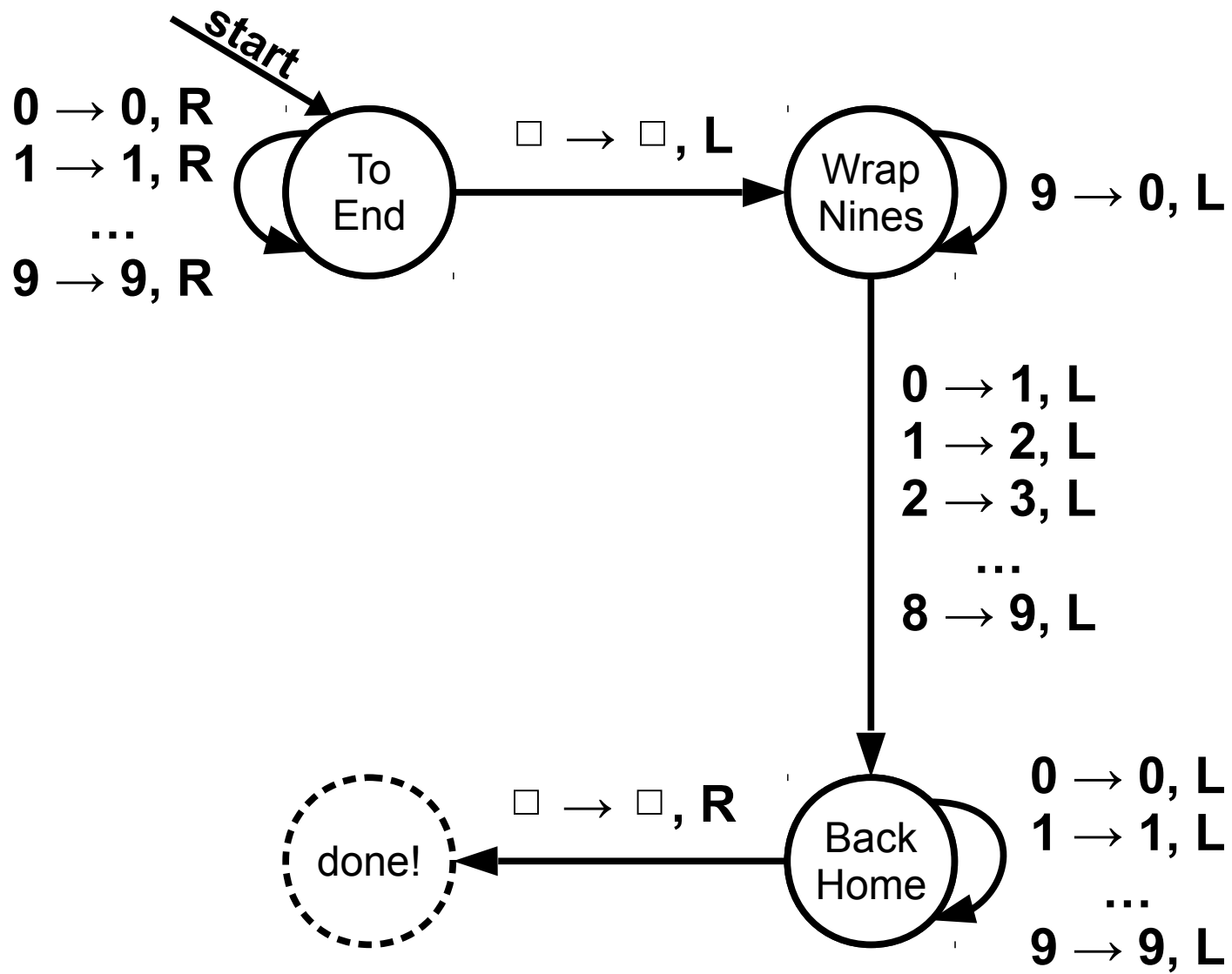


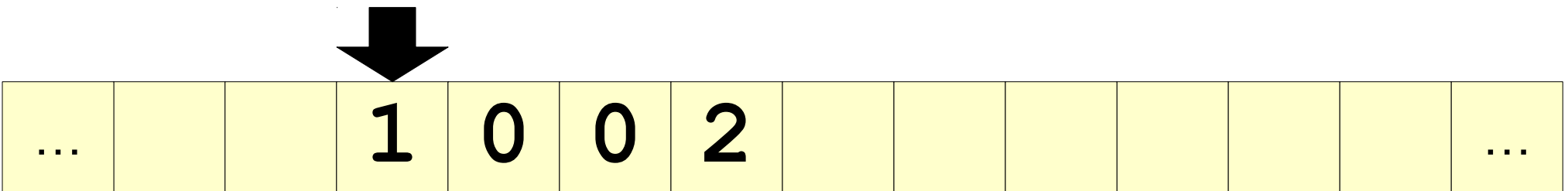
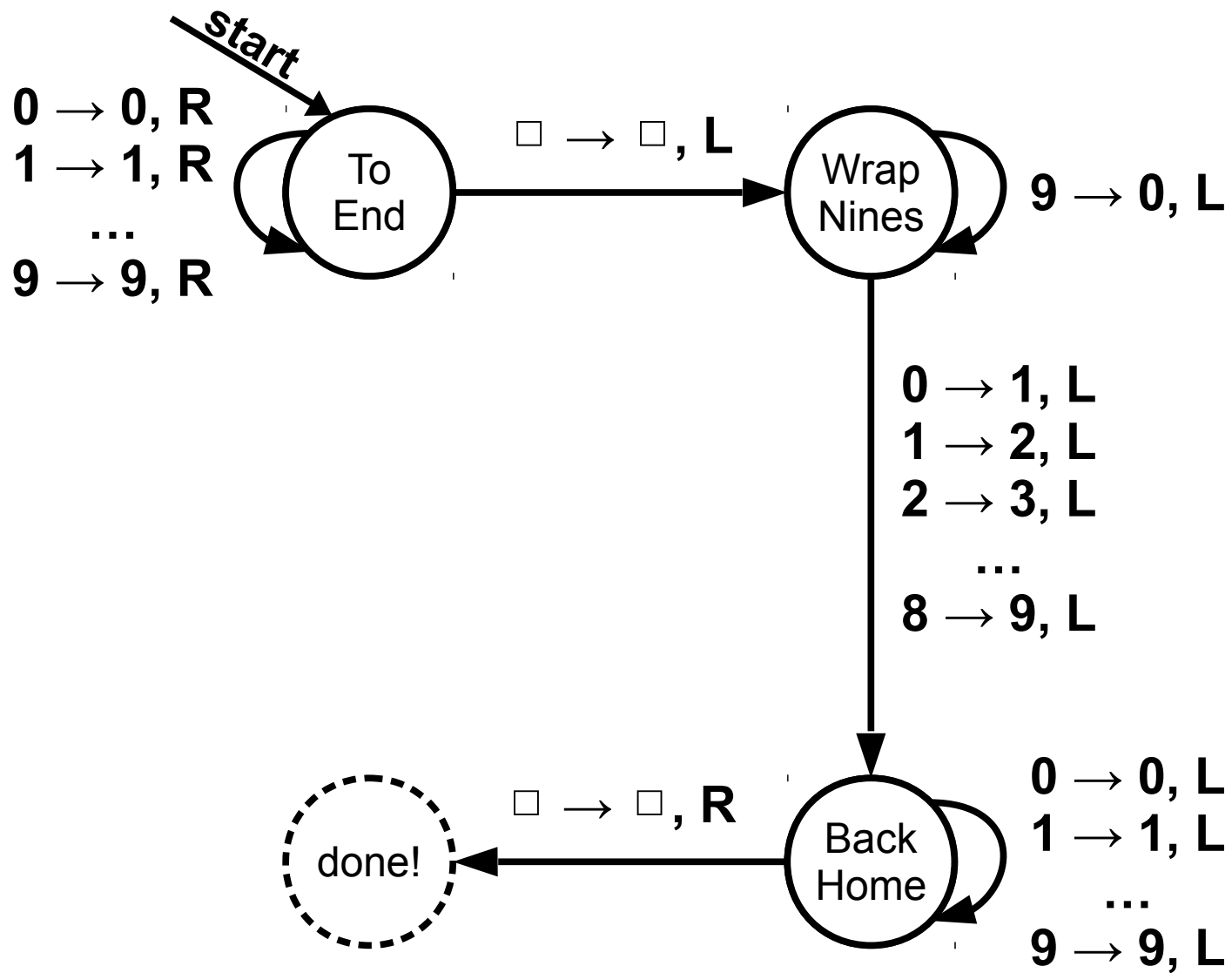


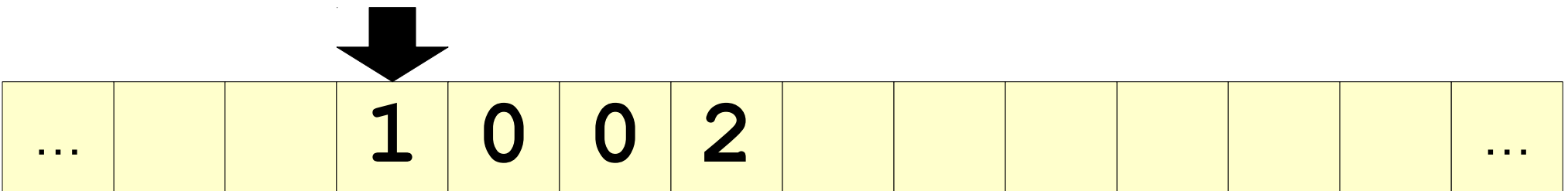
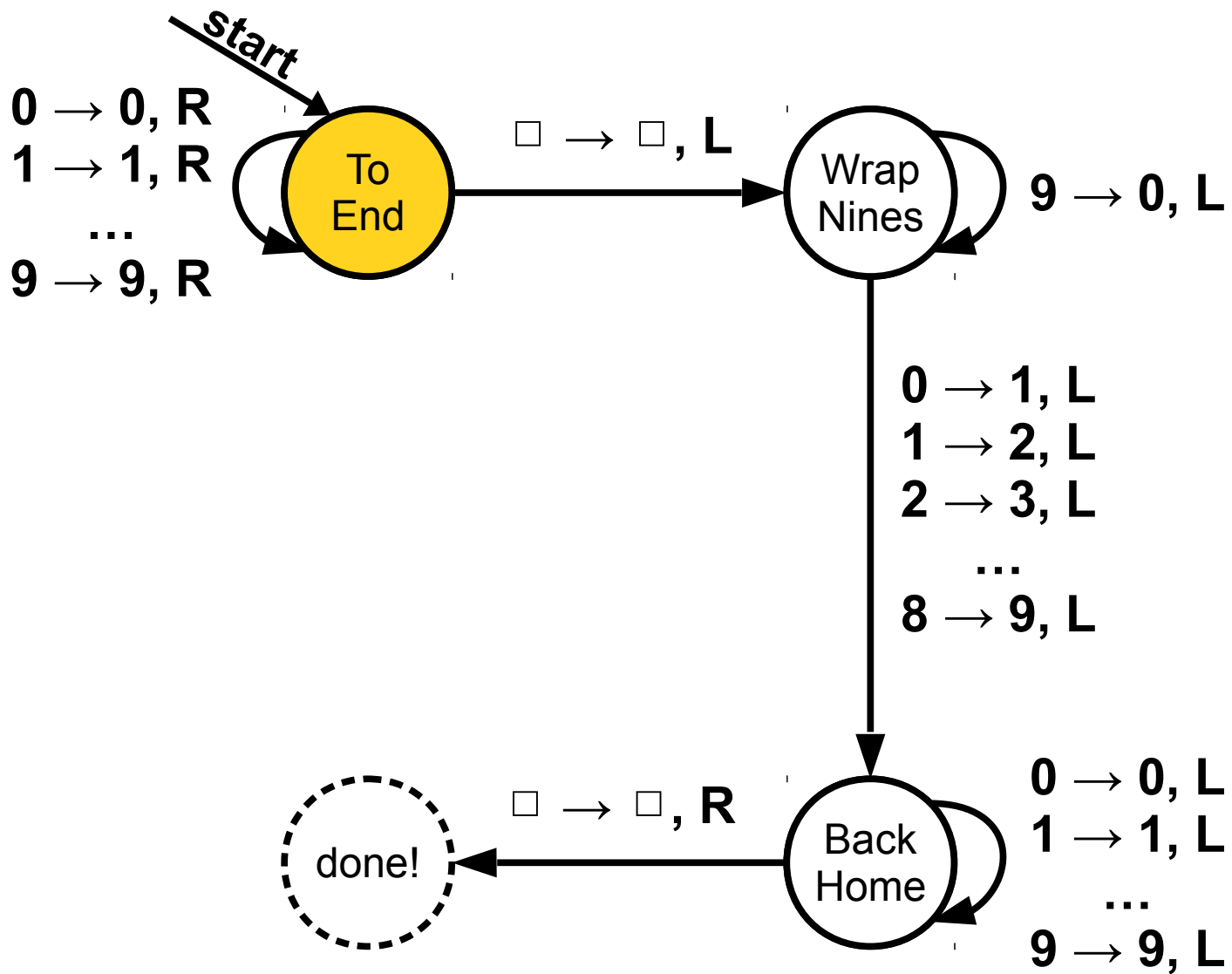


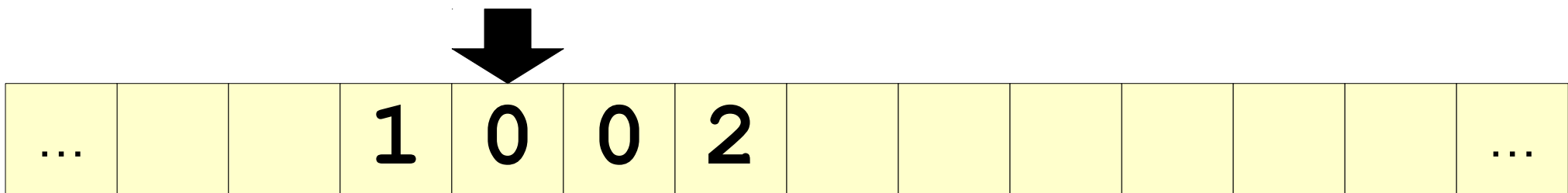
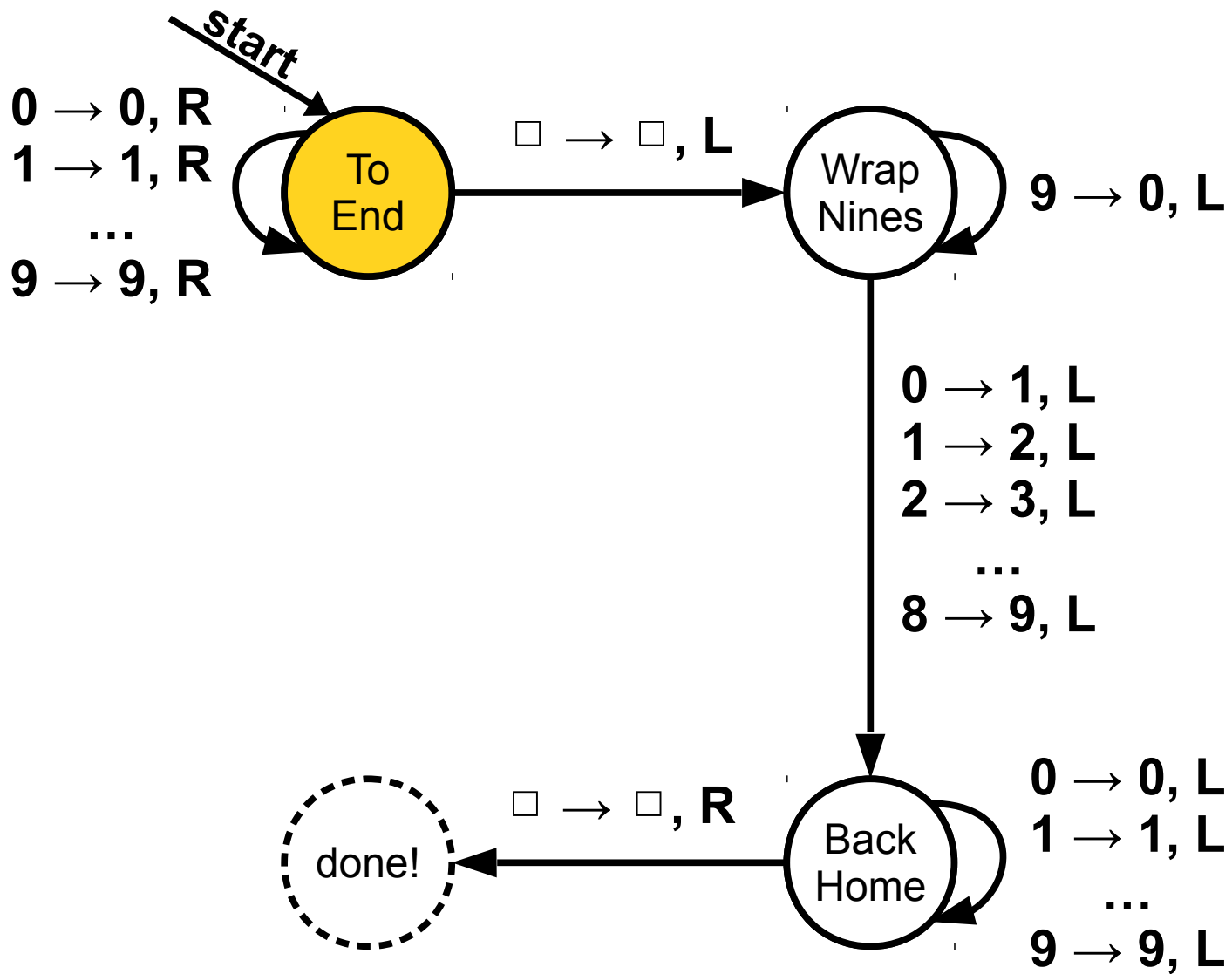


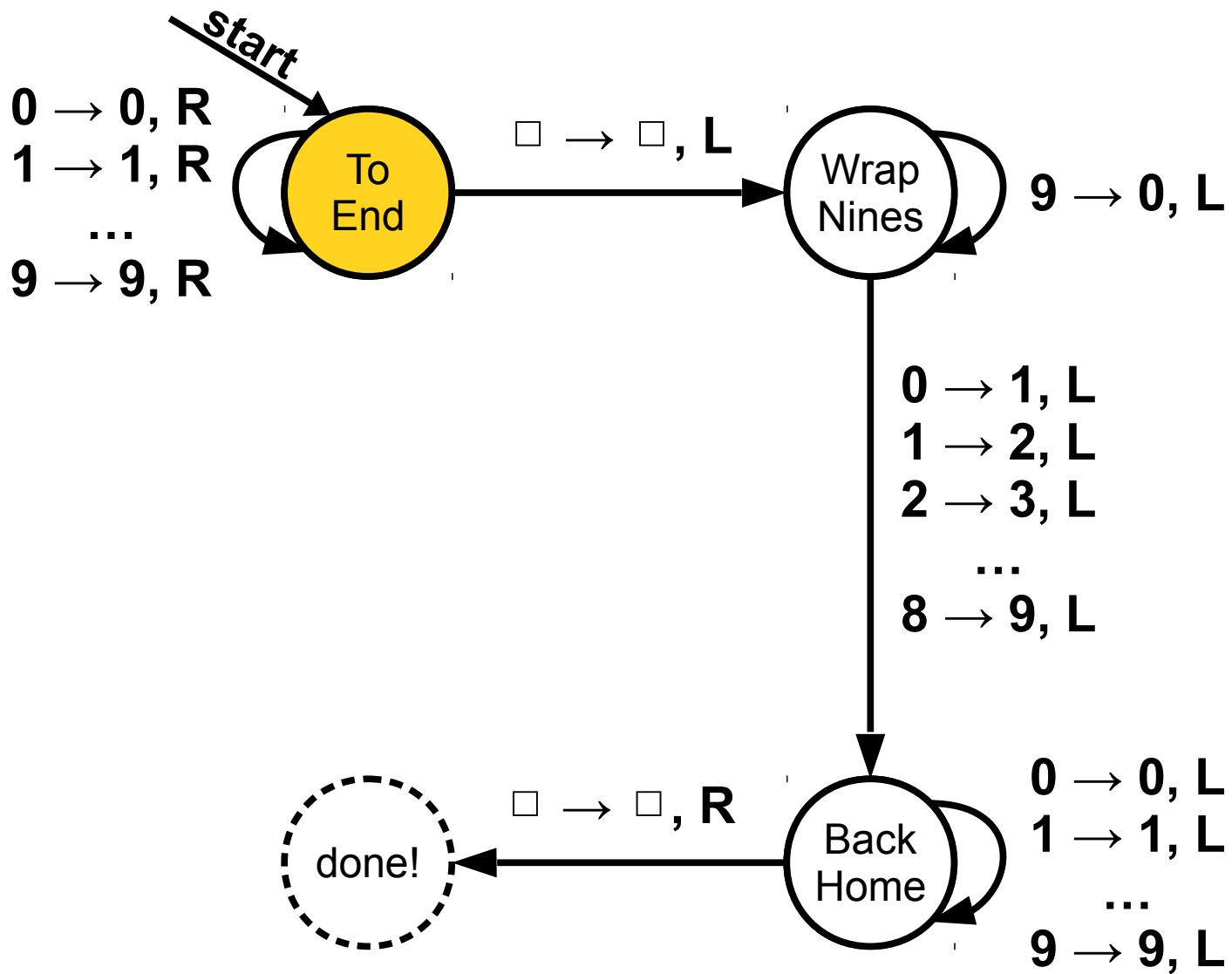


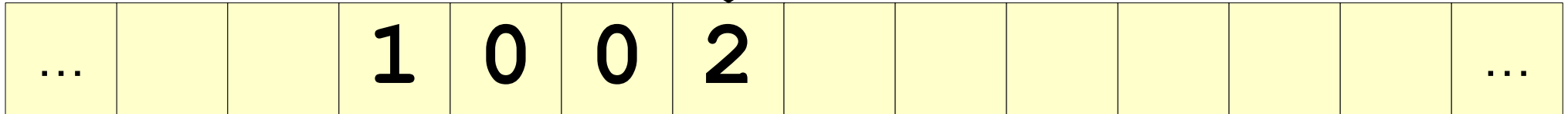
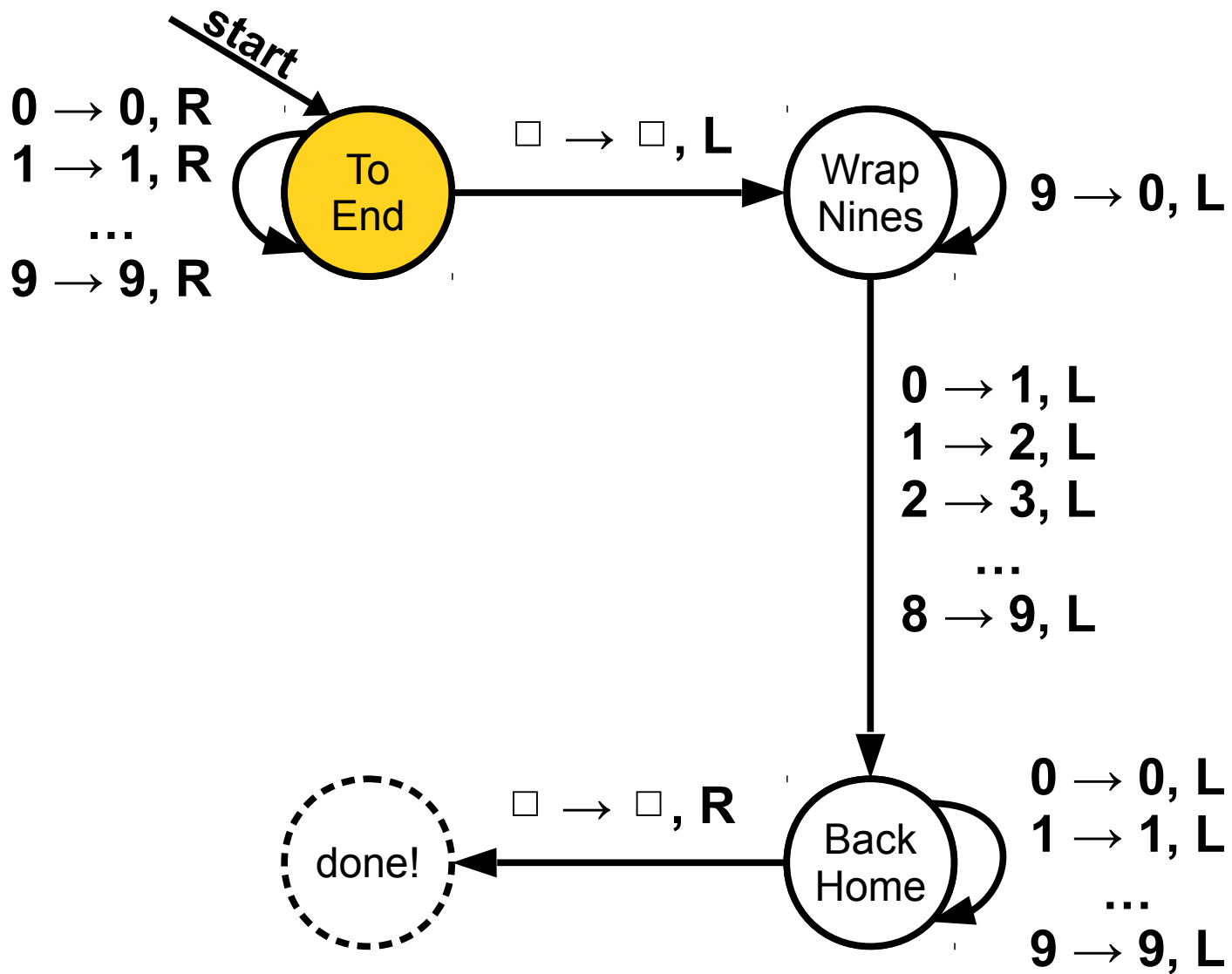


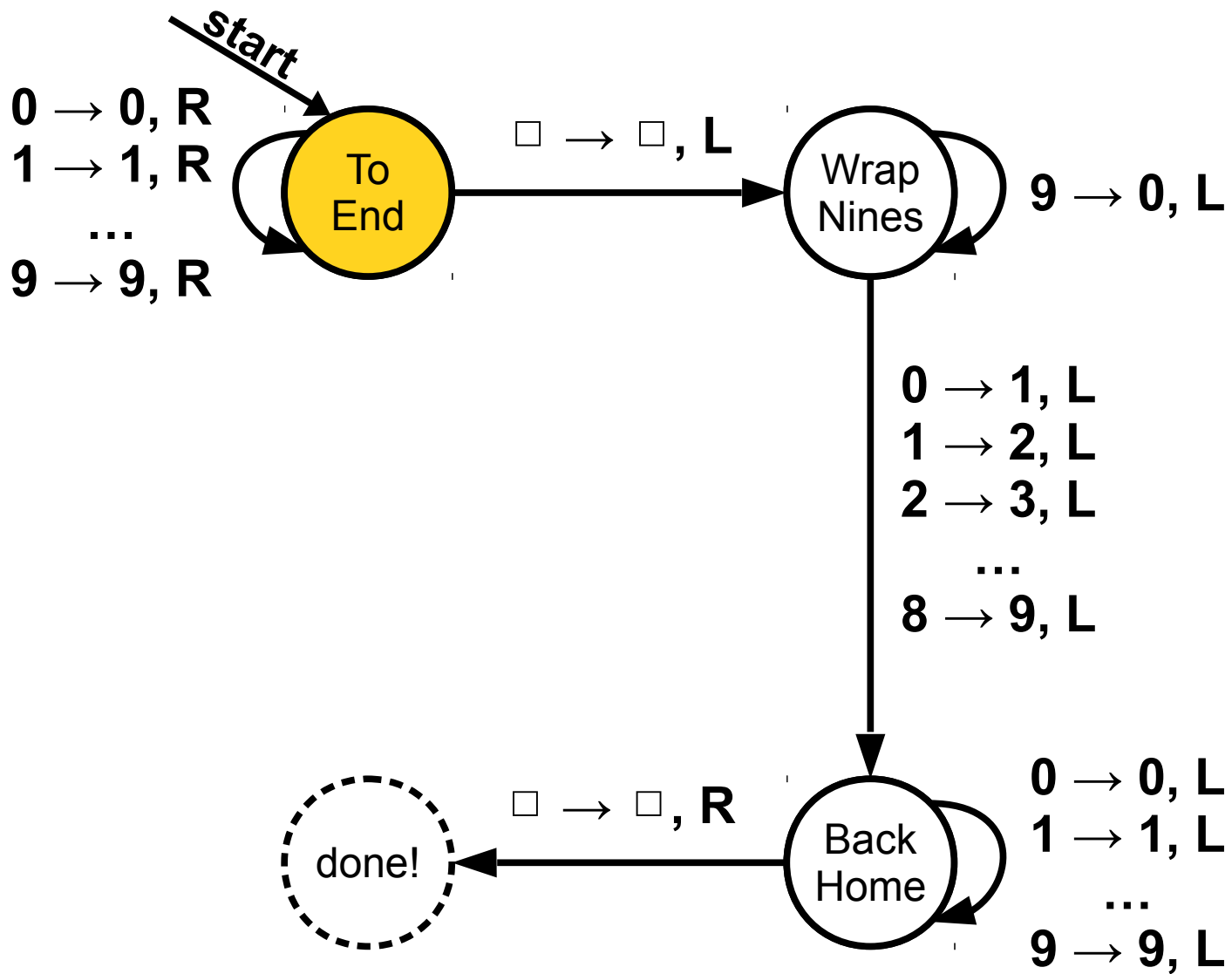


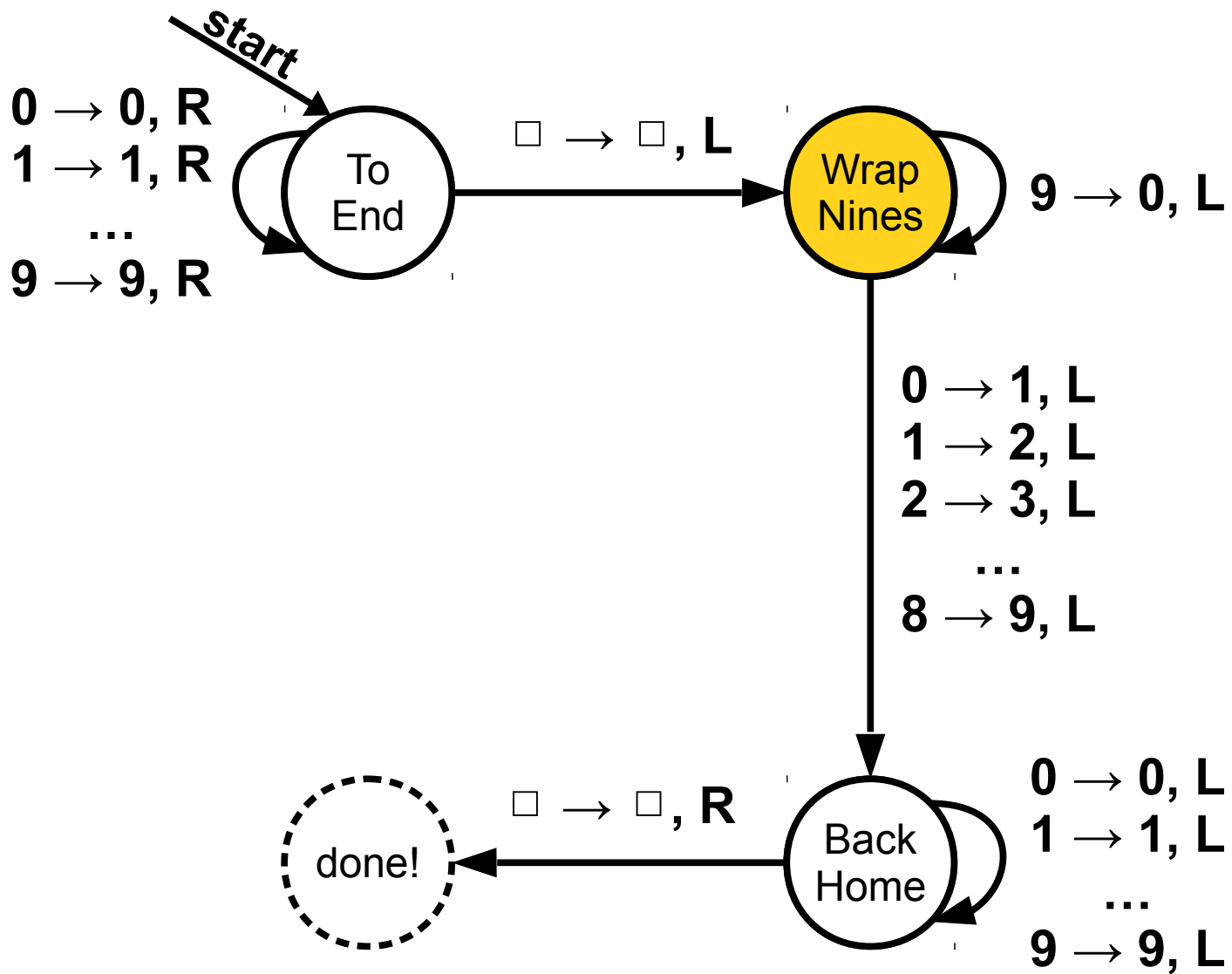


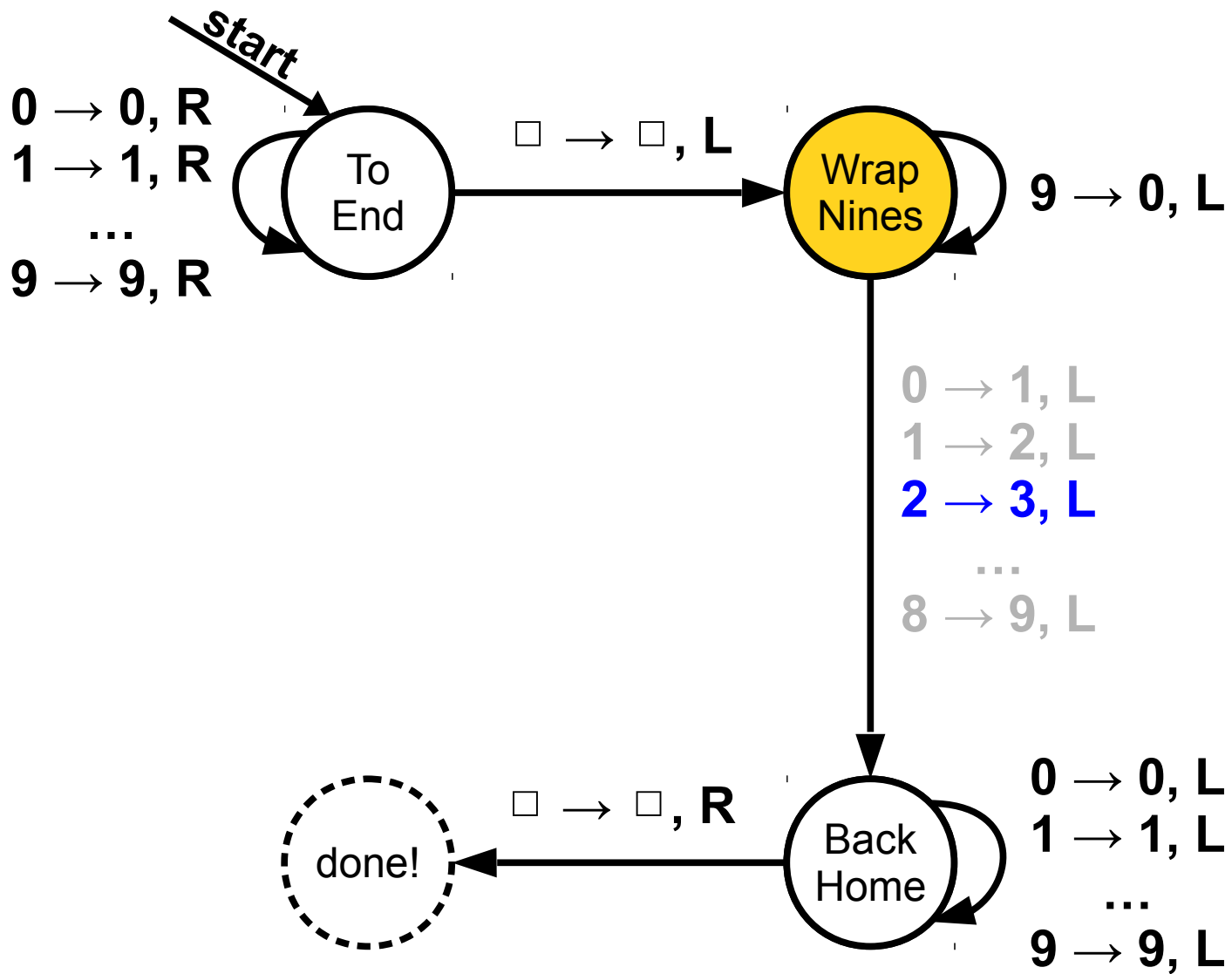


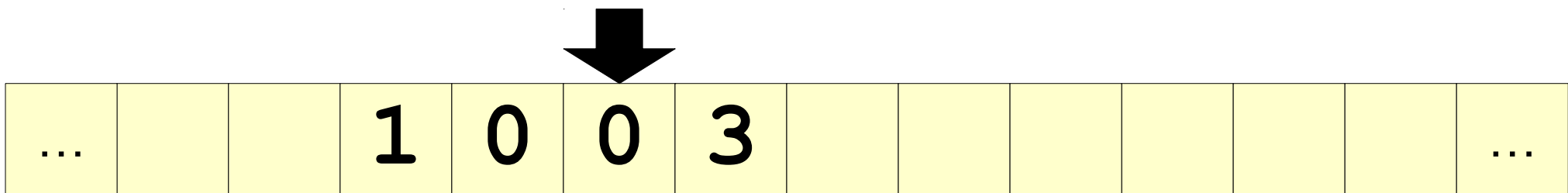
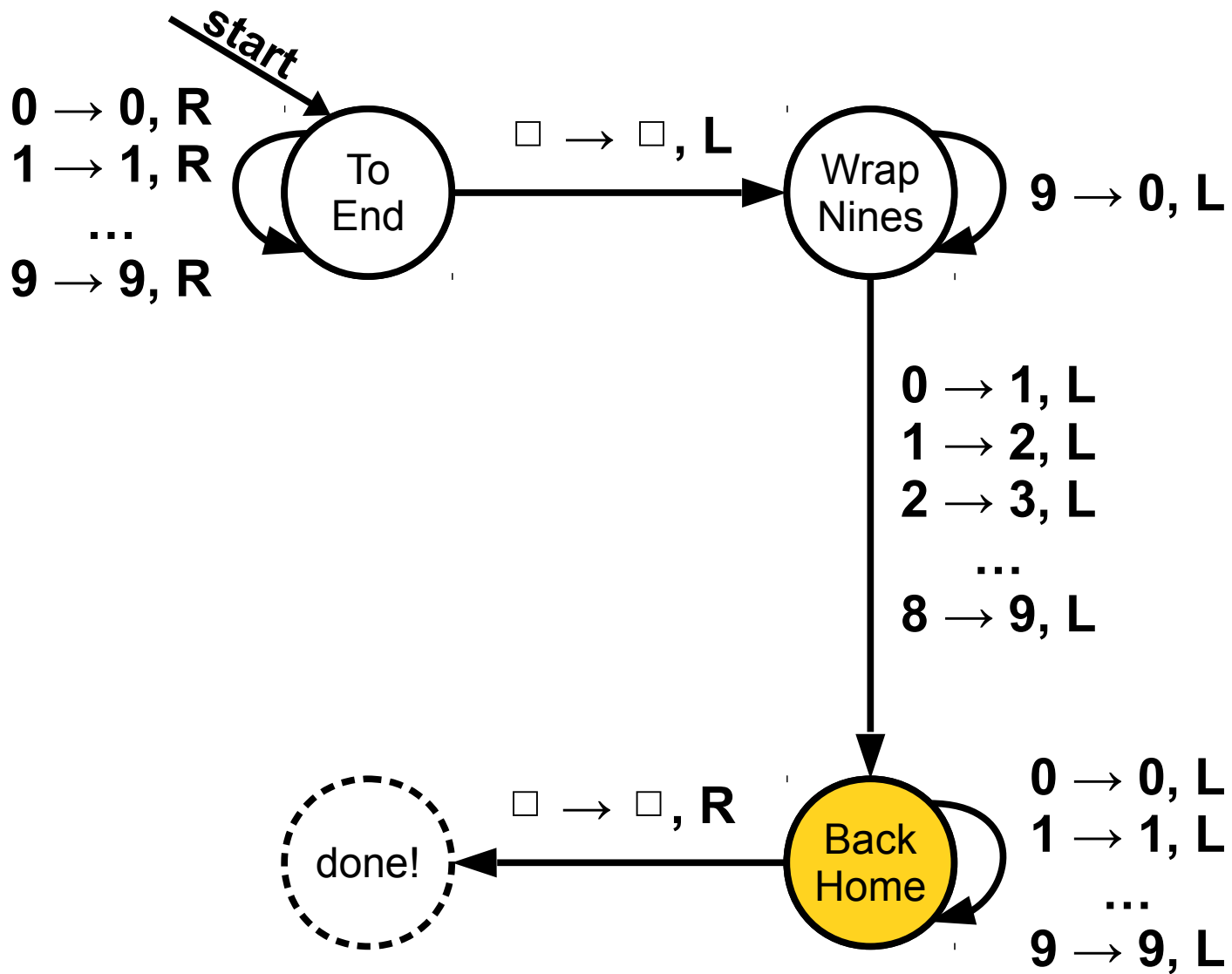


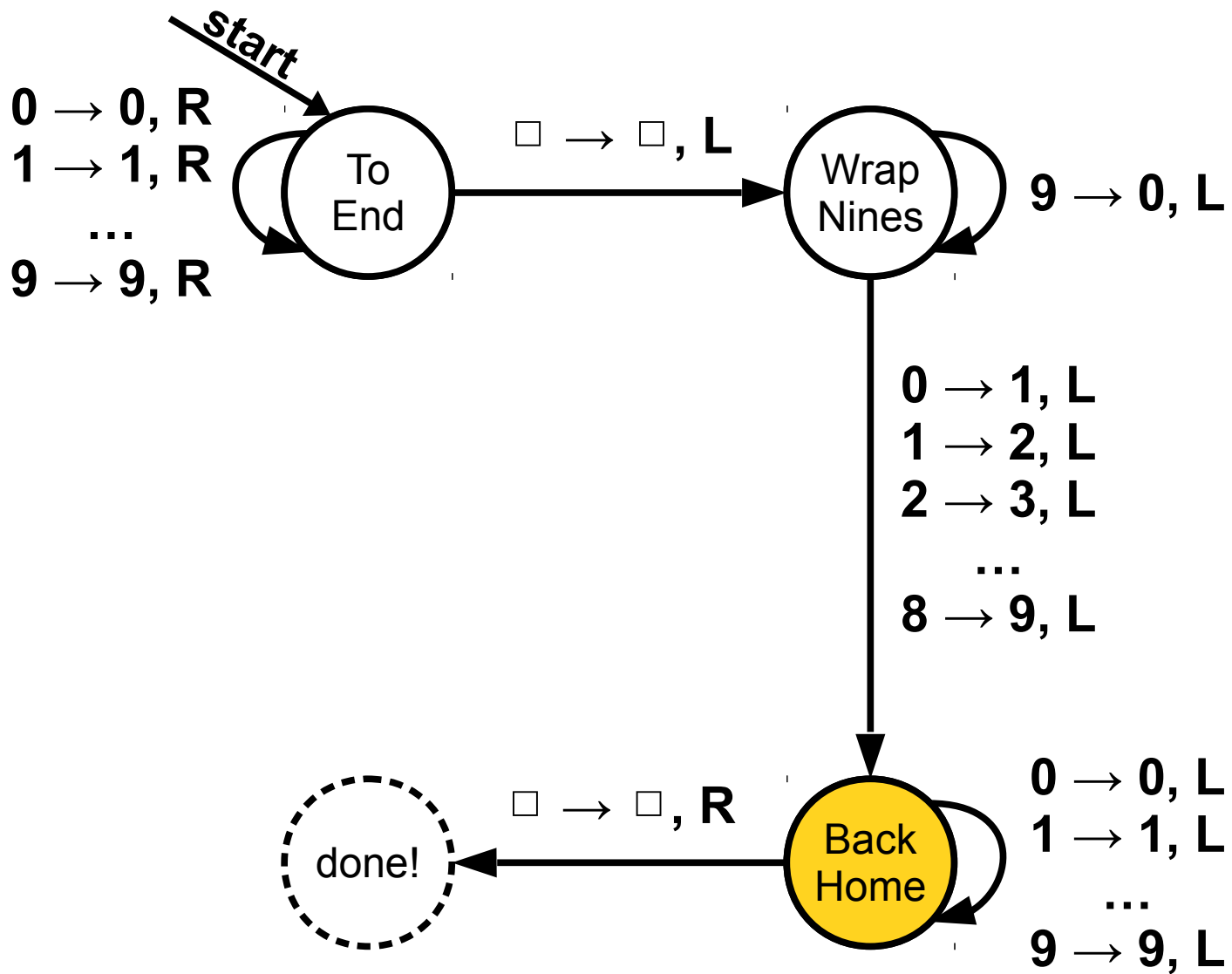


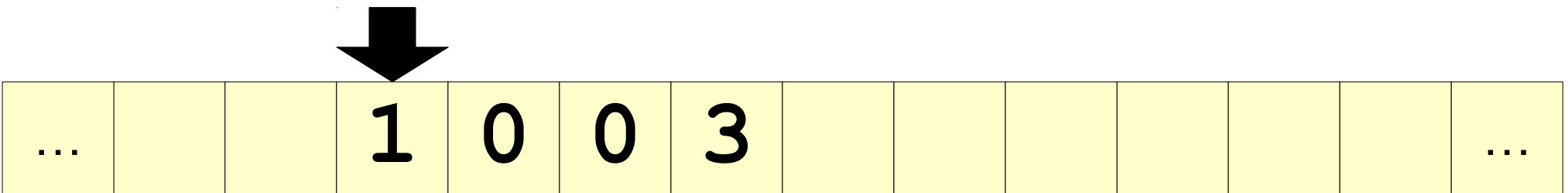
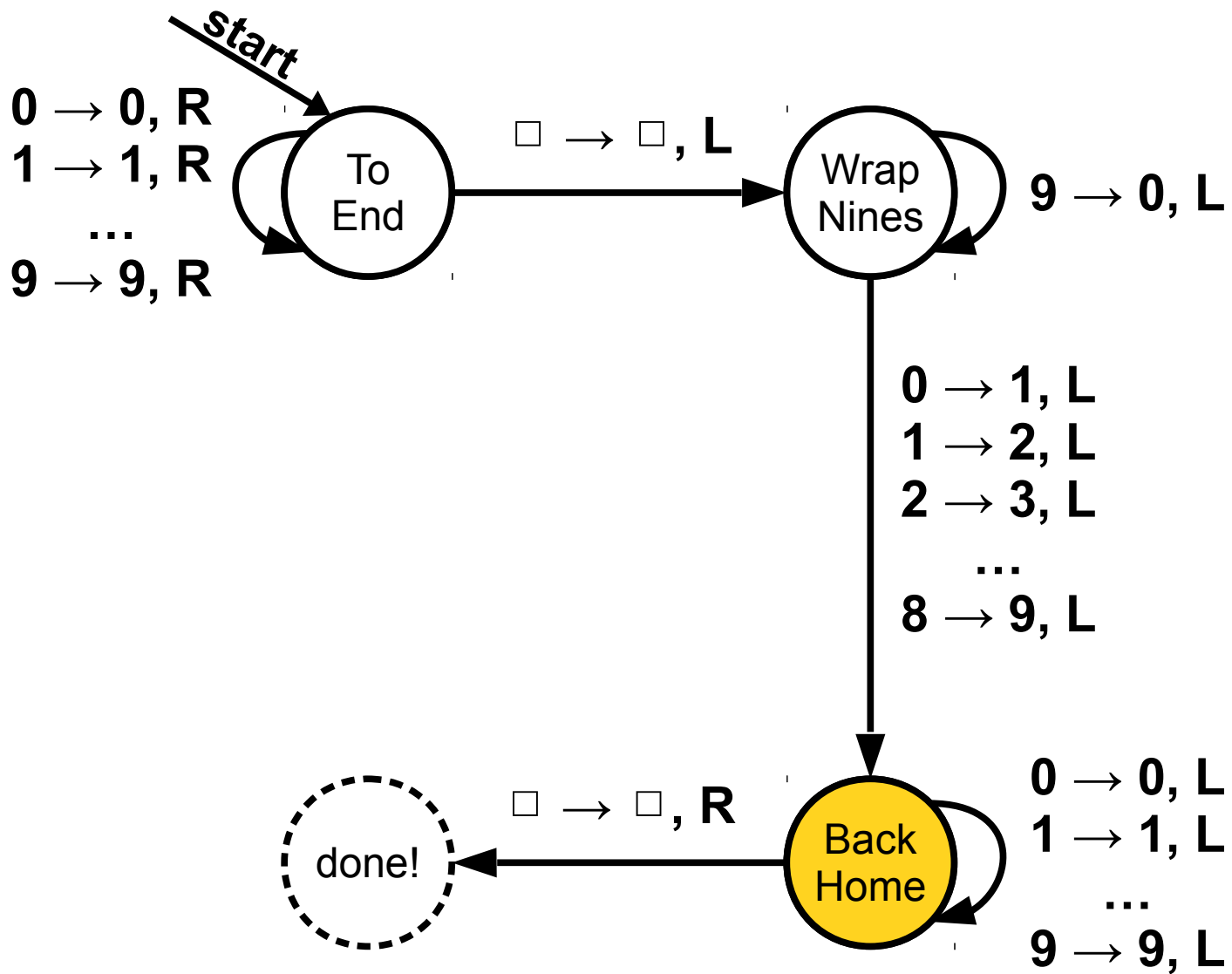


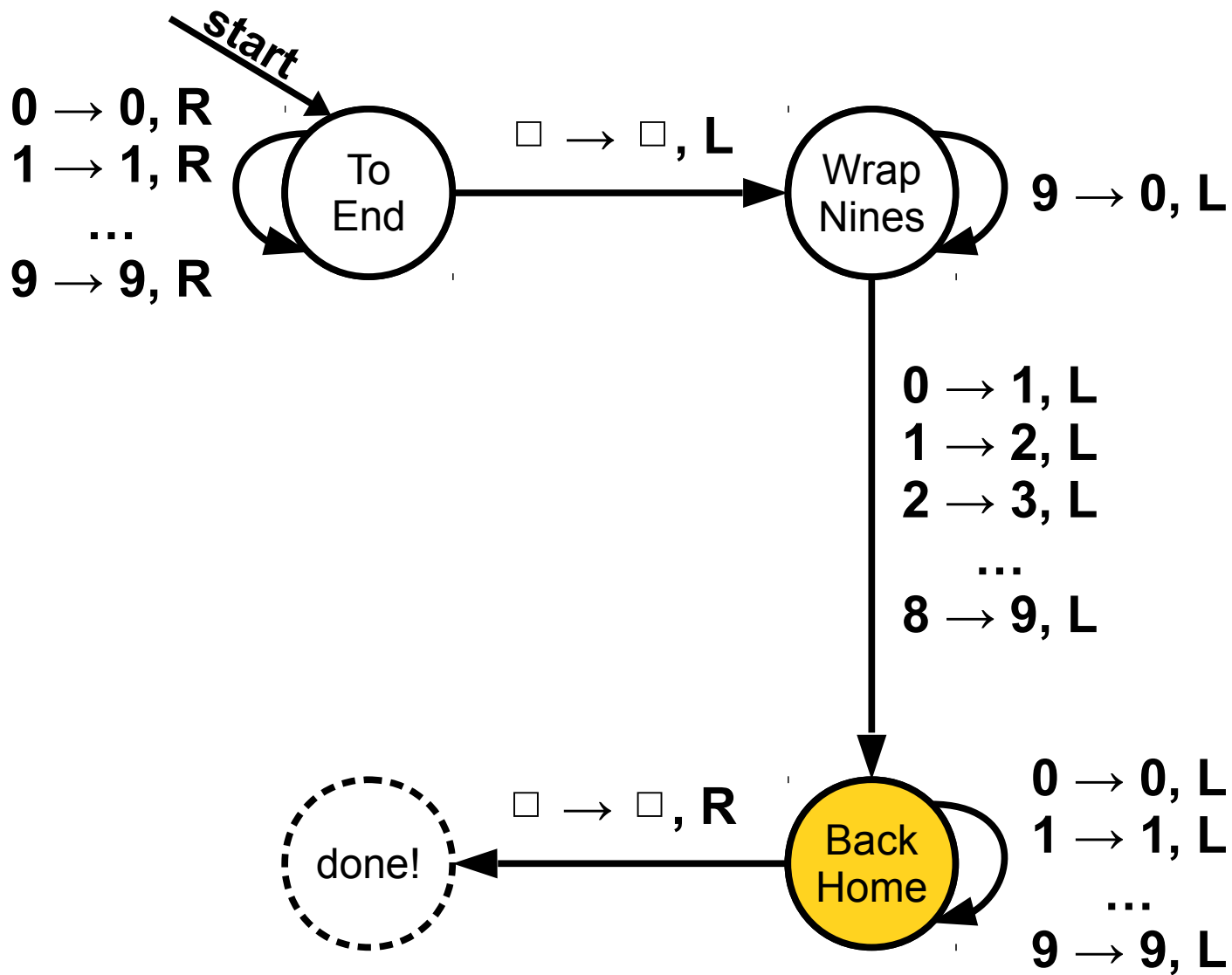


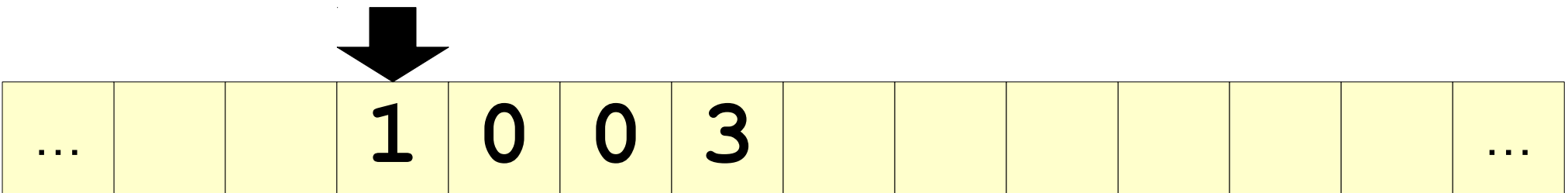
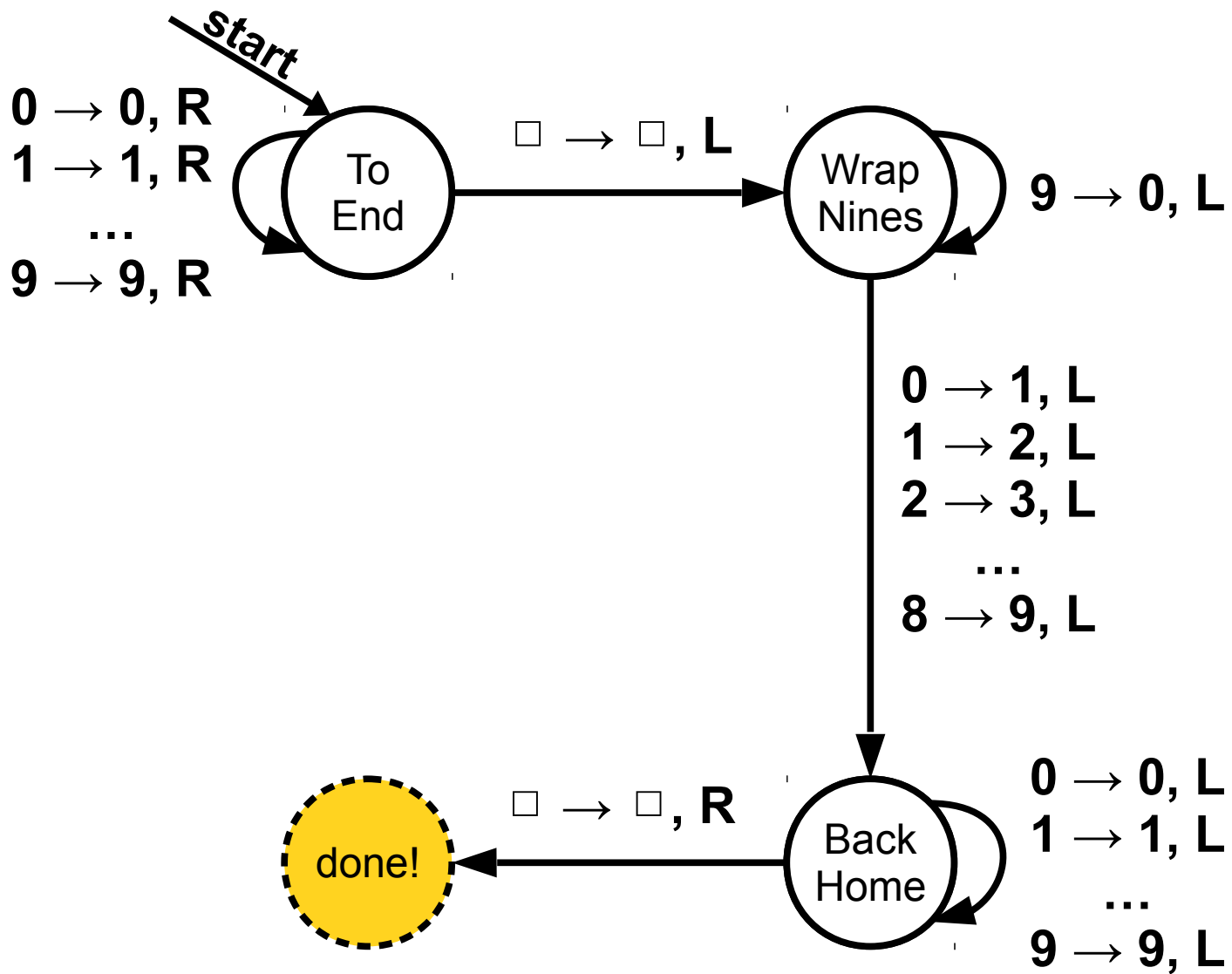


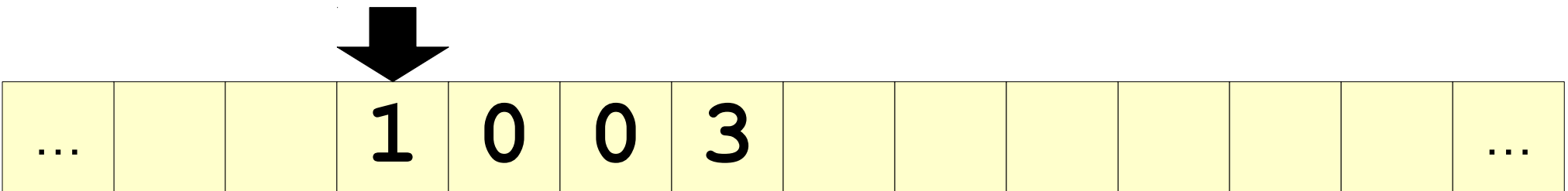
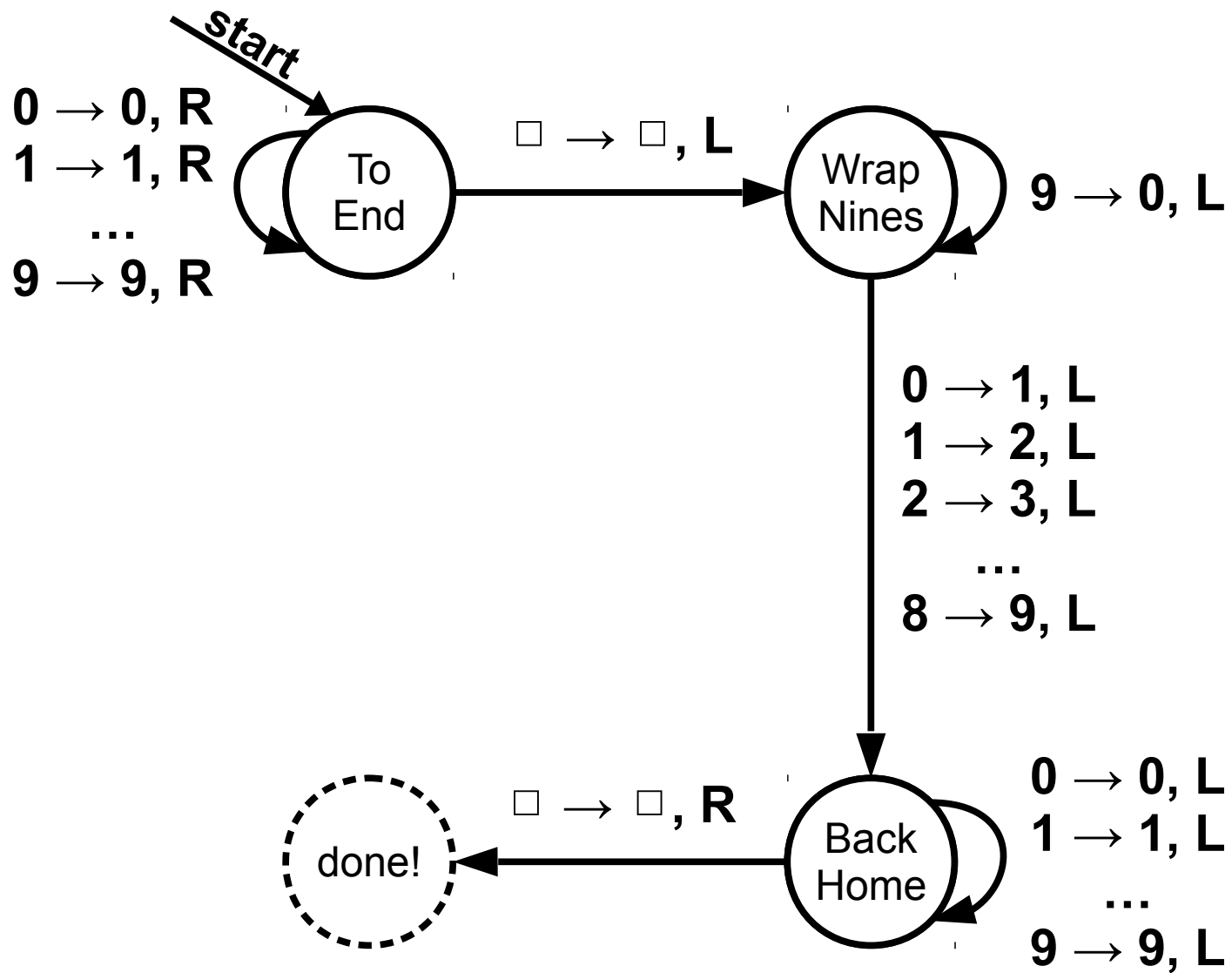


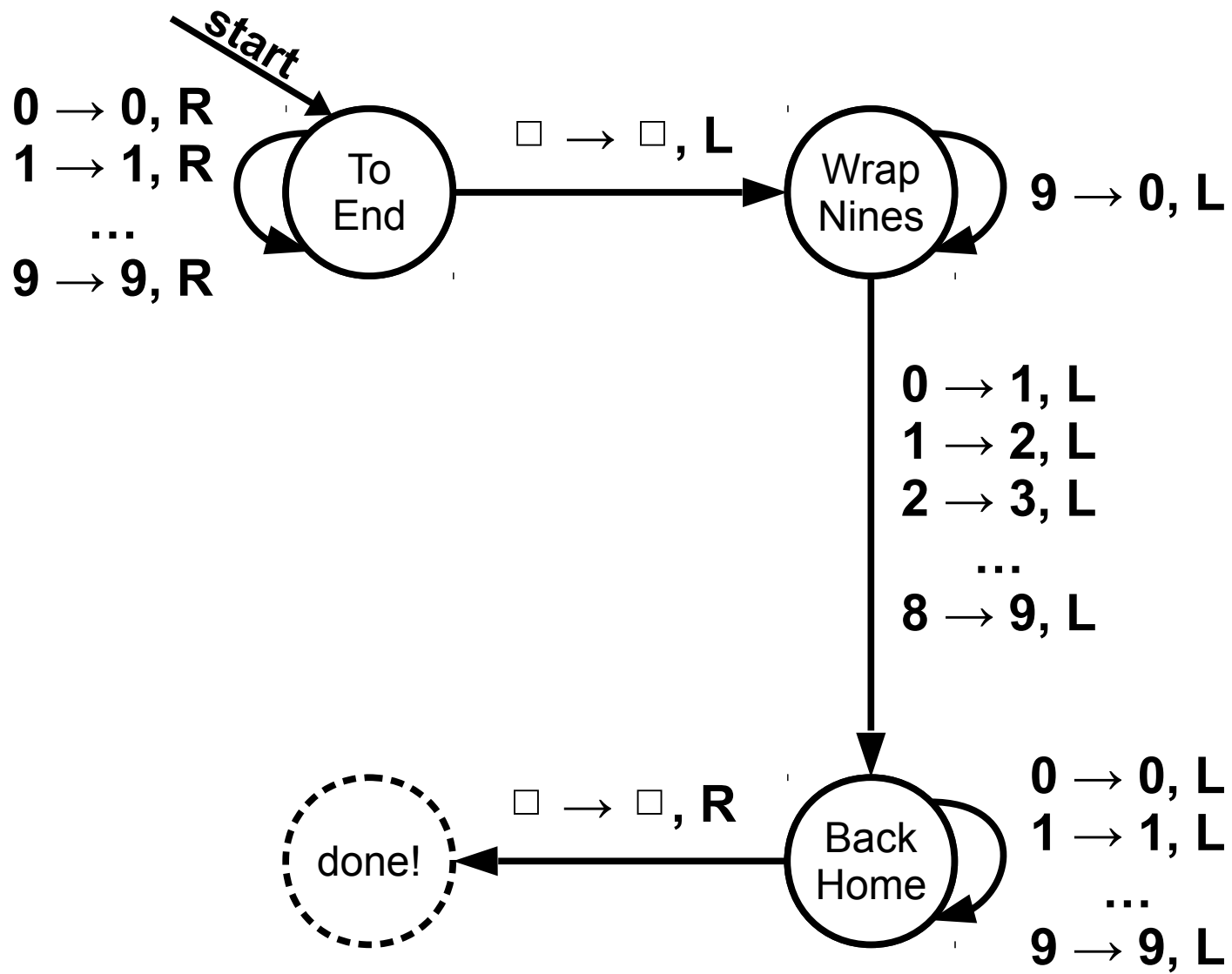


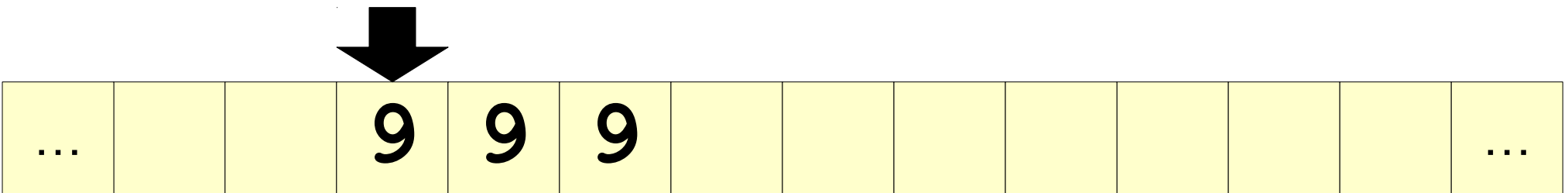
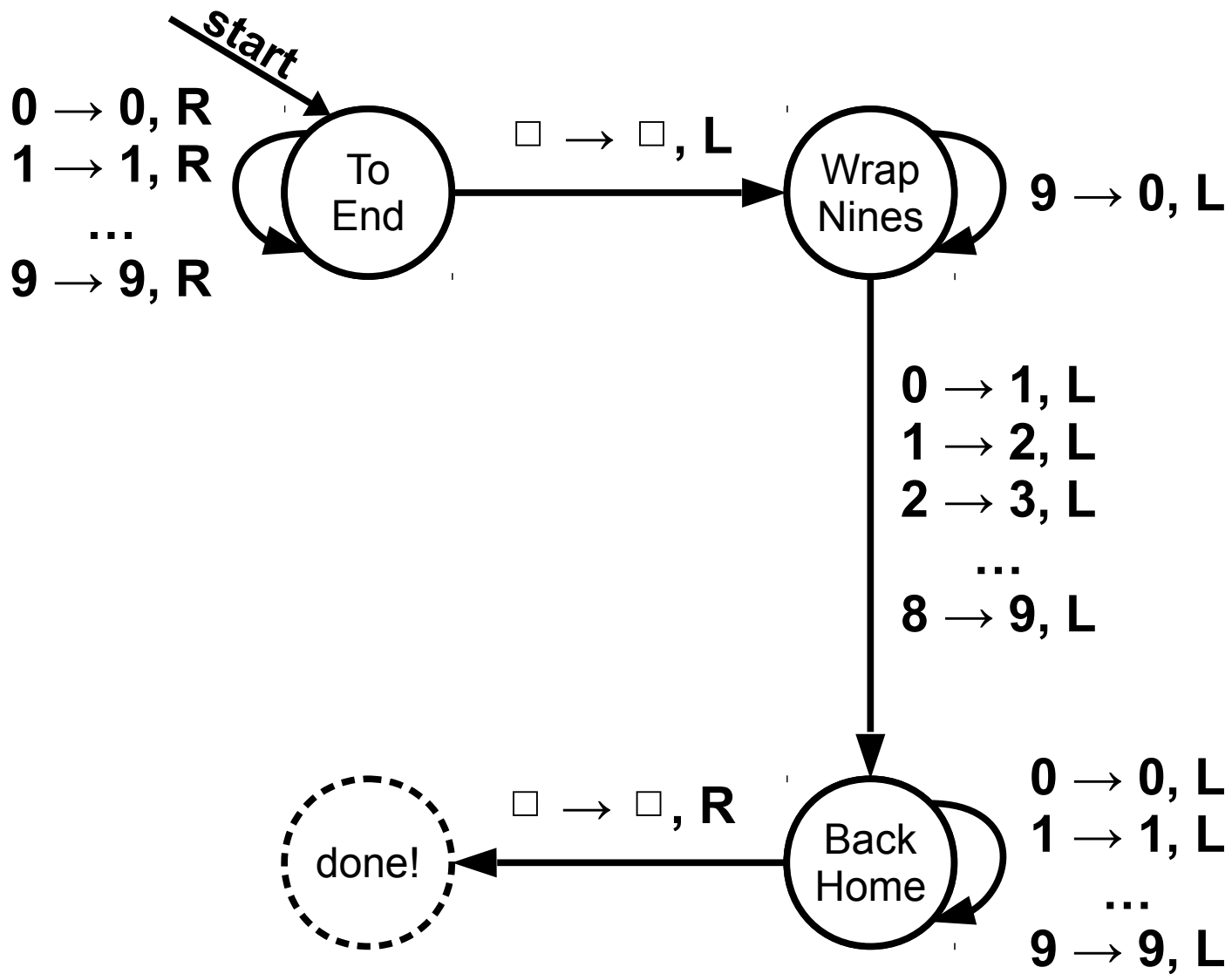


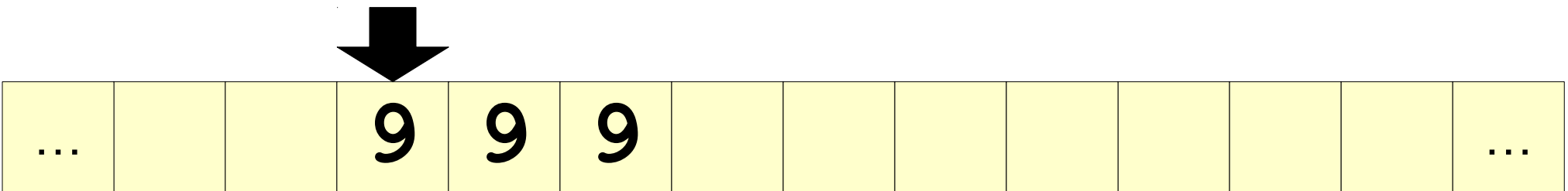
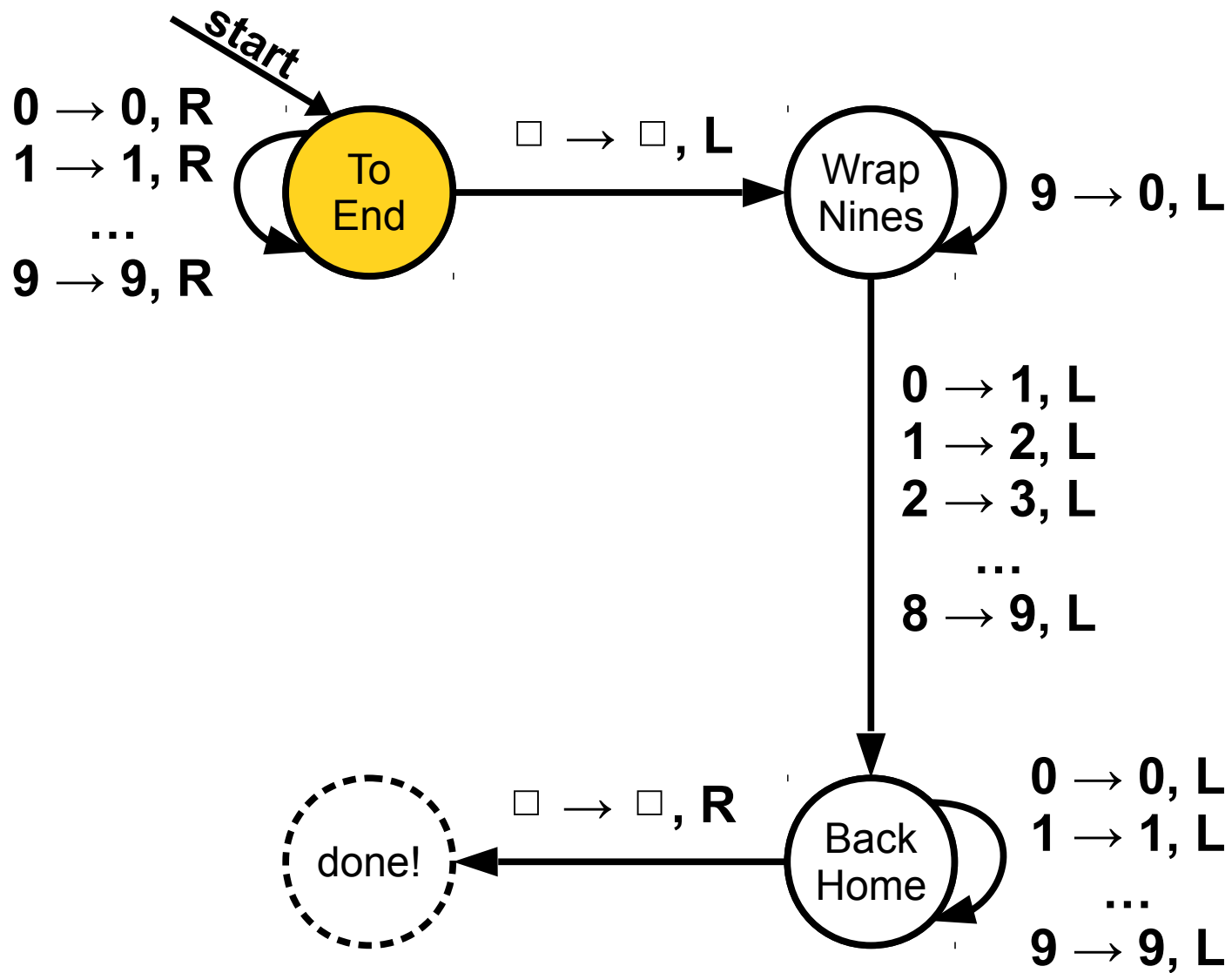


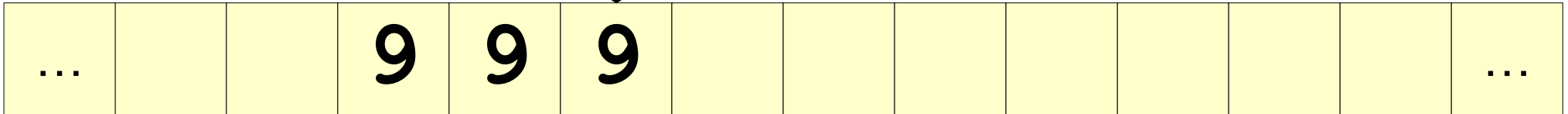
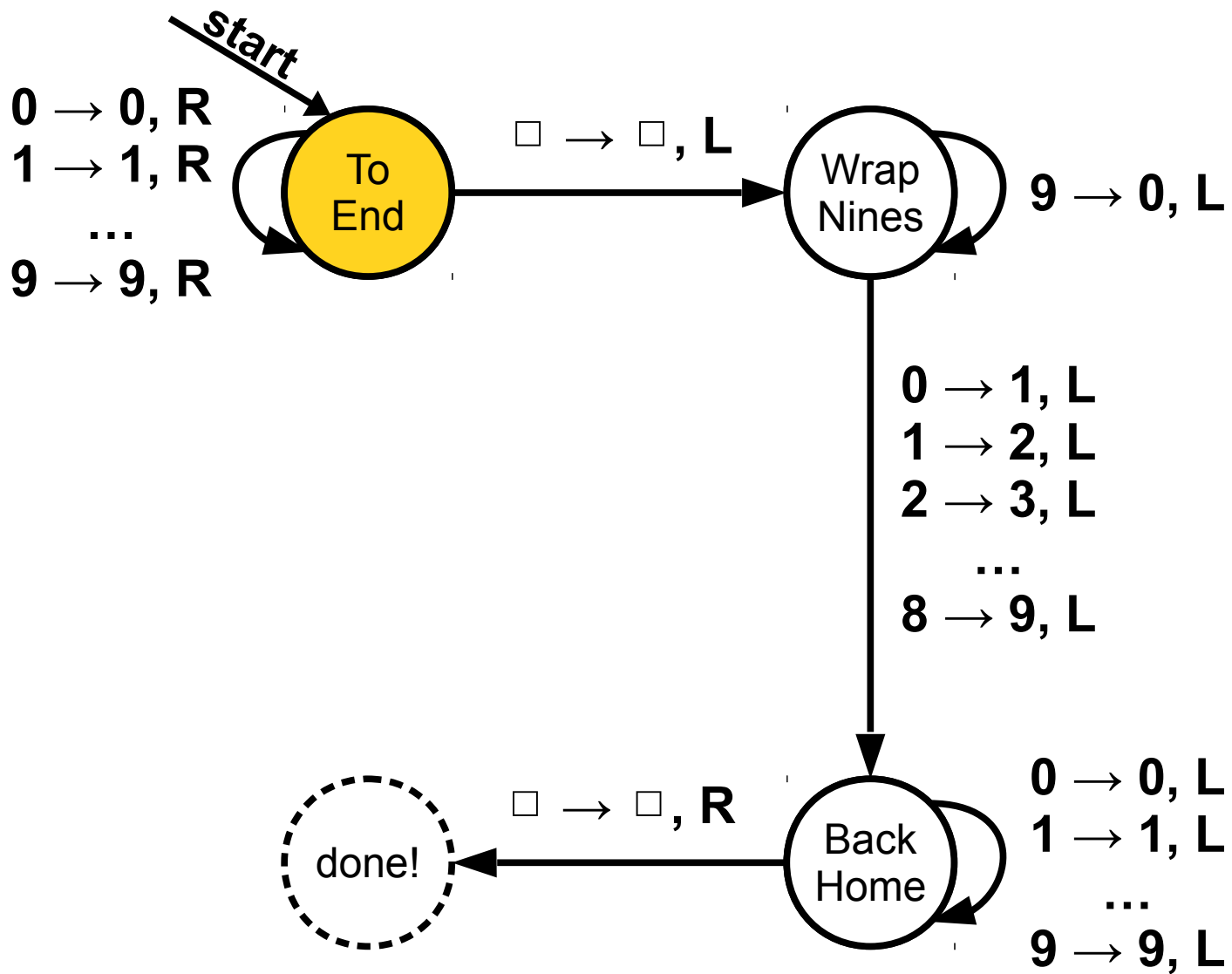


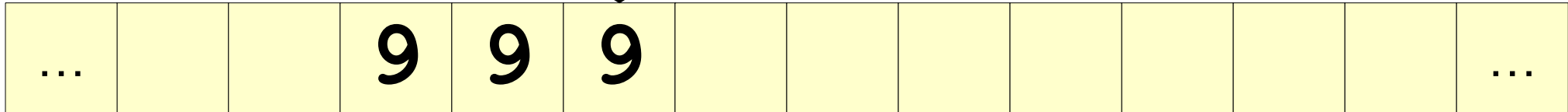
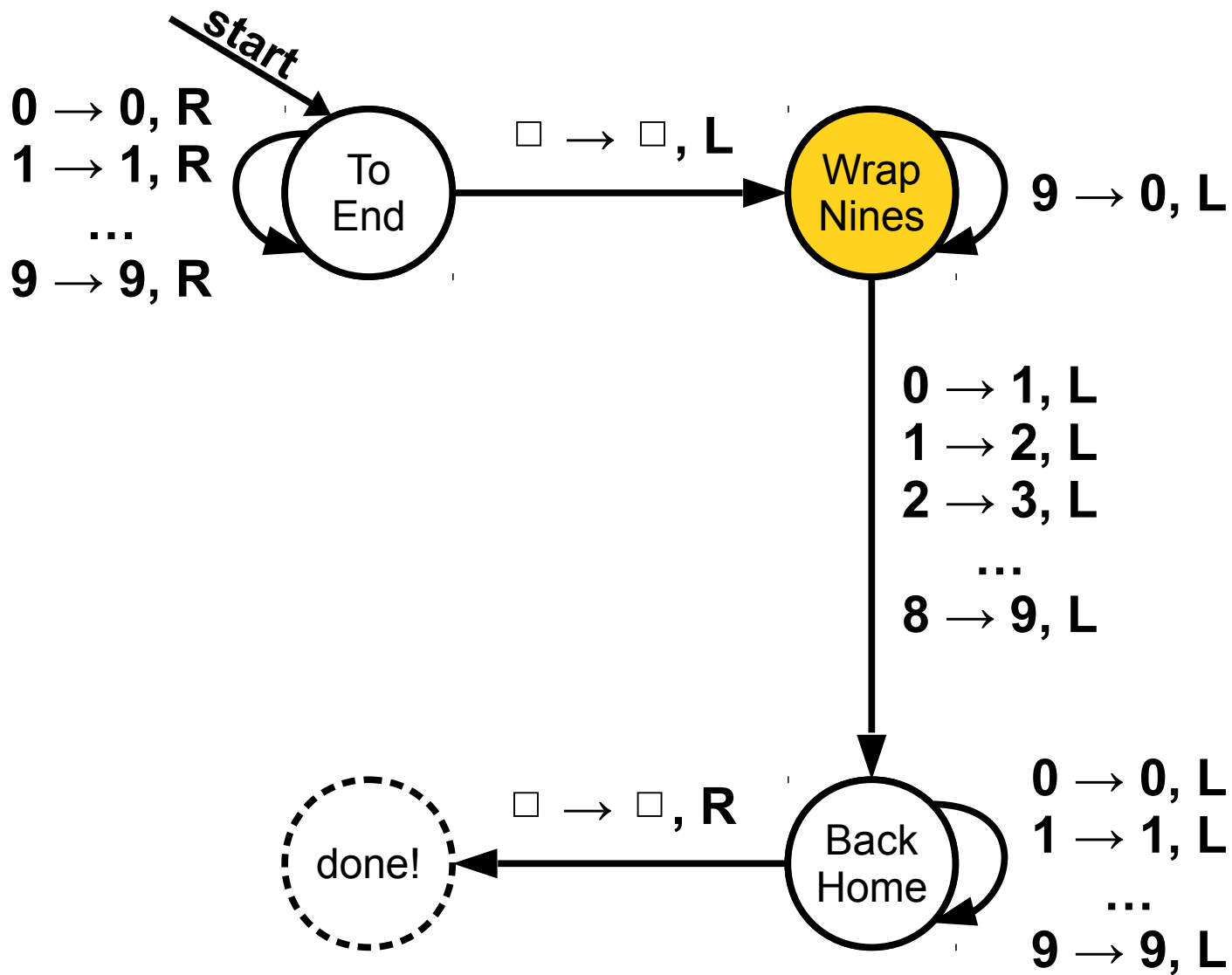


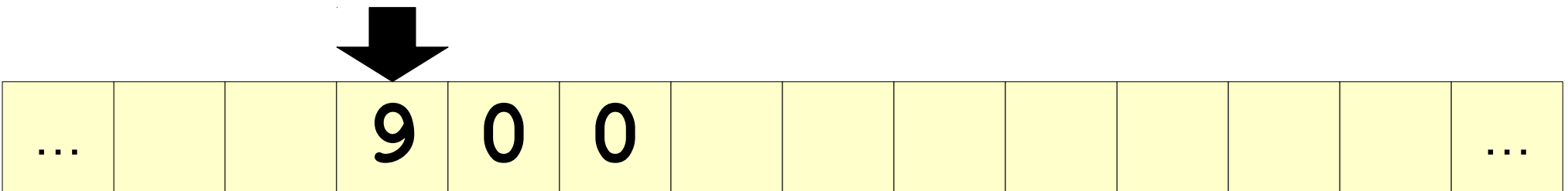
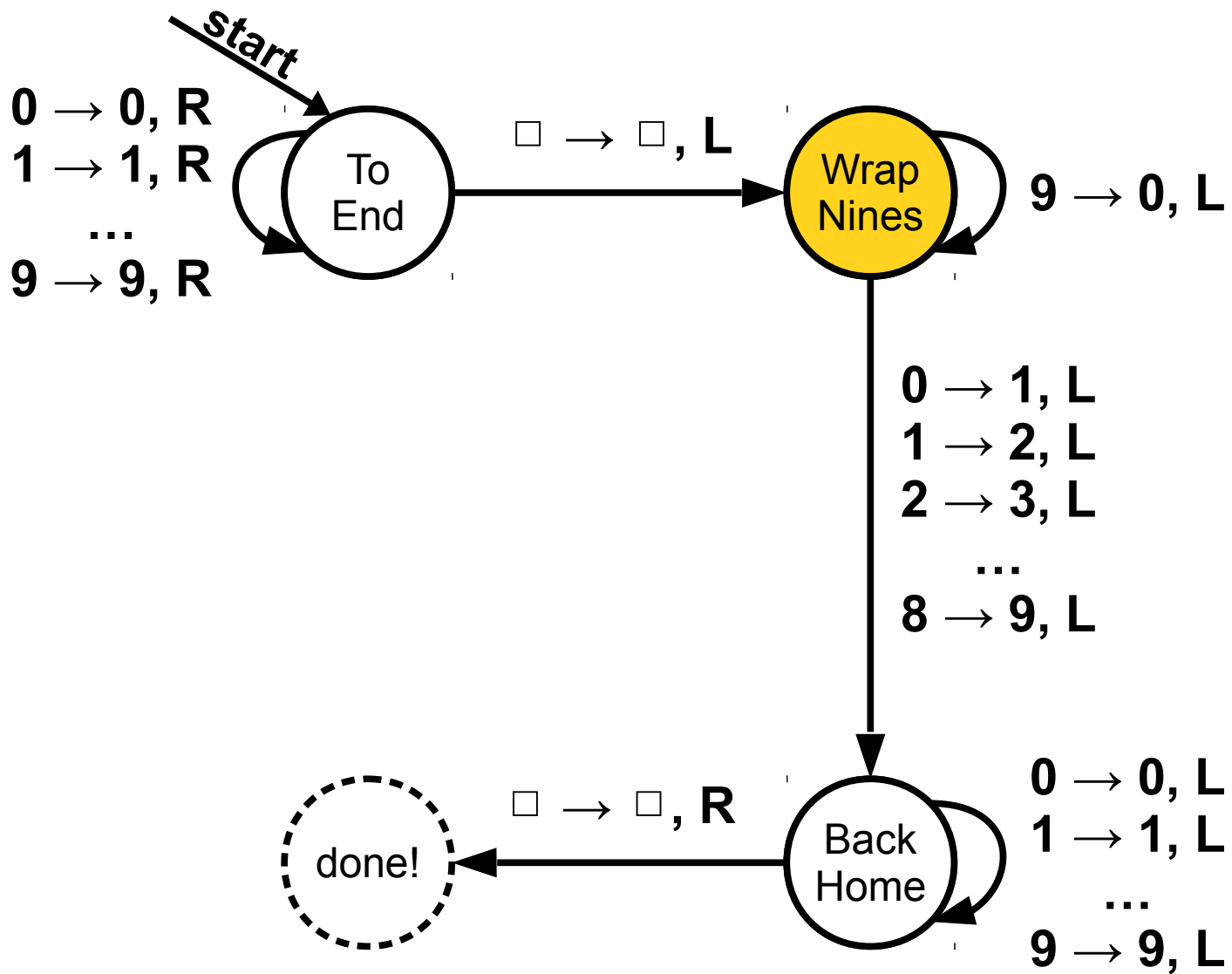


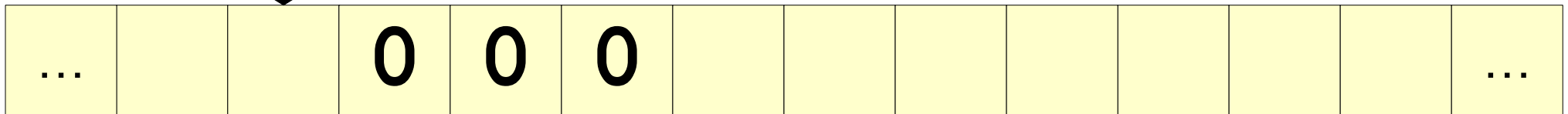
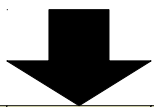
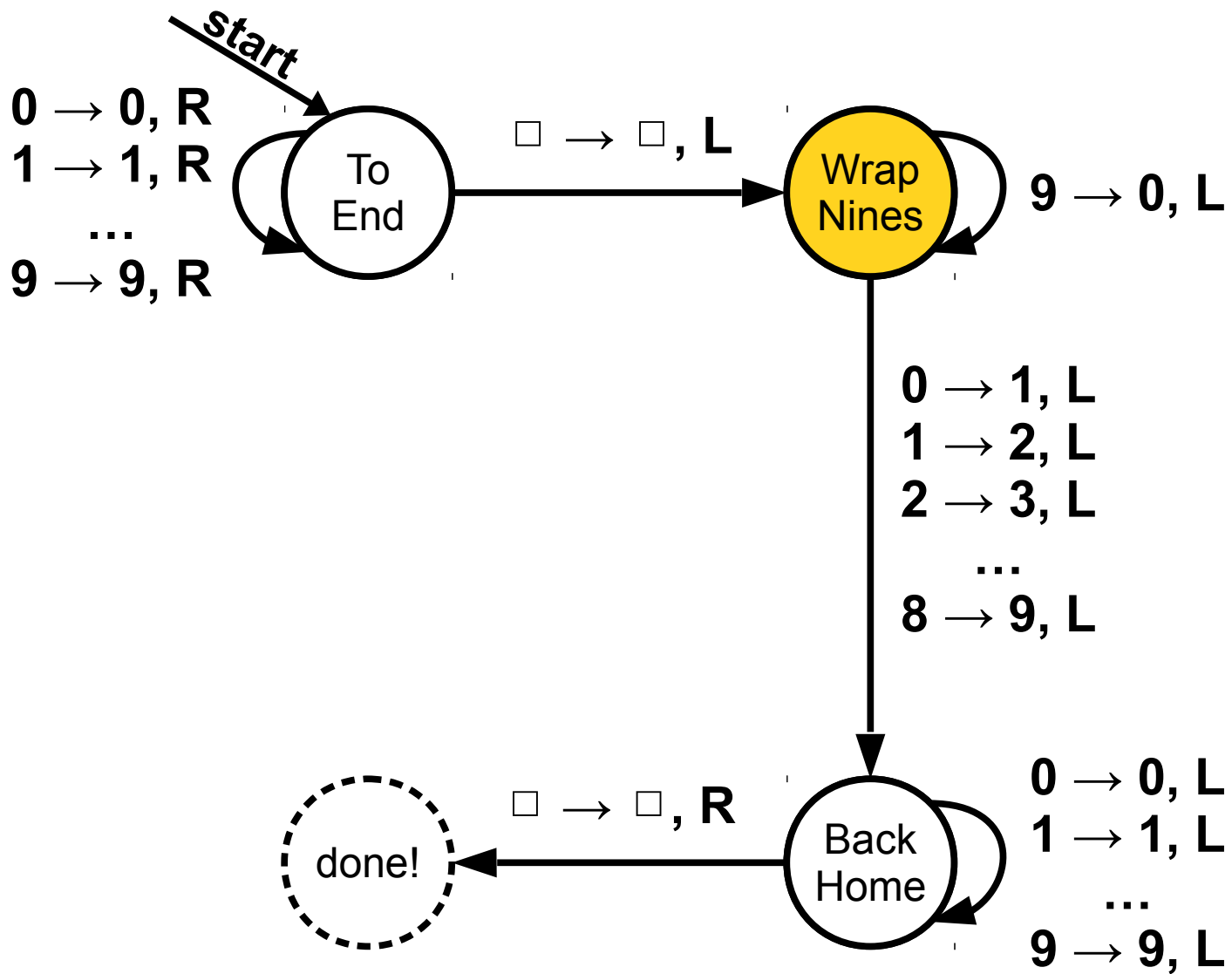


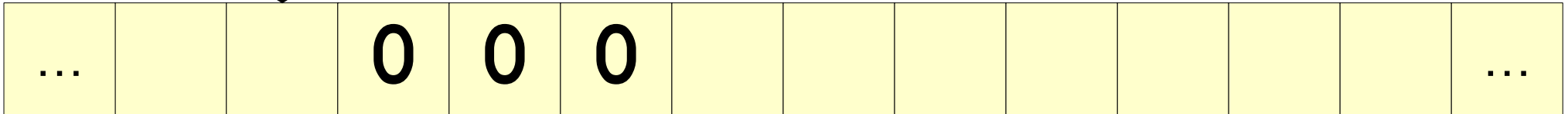
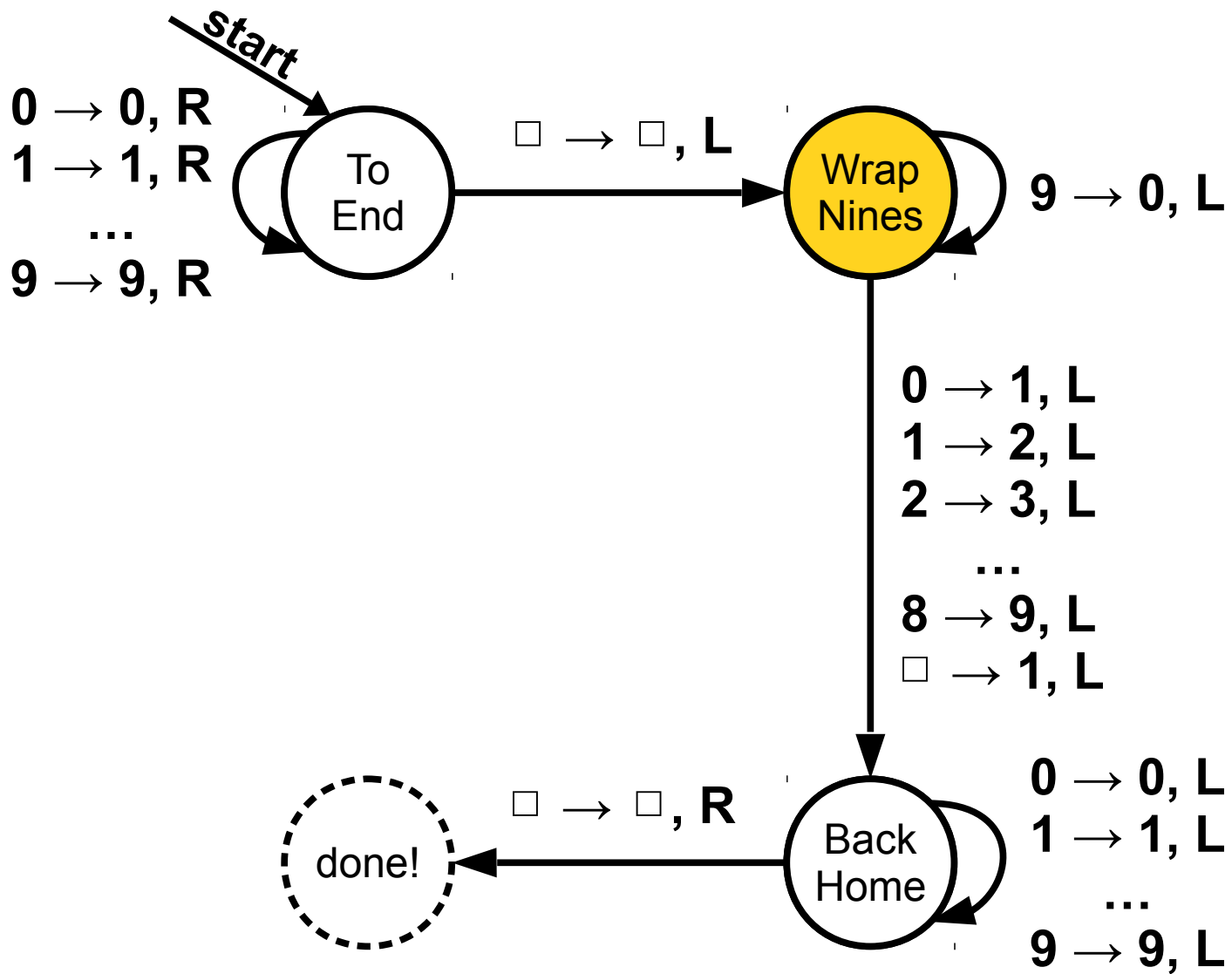


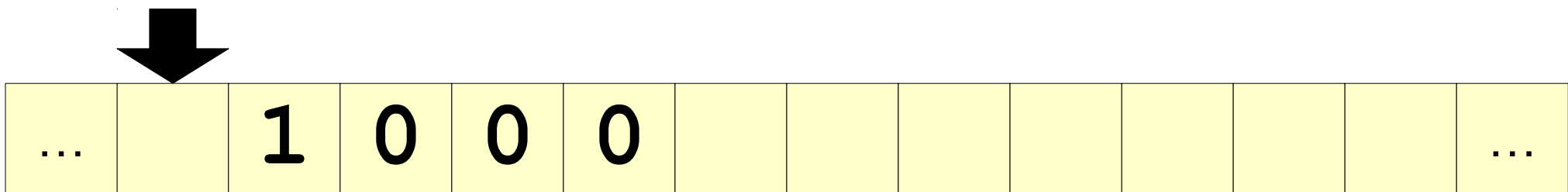
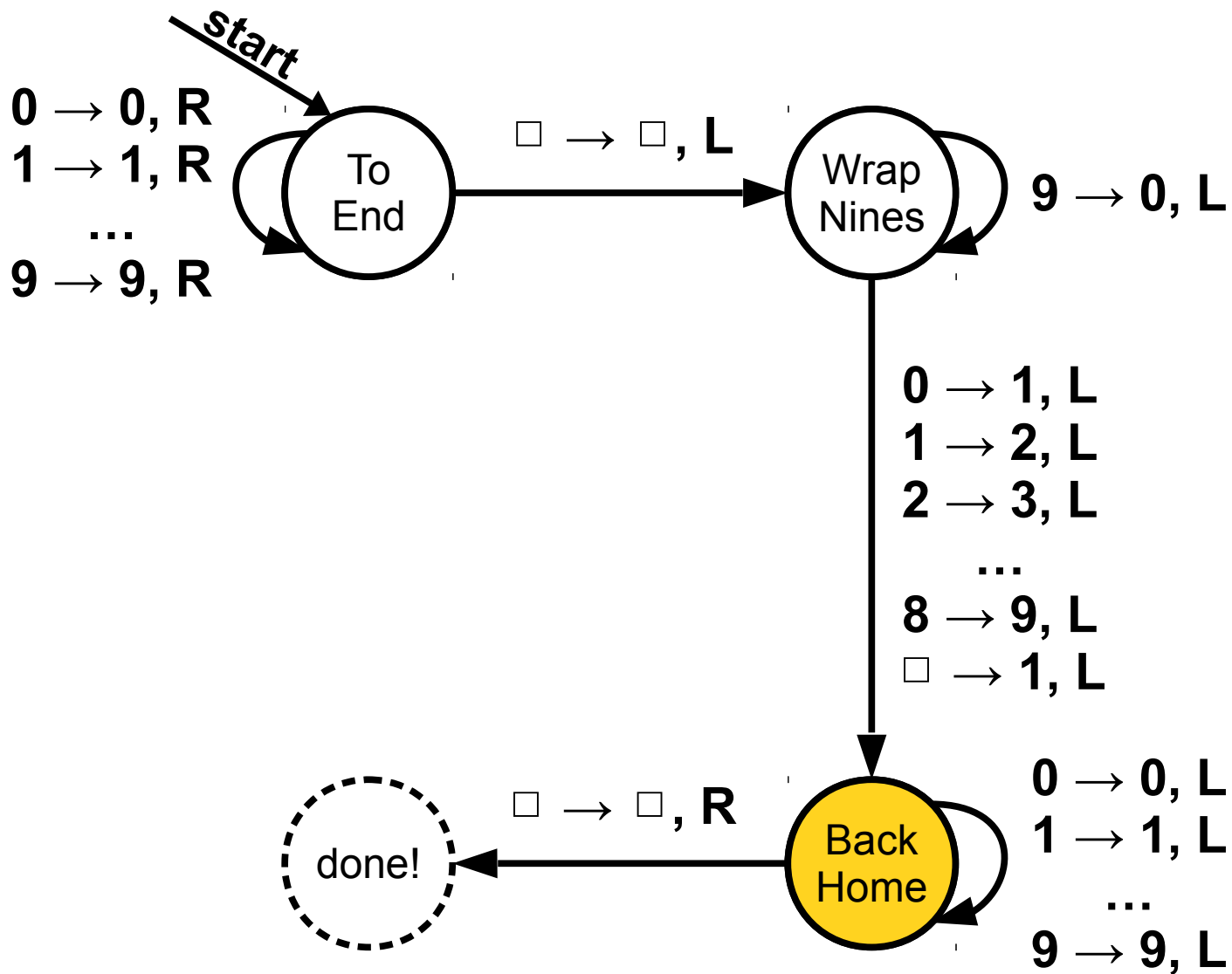


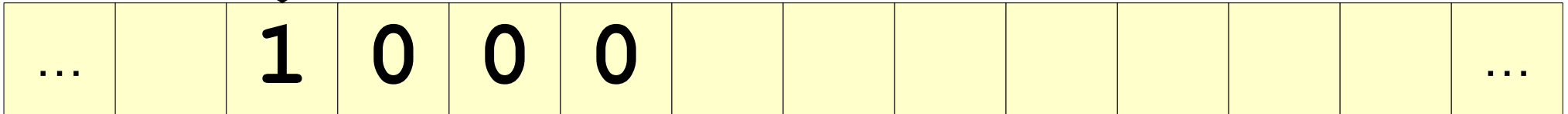
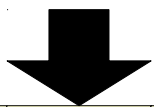
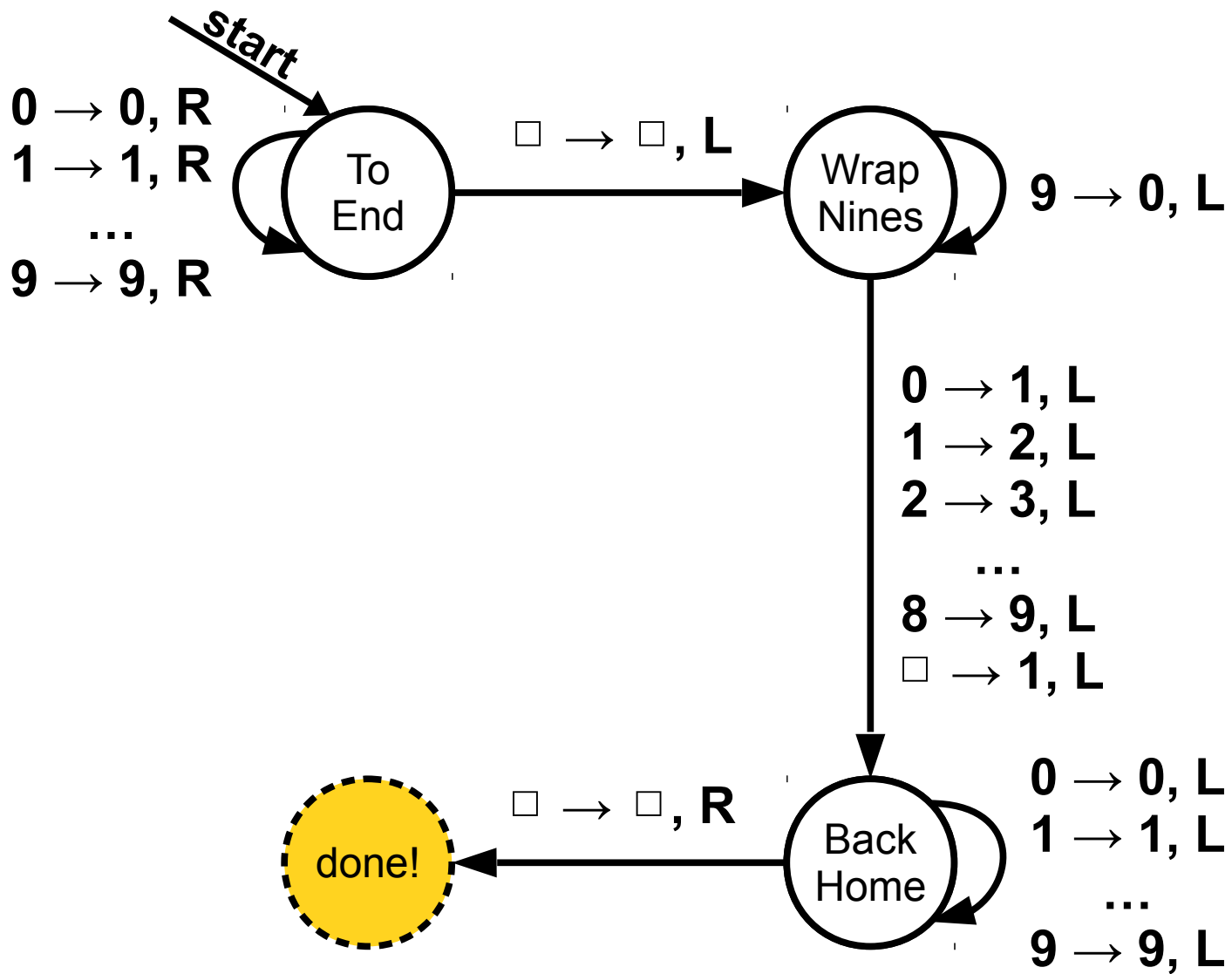


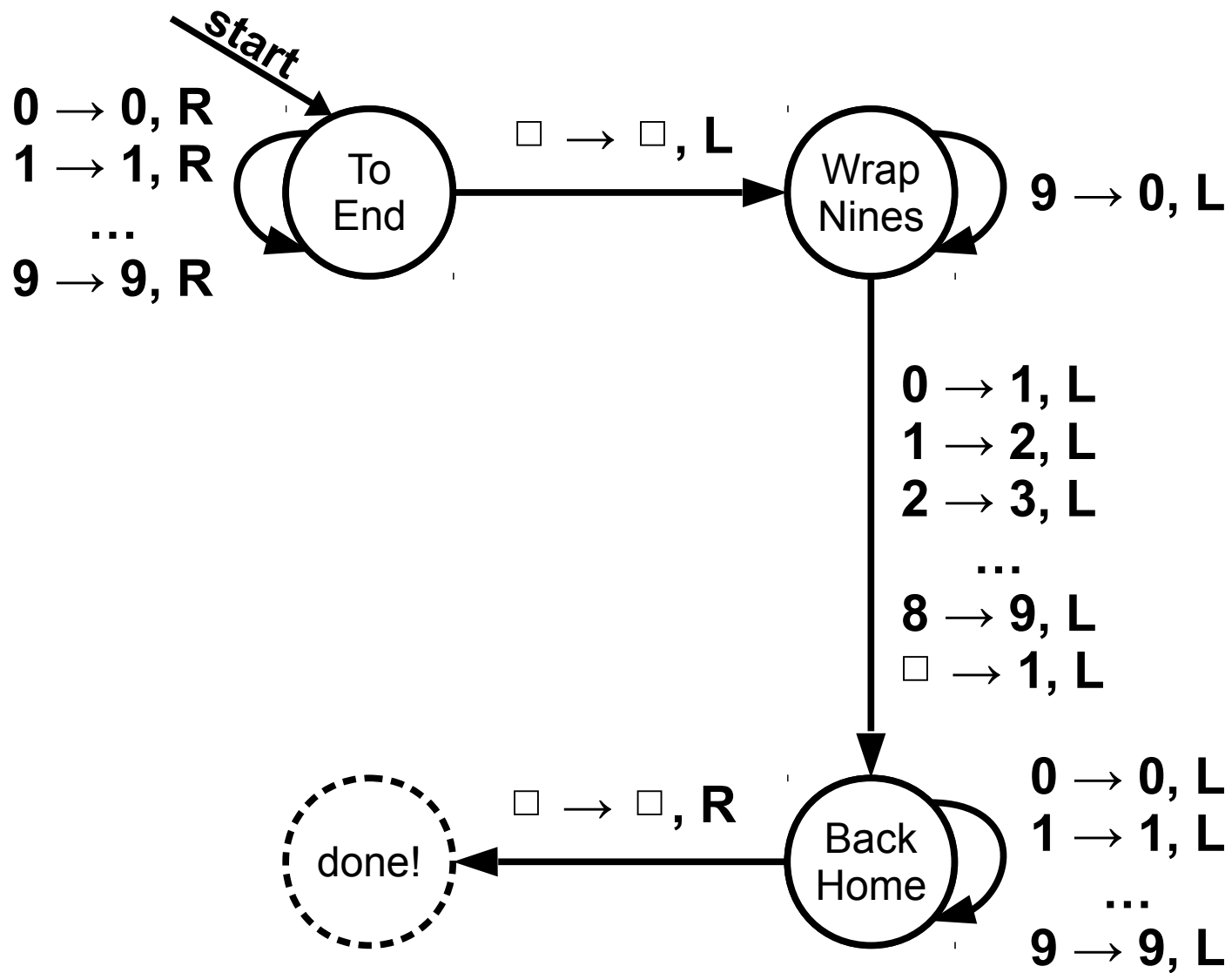






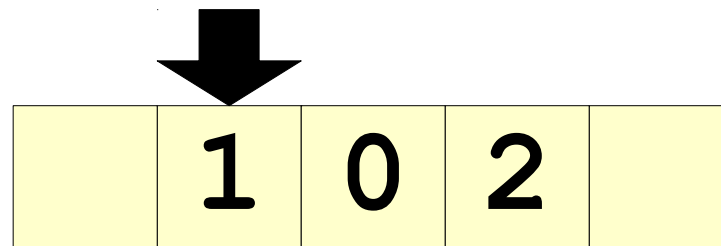






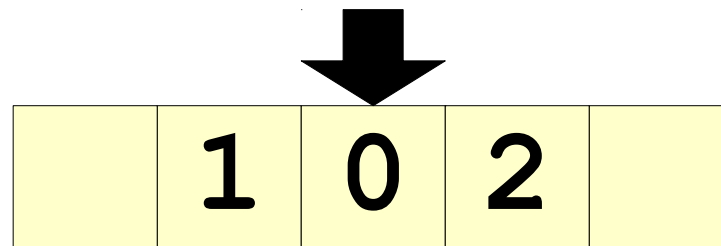
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



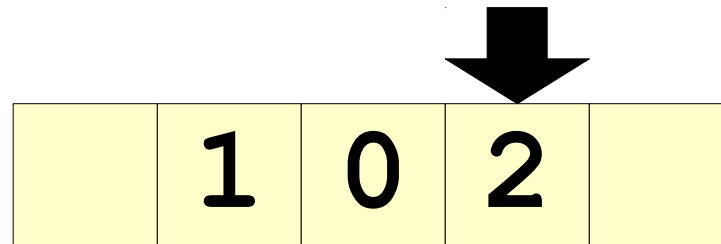
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



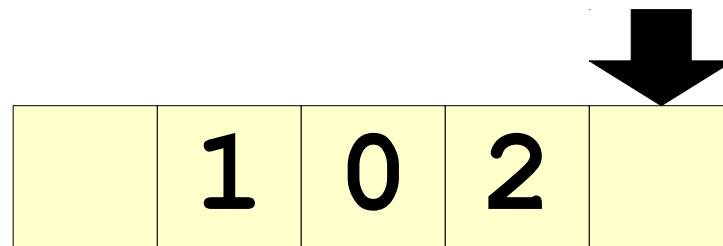
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



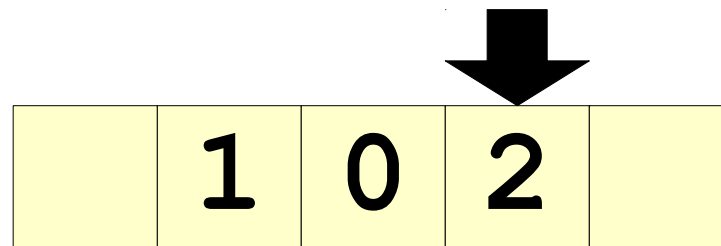
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



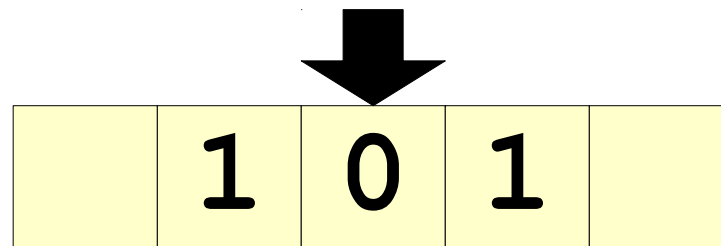
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



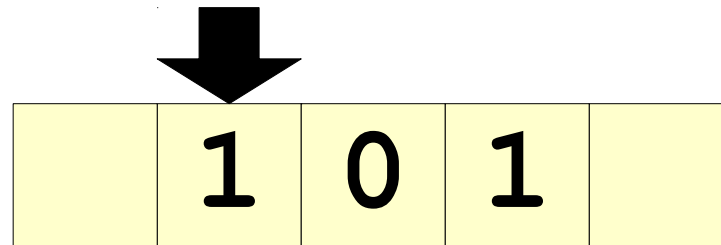
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



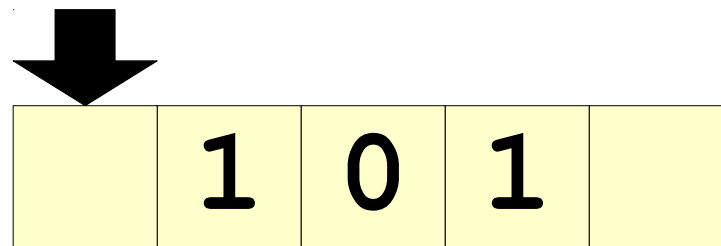
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



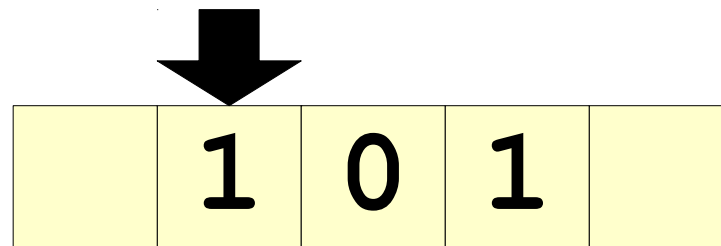
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



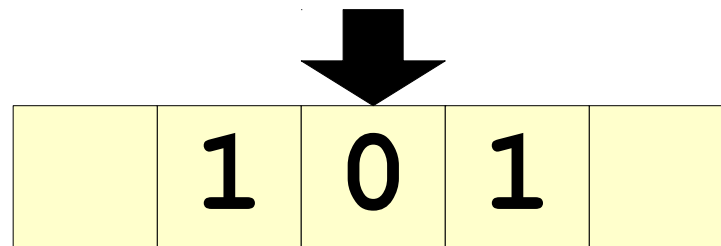
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



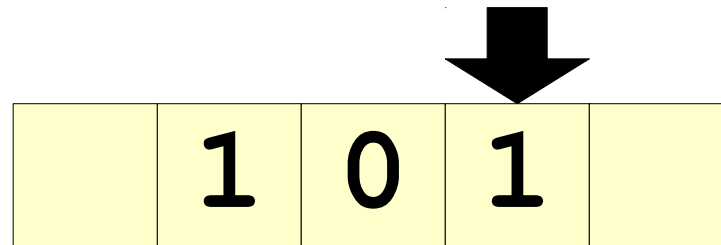
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



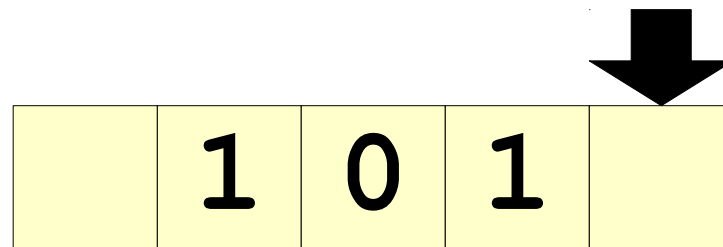
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



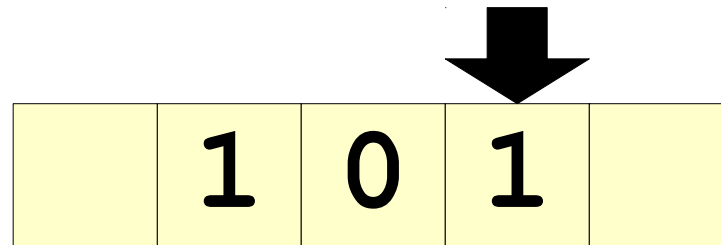
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



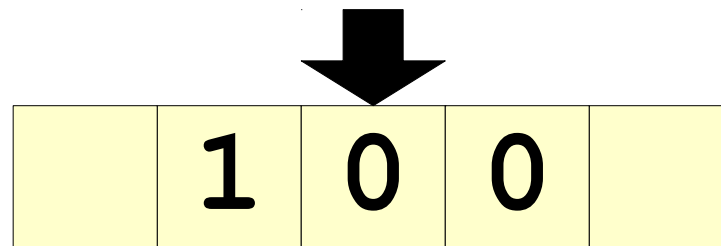
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



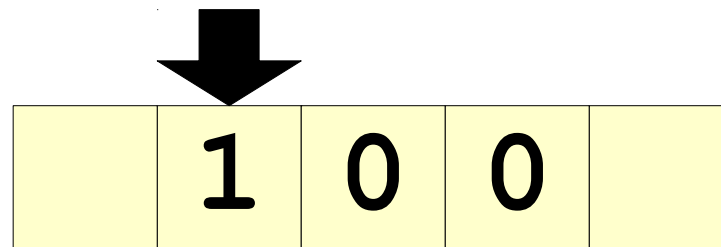
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



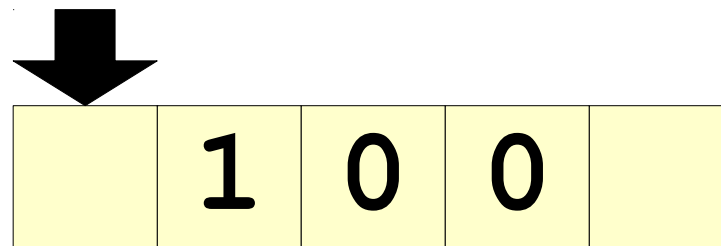
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



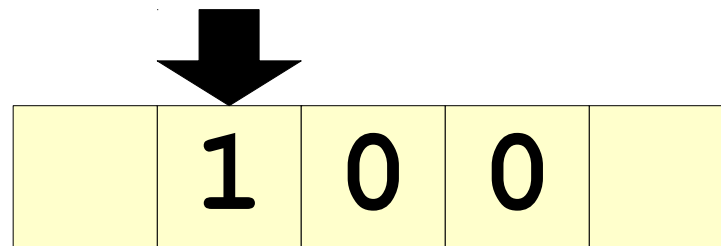
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



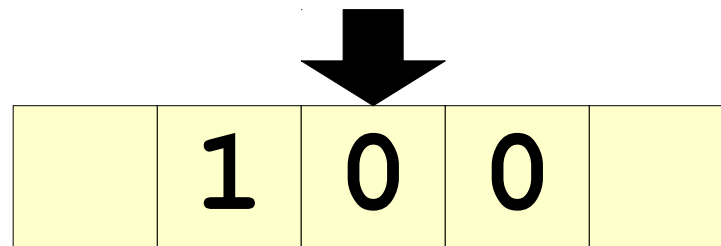
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



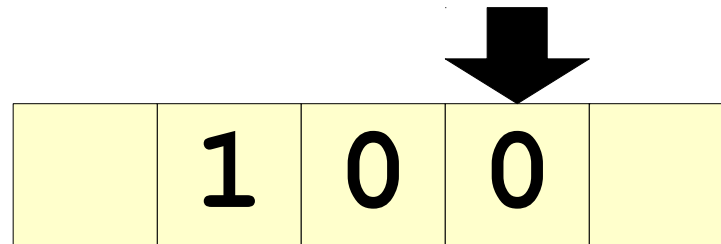
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



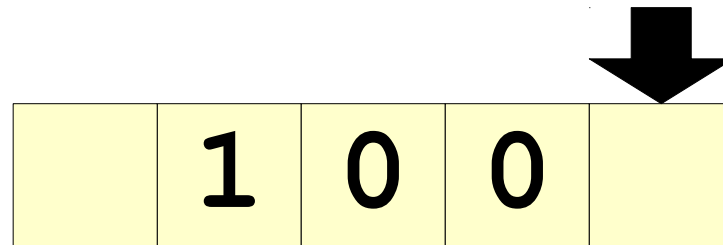
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



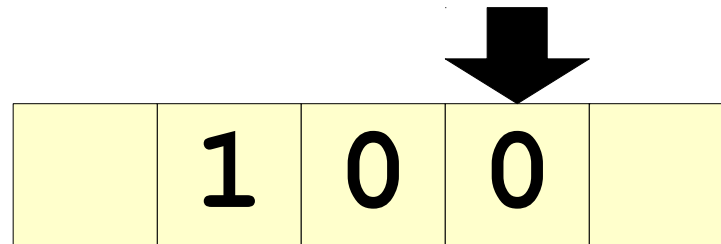
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



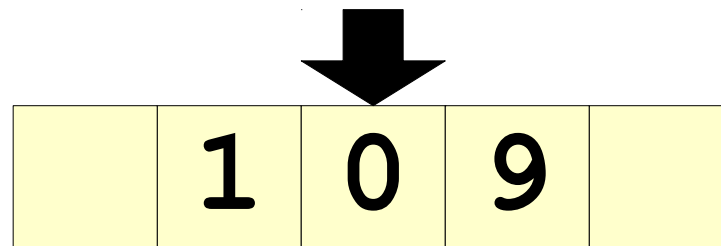
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



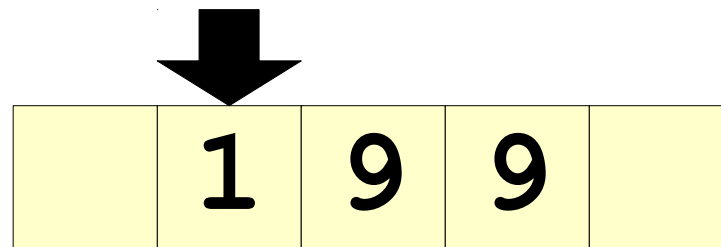
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



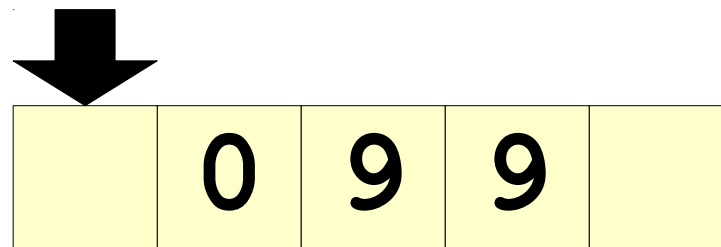
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



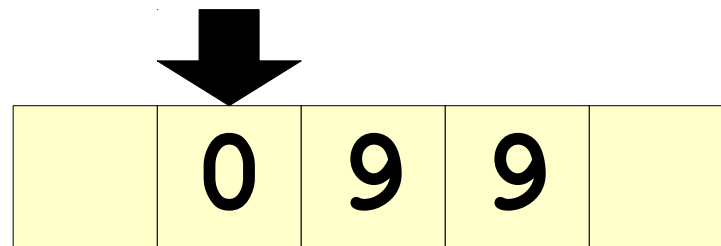
Decrementing Numbers

- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.



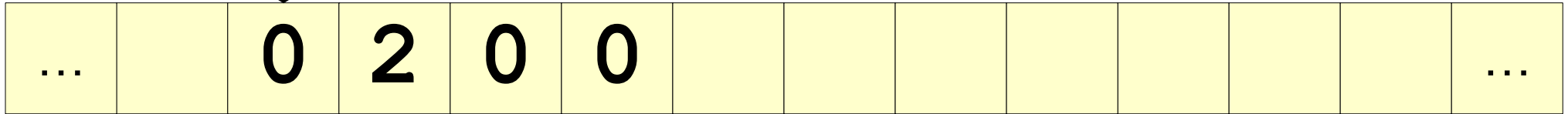
Decrementing Numbers

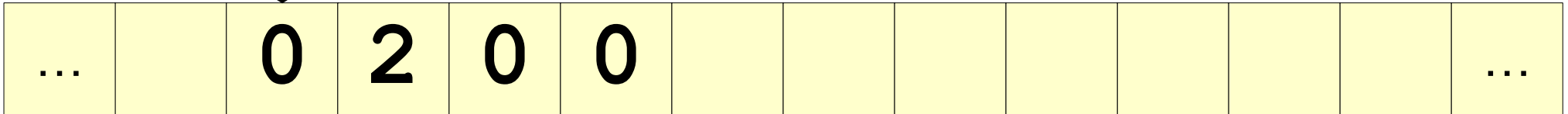
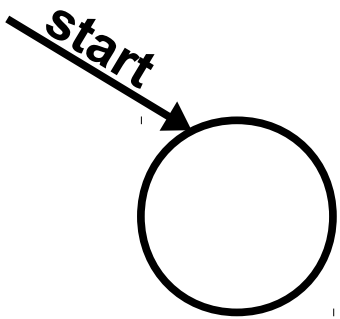
- Now, let's build a TM that decrements a number.
- We'll assume that
 - the tape head points at the start of a number,
 - there is at least one blank on each side of the number.
- The tape head will end at the start of the number after decrementing it.
- If the number is 0, then the subroutine should somehow signal this rather than making the number negative.

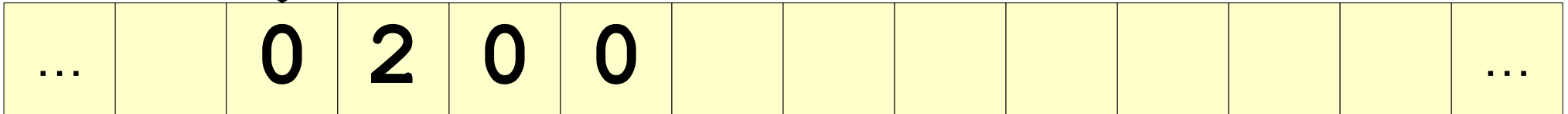
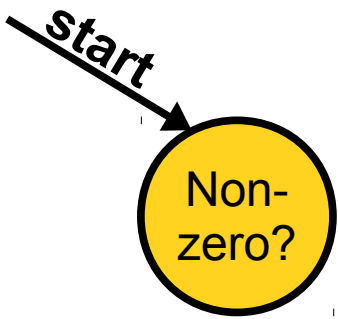


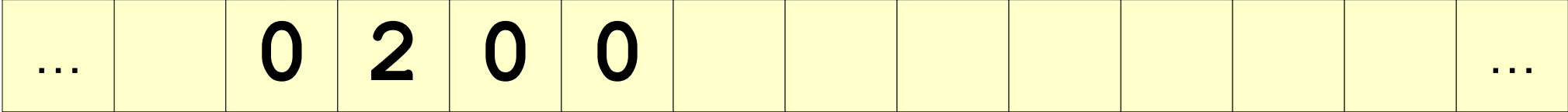
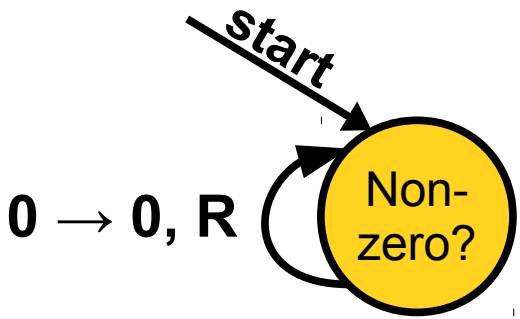
Decrementing Numbers

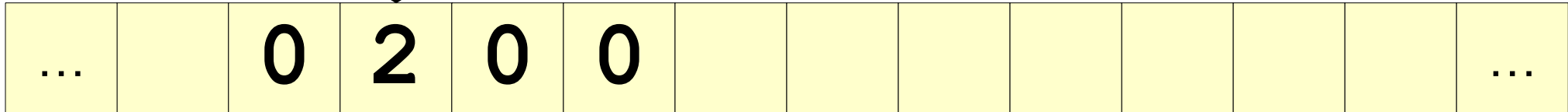
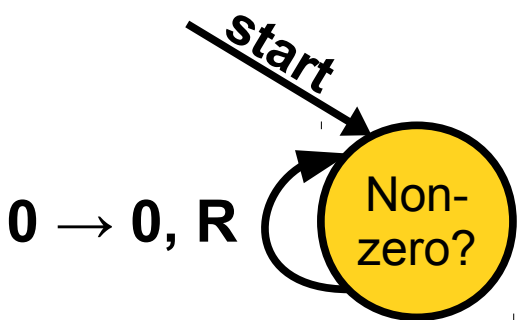
```
decrement(num) {  
    go to the end of the number;  
    if (every digit was 0) {  
        signal that we're done;  
    }  
    while (the current digit is 0) {  
        set the current digit to 9;  
        back up one digit;  
    }  
    decrement the current digit;  
    go to the start of the number;  
}
```

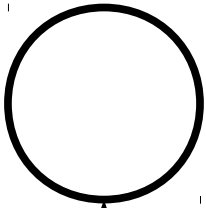




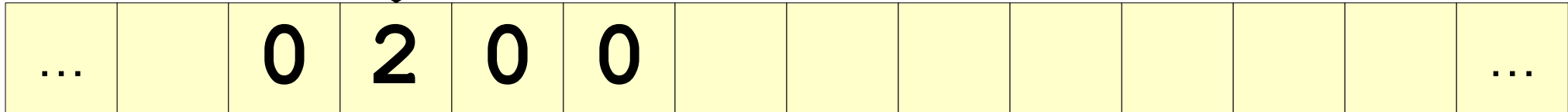
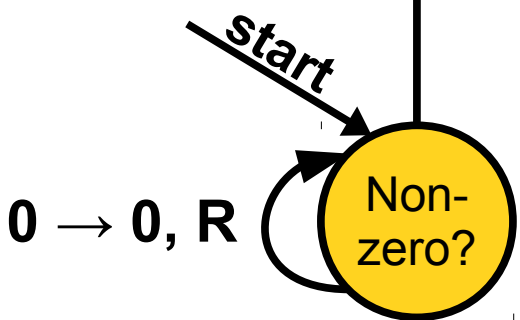


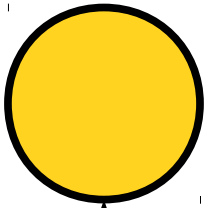




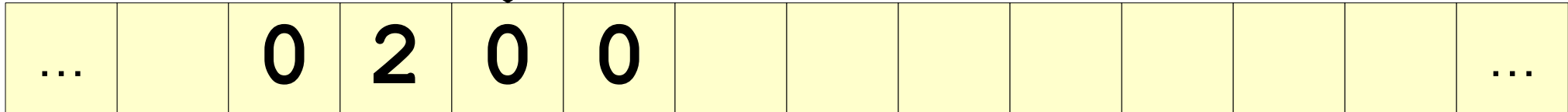
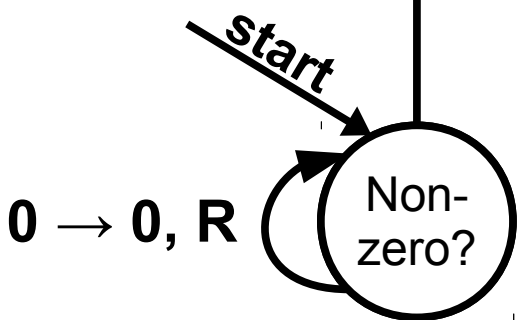


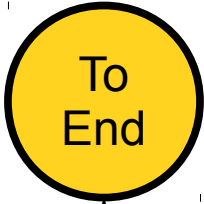
1 → 1, R
2 → 2, R
...
9 → 9, R





1 → 1, R
2 → 2, R
...
9 → 9, R





1 → 1, R

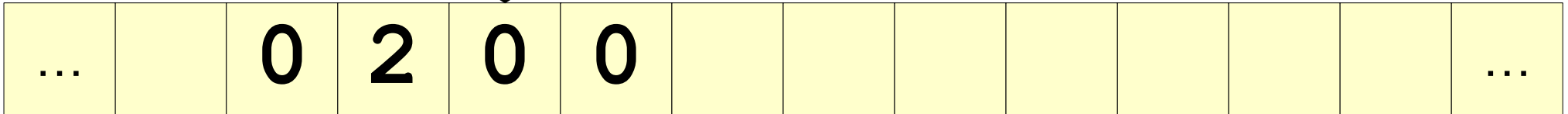
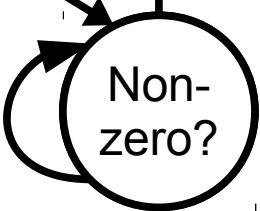
2 → 2, R

...

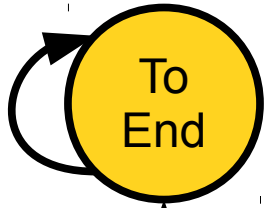
9 → 9, R

start

0 → 0, R



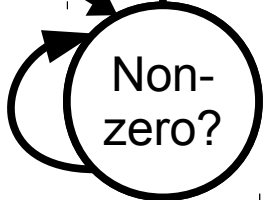
0 → **0**, **R**
1 → **1**, **R**
...
9 → **9**, **R**



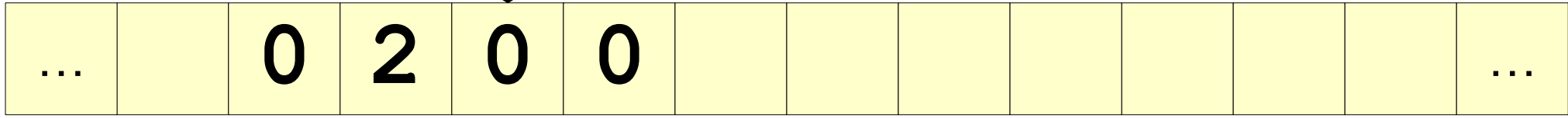
1 → **1**, **R**
2 → **2**, **R**
...
9 → **9**, **R**



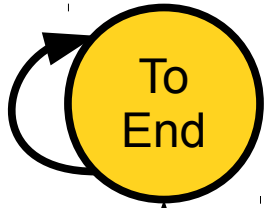
0 → **0**, **R**



start



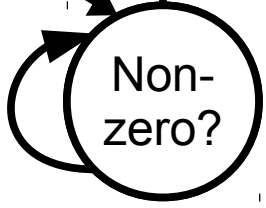
0 → **0**, **R**
1 → **1**, **R**
...
9 → **9**, **R**



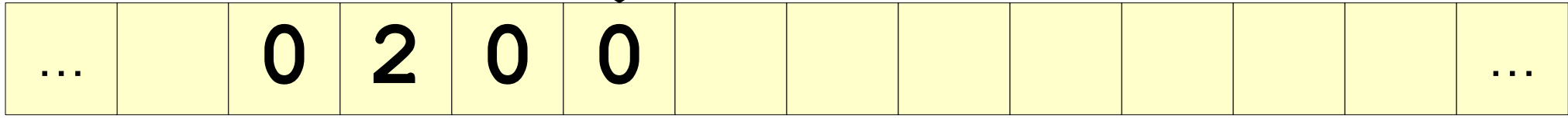
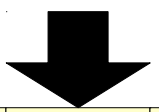
1 → **1**, **R**
2 → **2**, **R**
...
9 → **9**, **R**

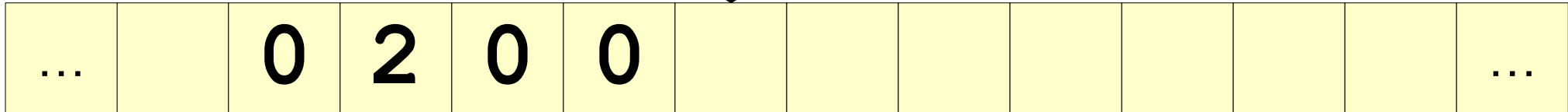
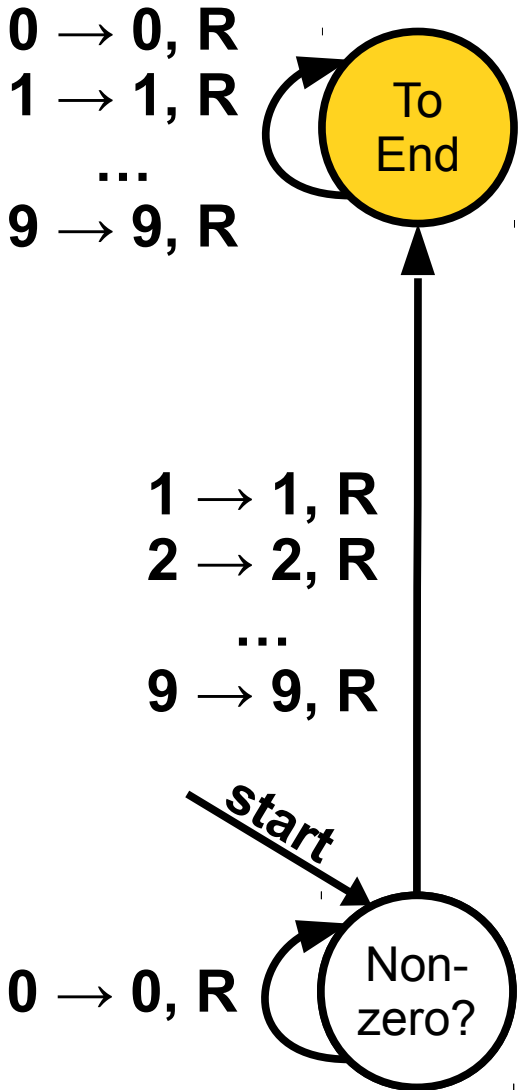


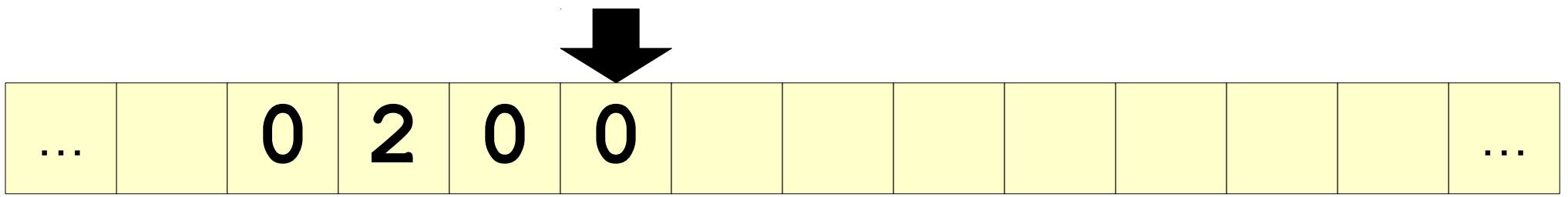
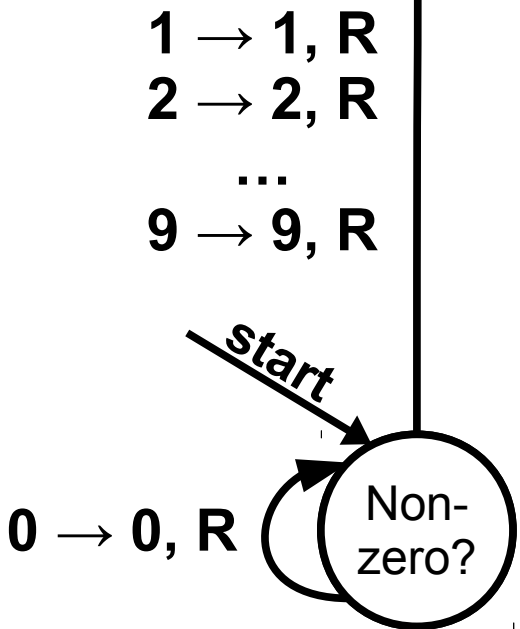
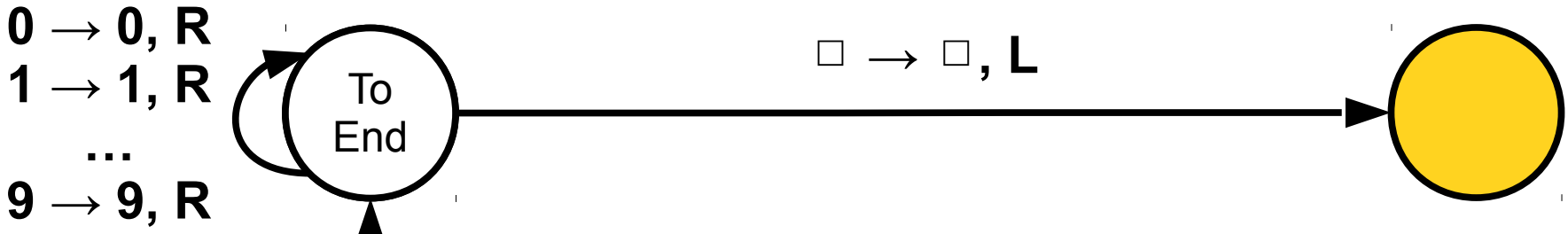
0 → **0**, **R**

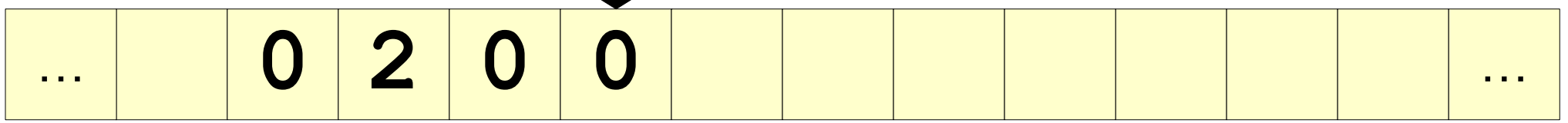
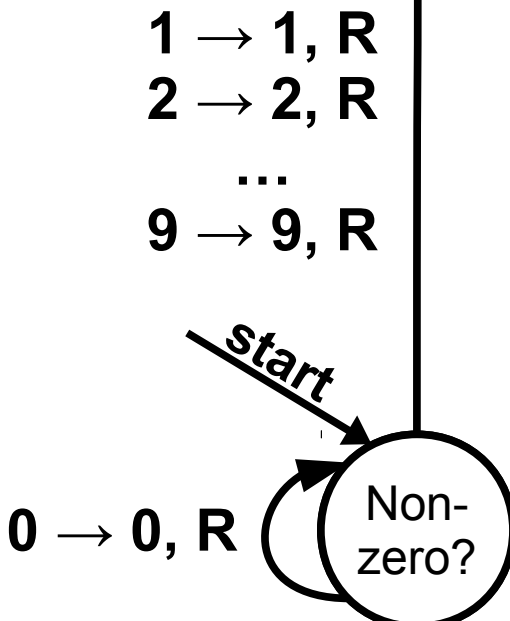
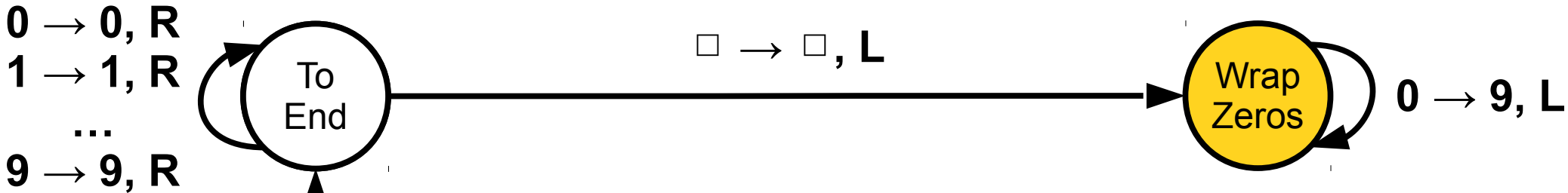


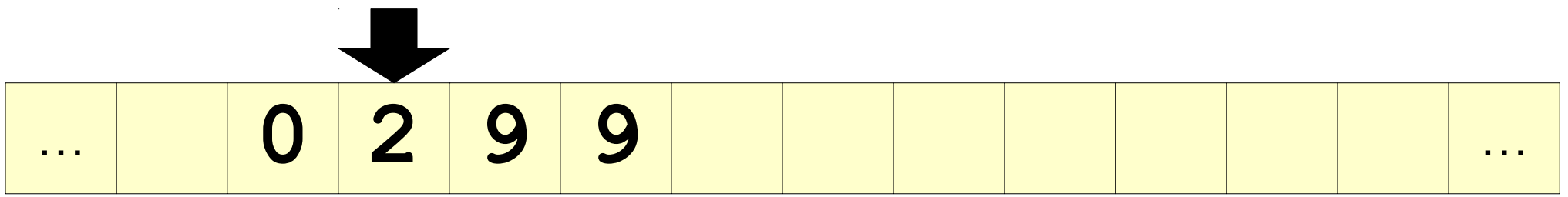
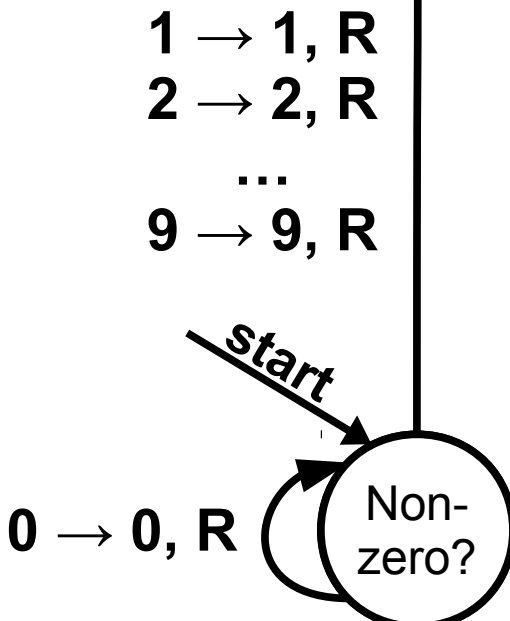
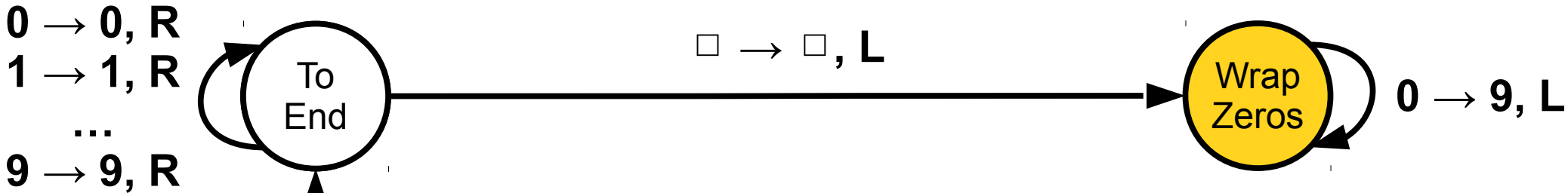
start

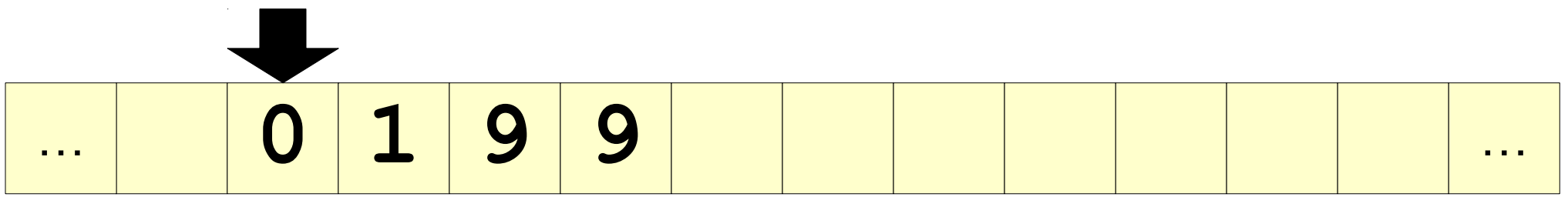
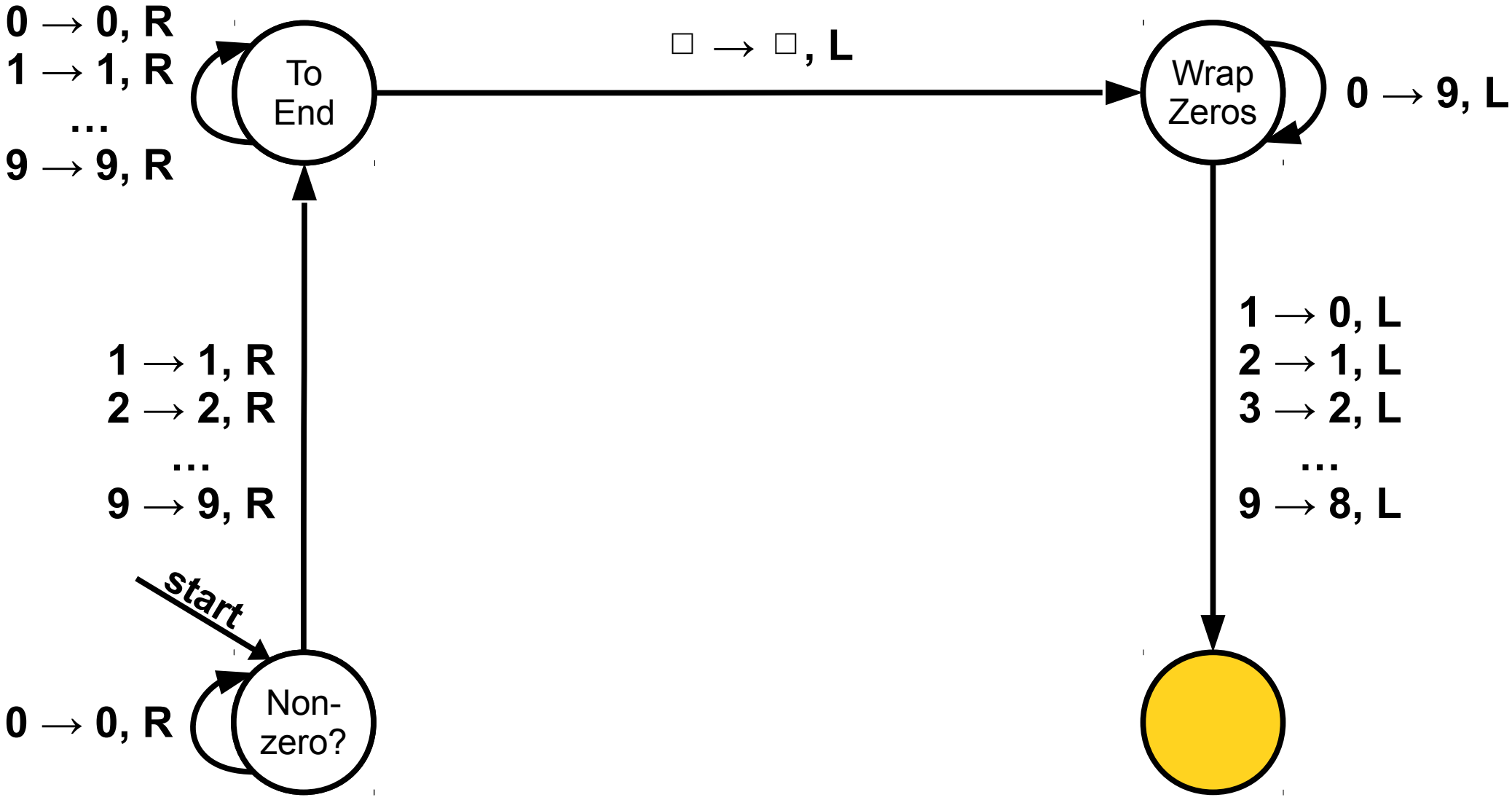


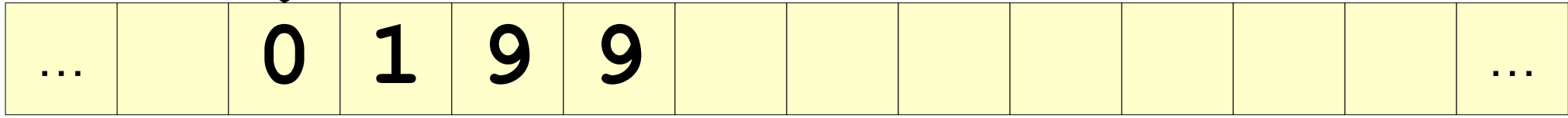
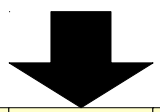
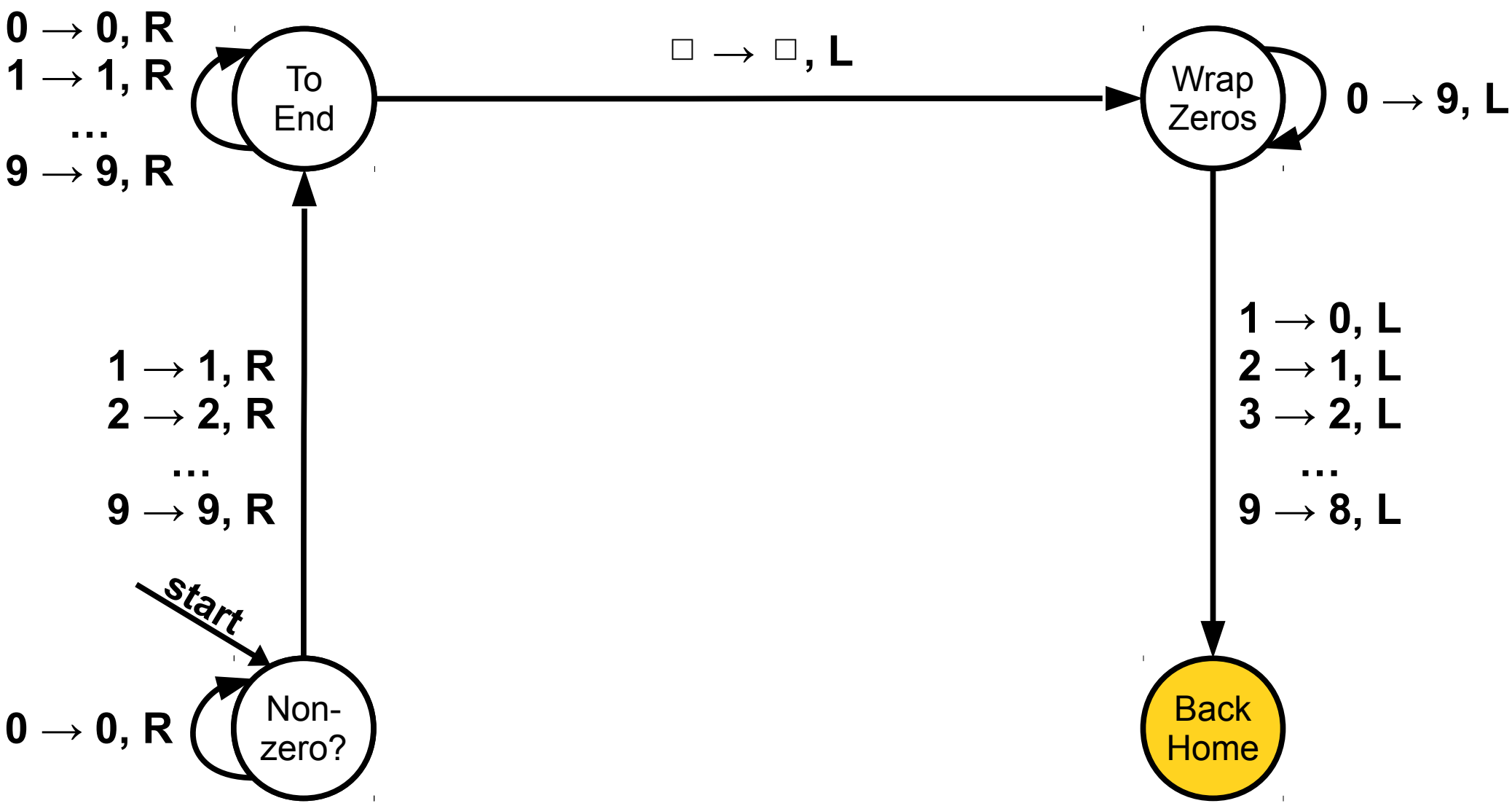


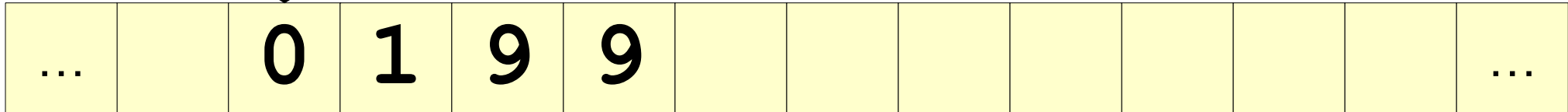
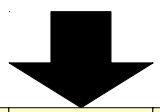
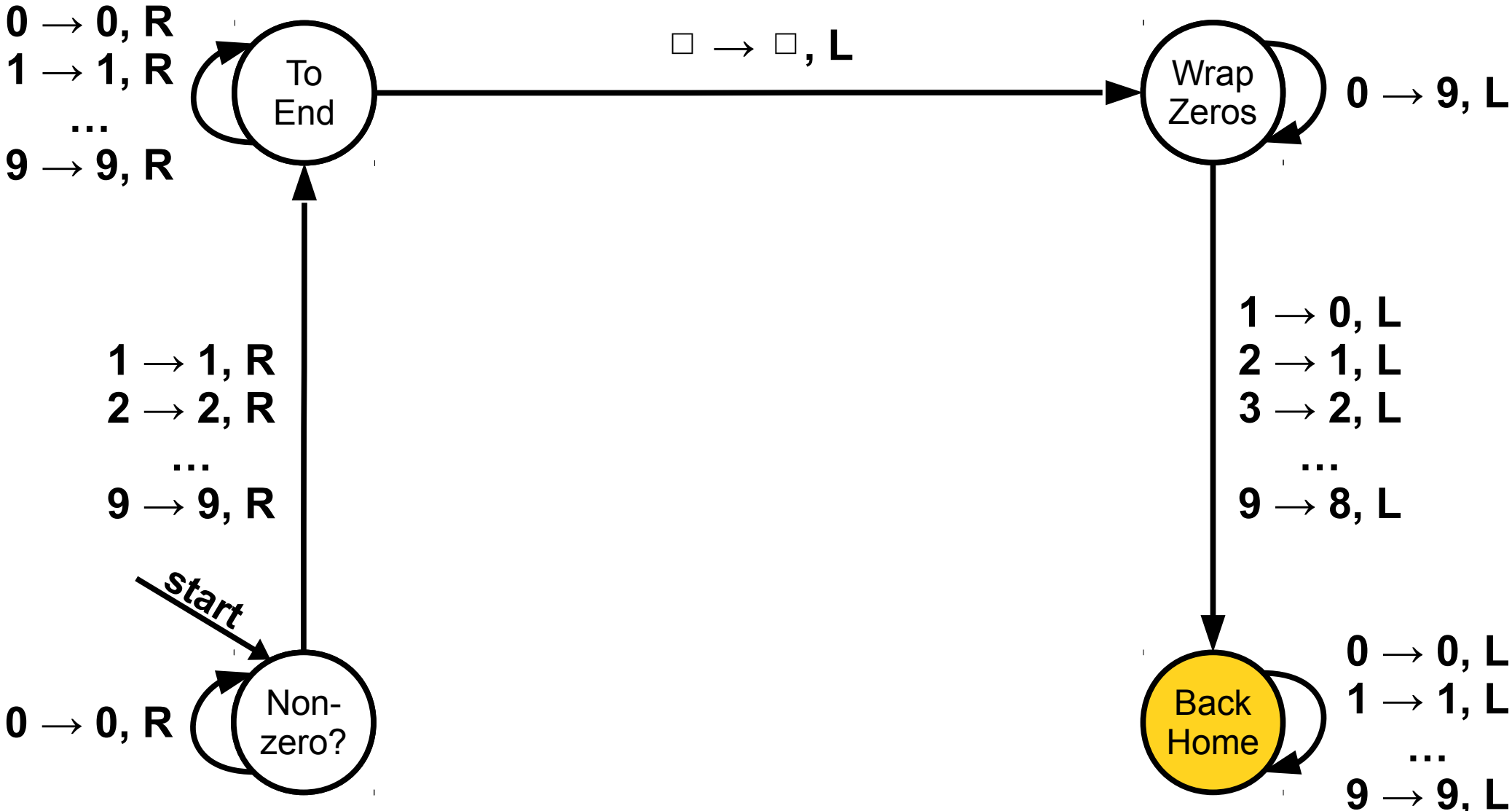


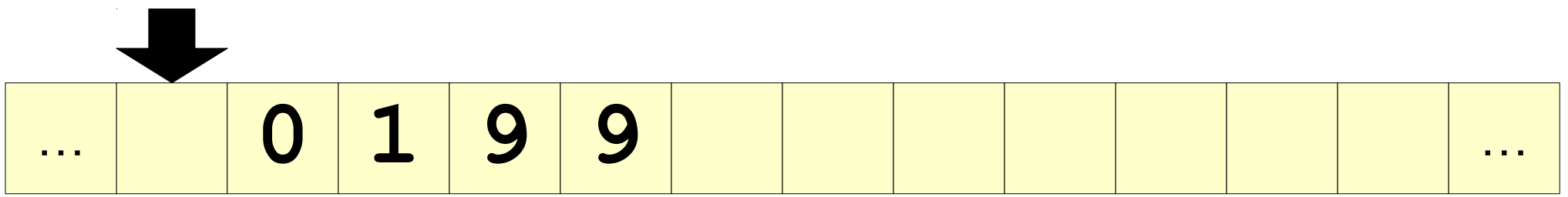
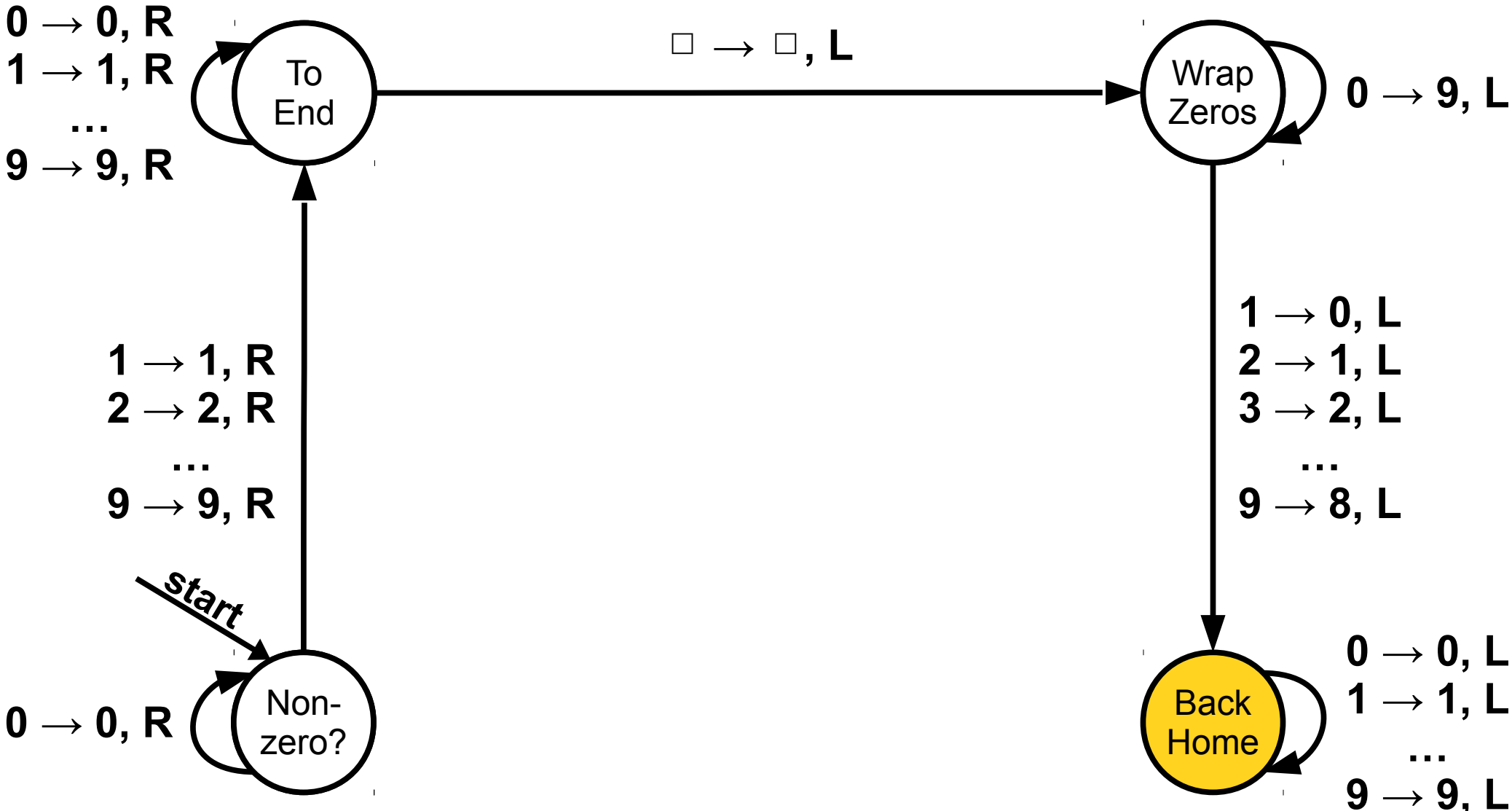


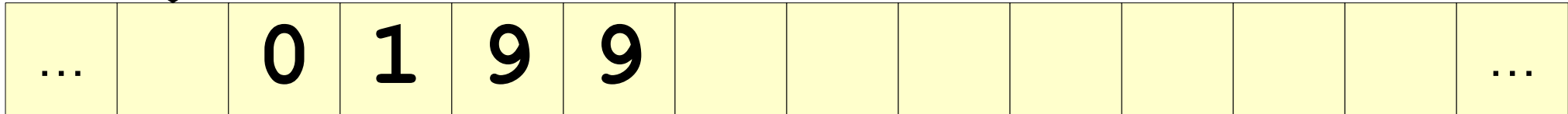
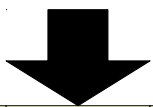
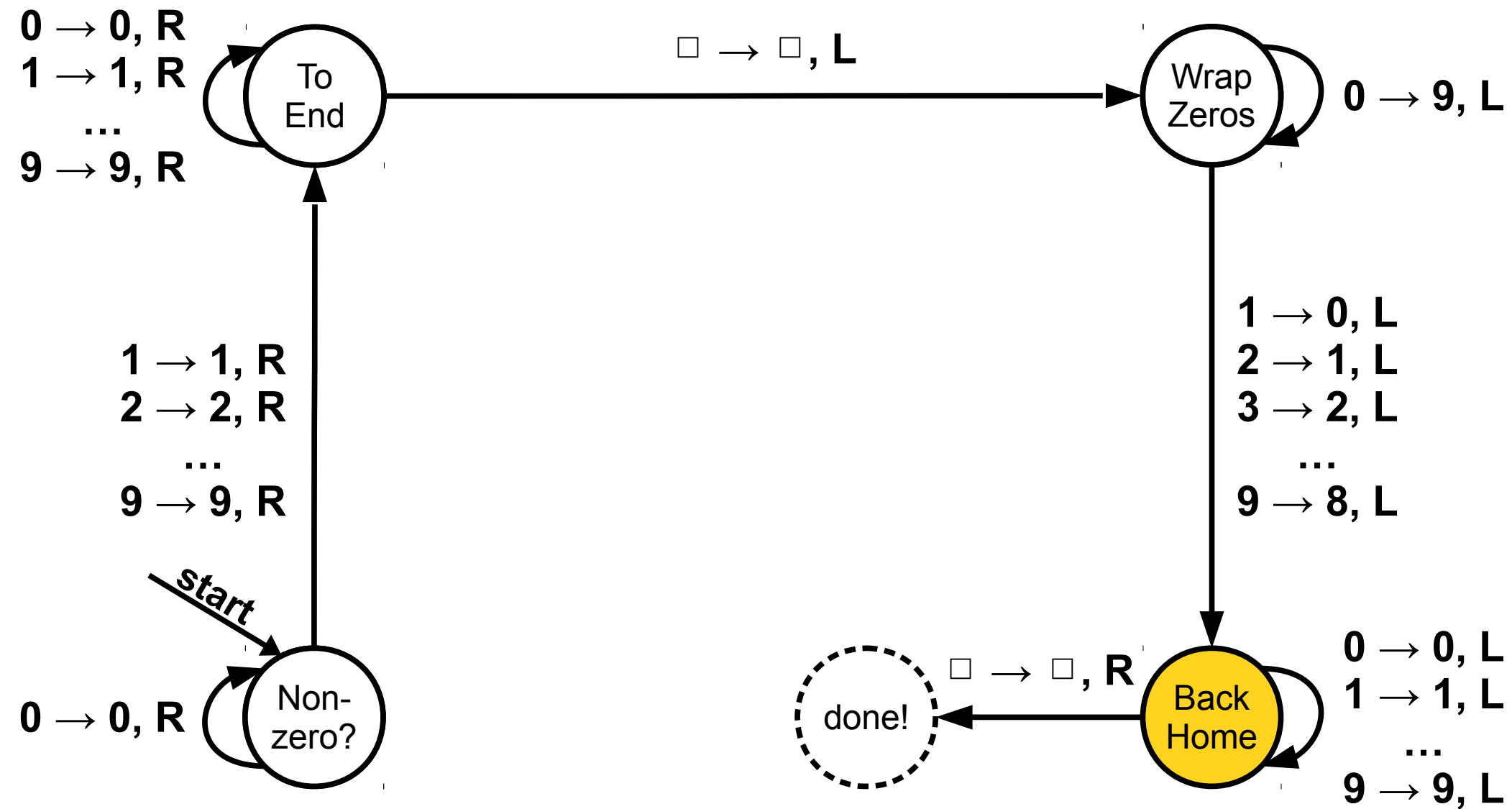


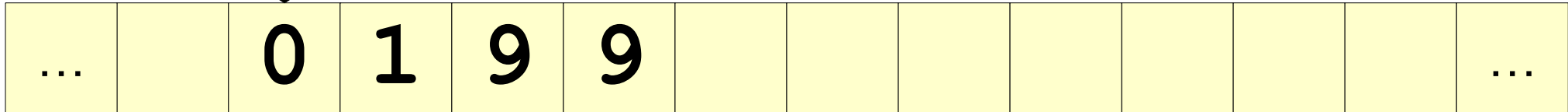
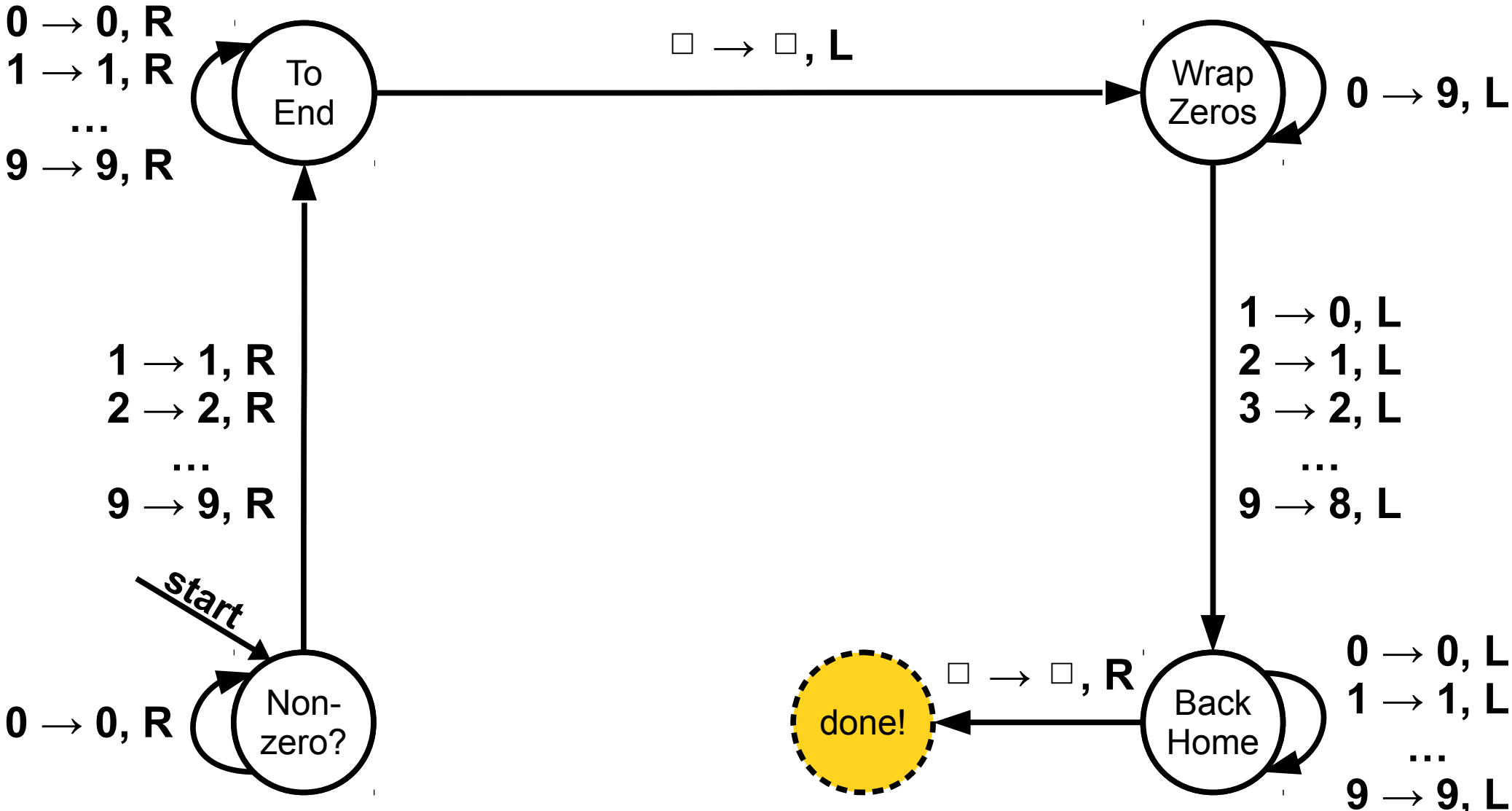


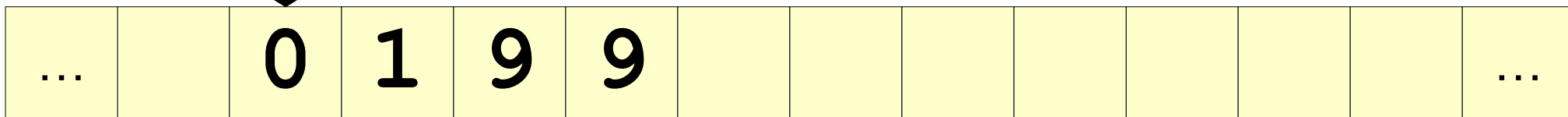
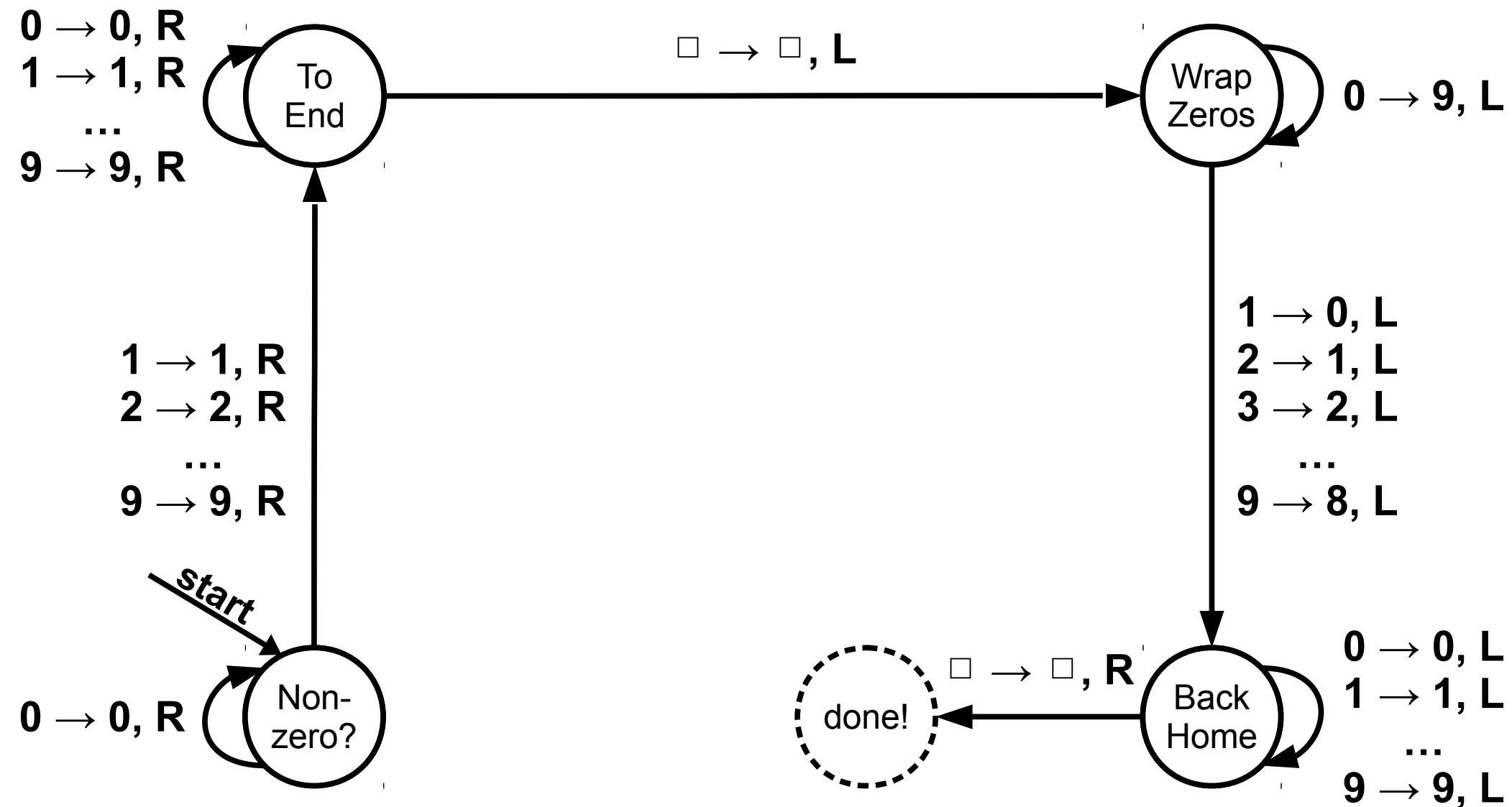


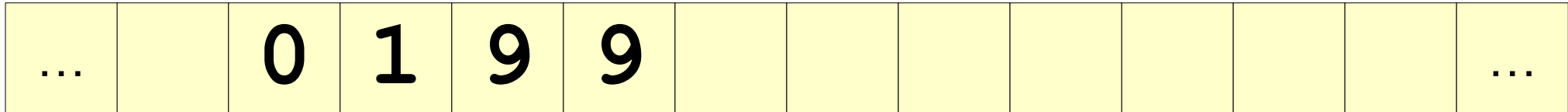
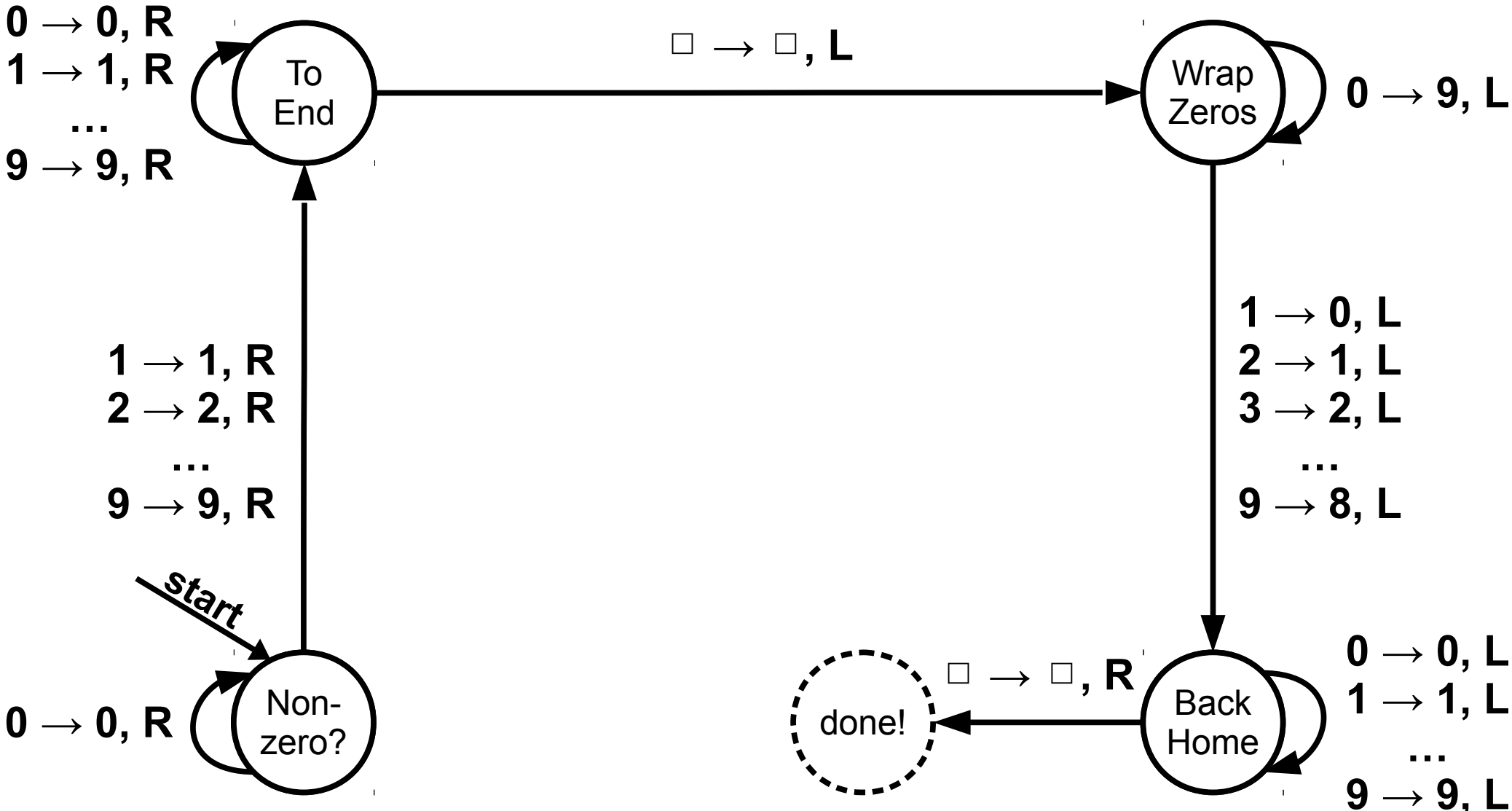


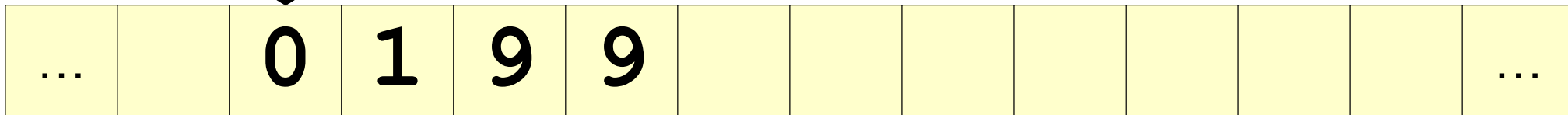
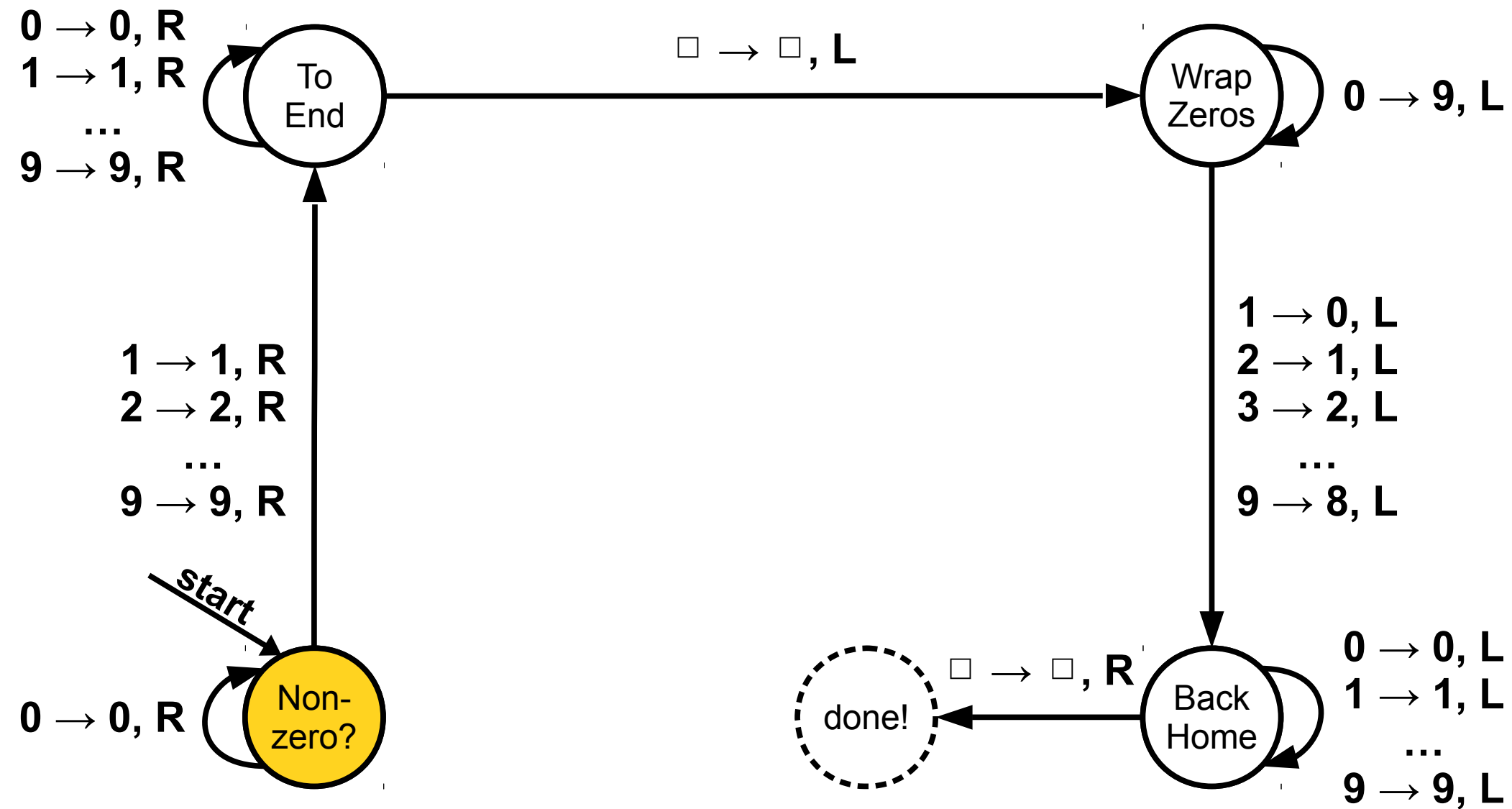


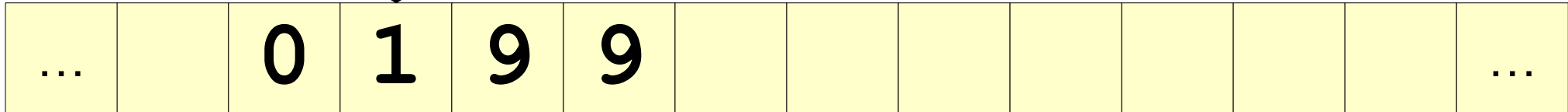
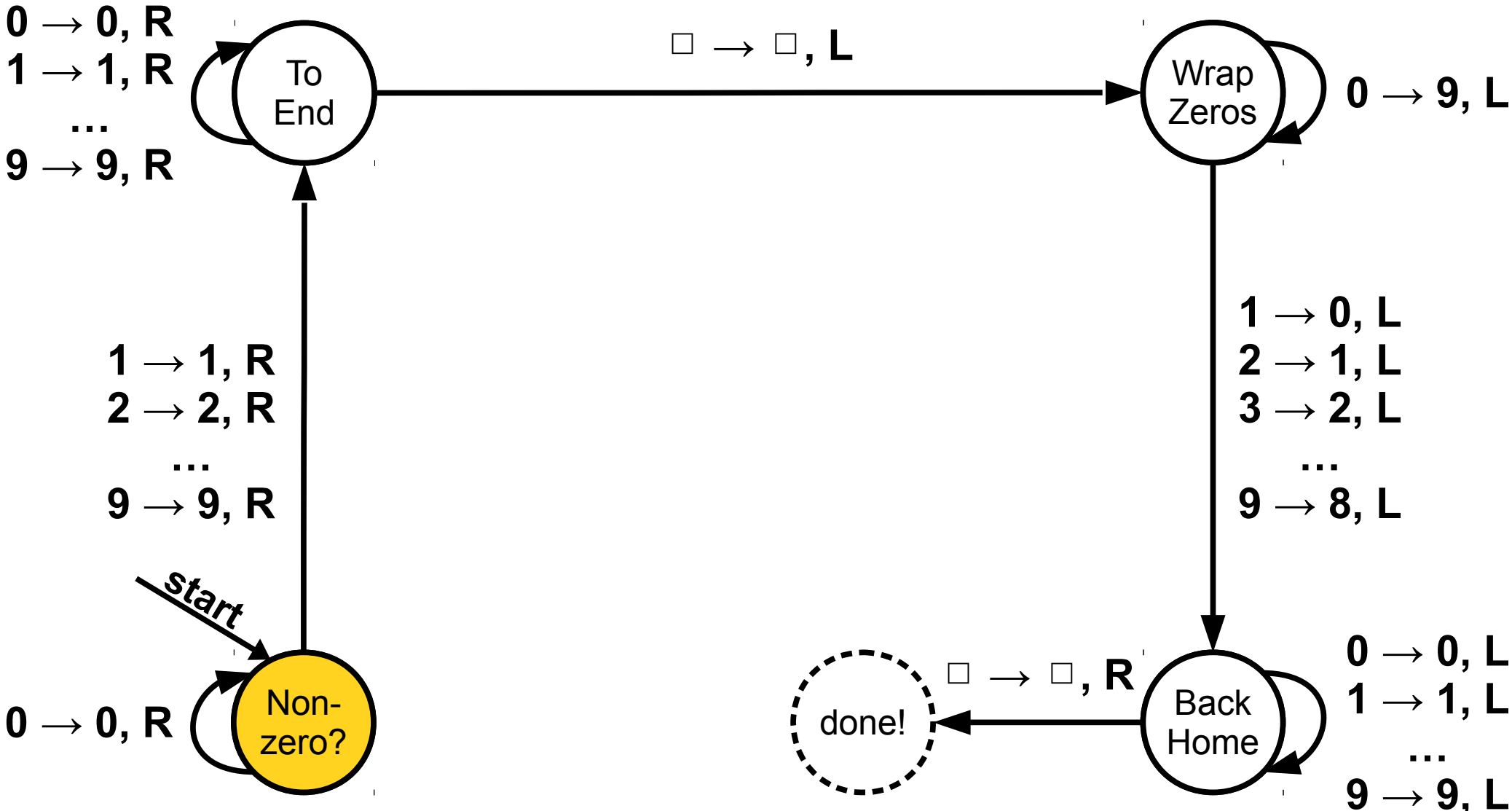


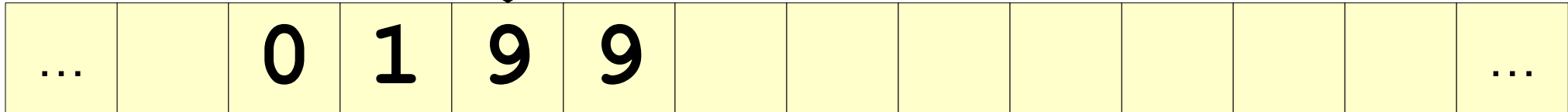
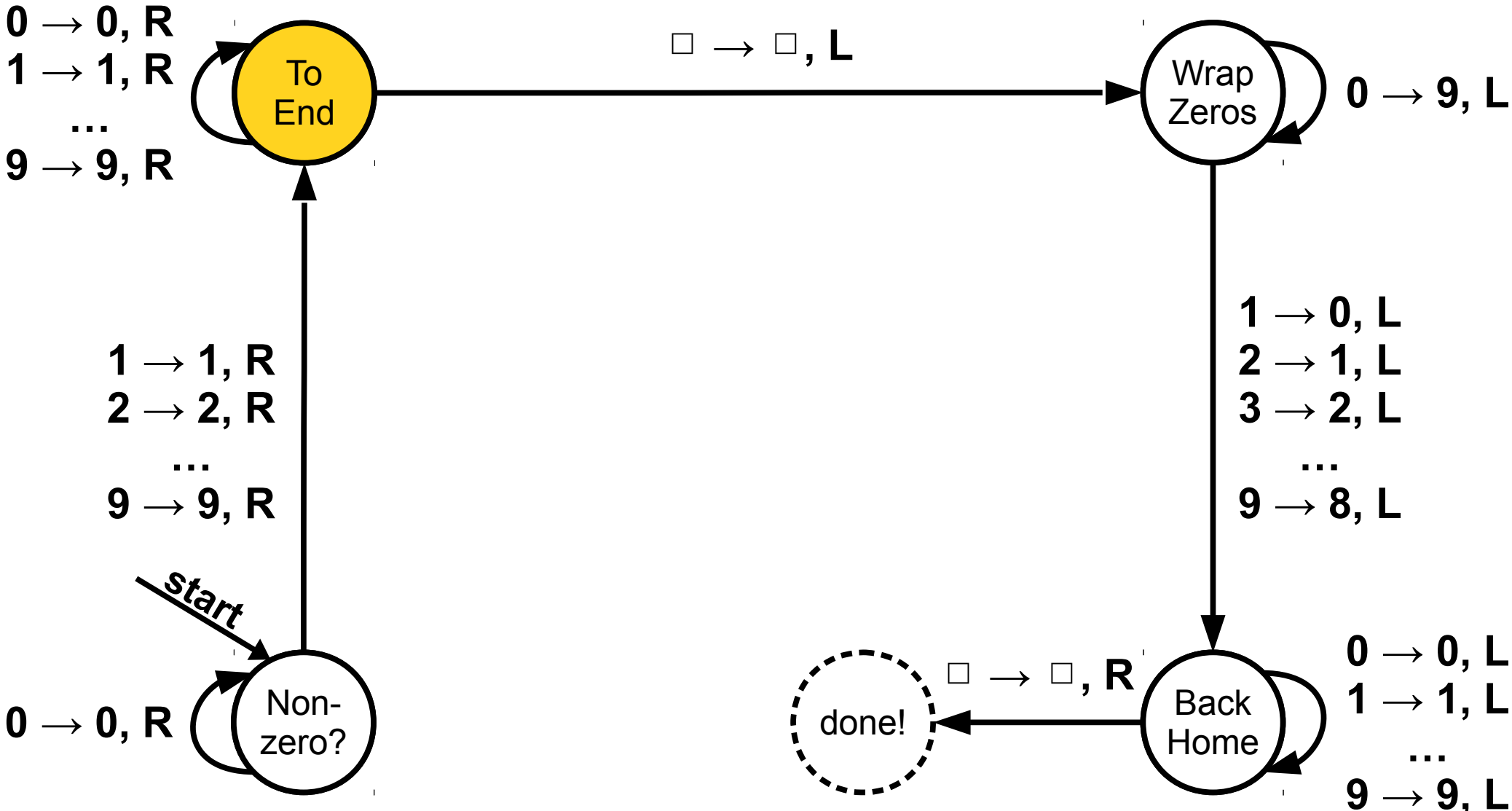


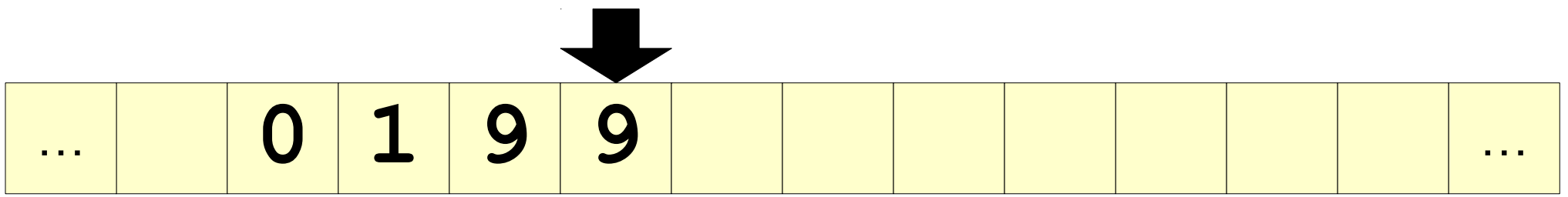
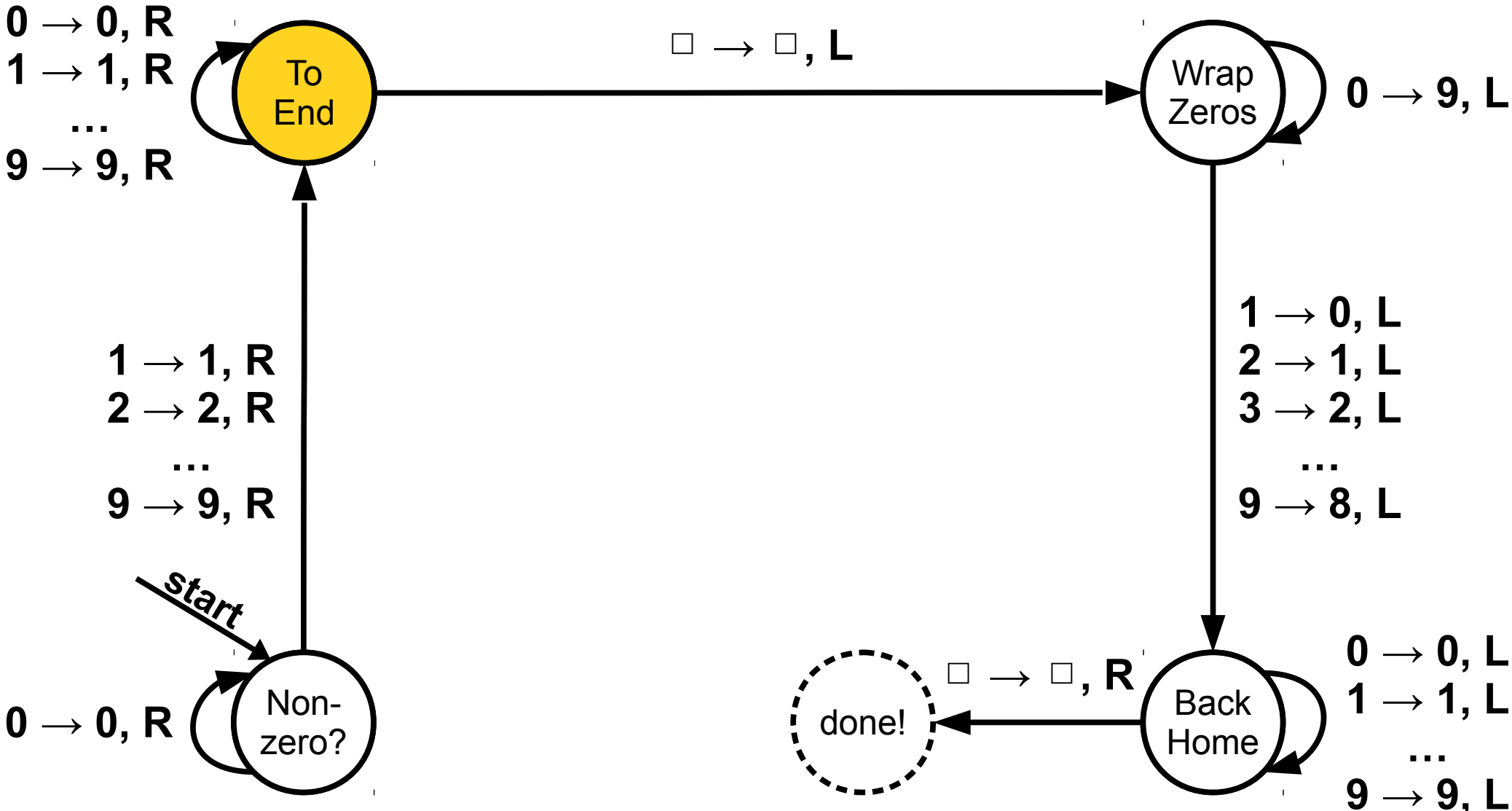


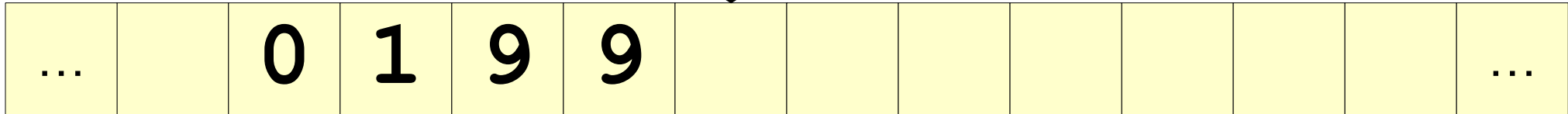
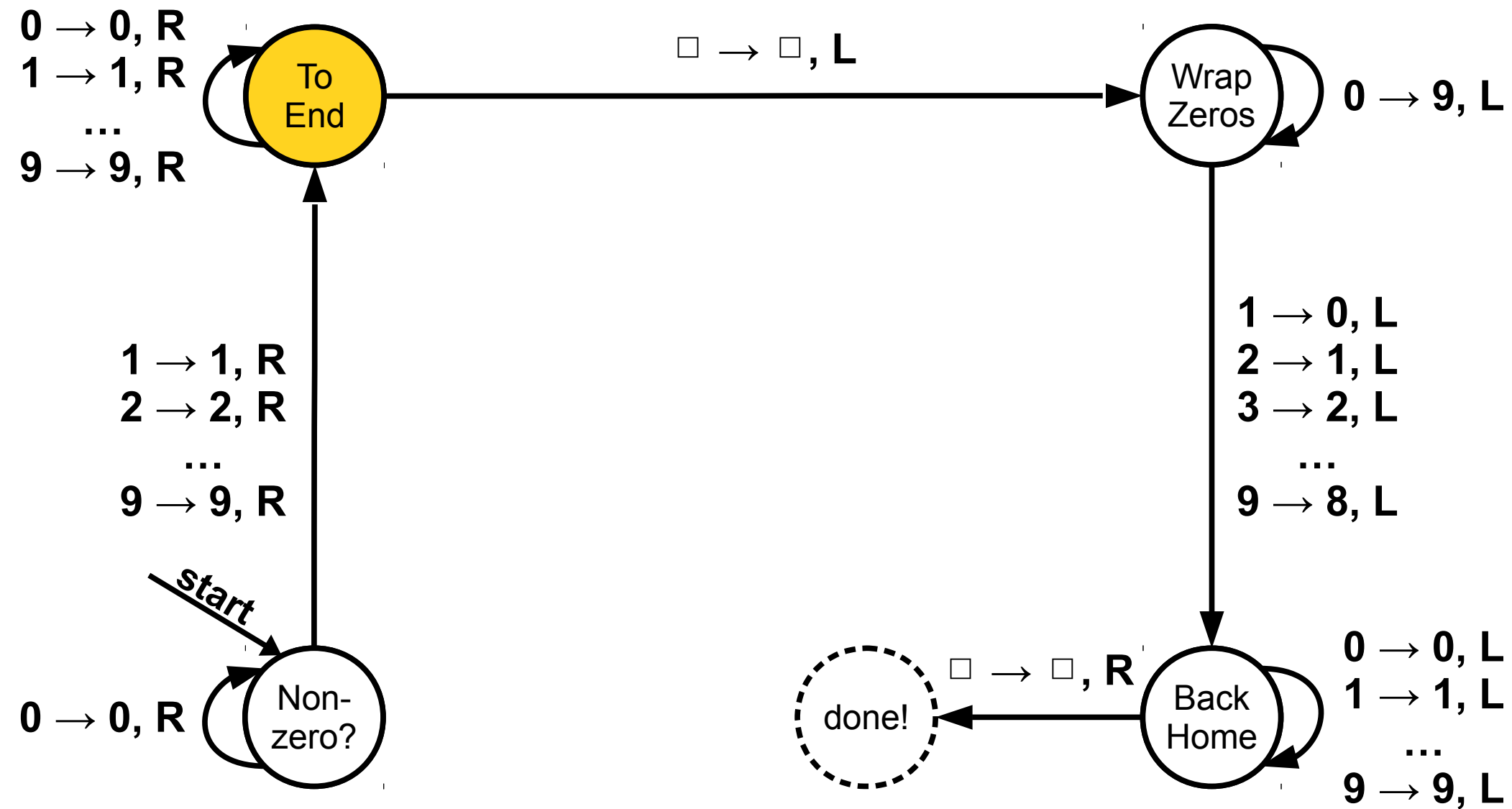


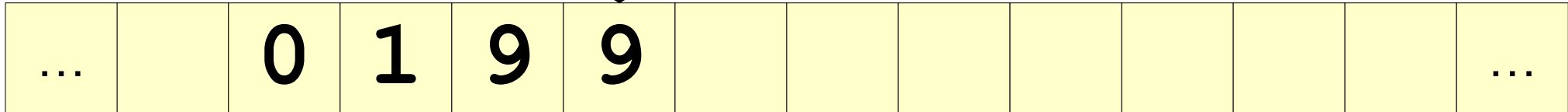
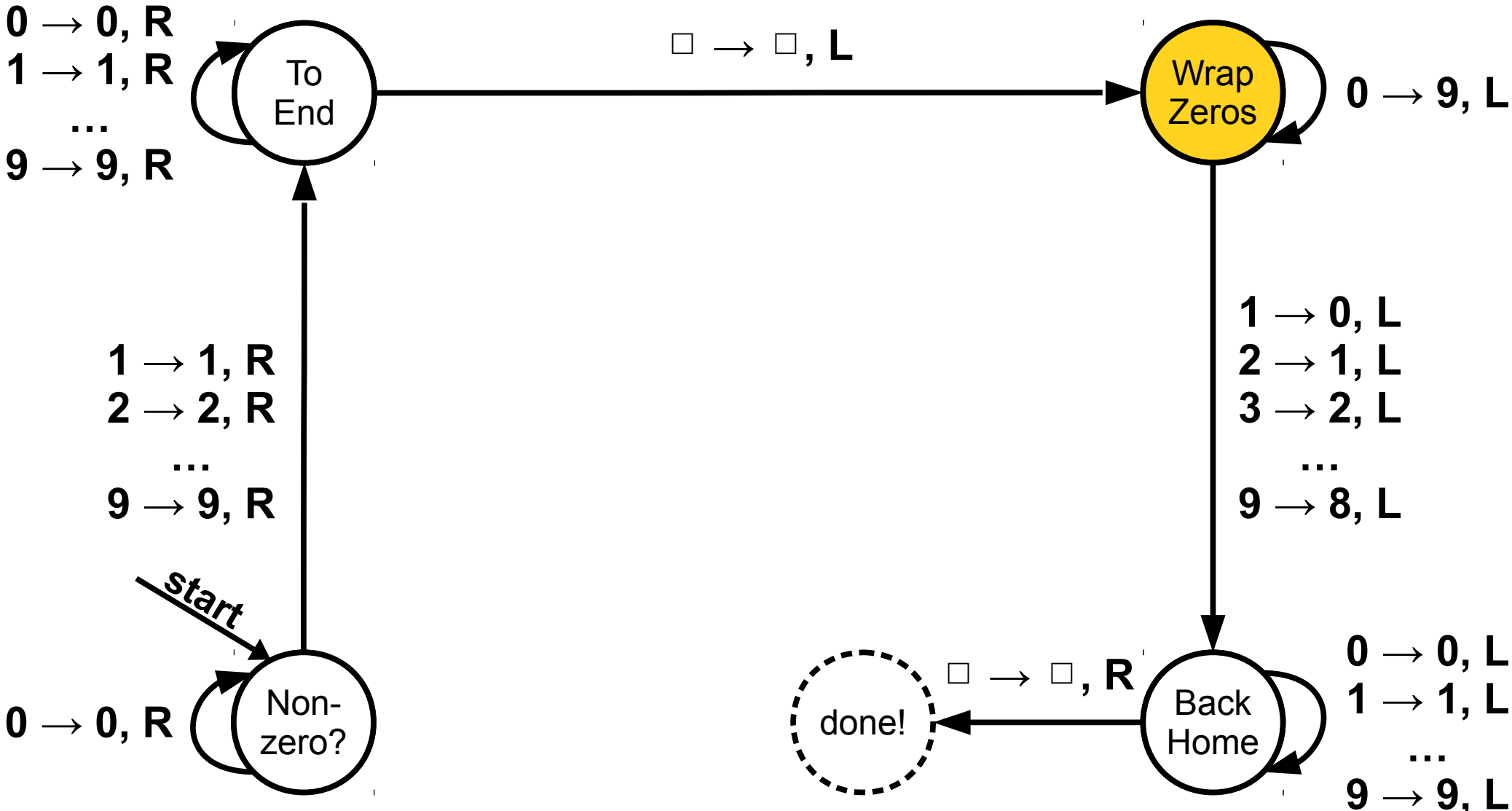


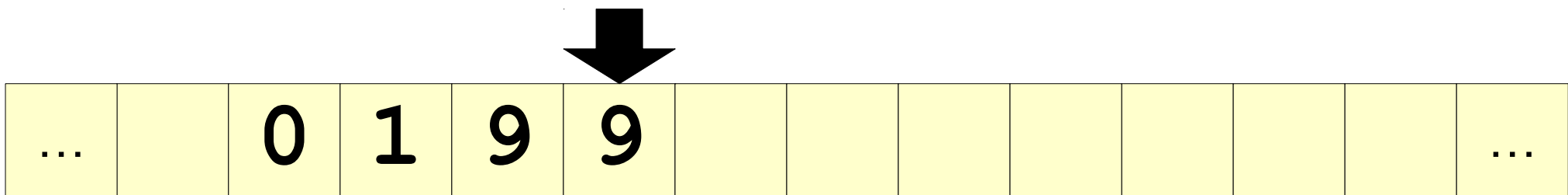
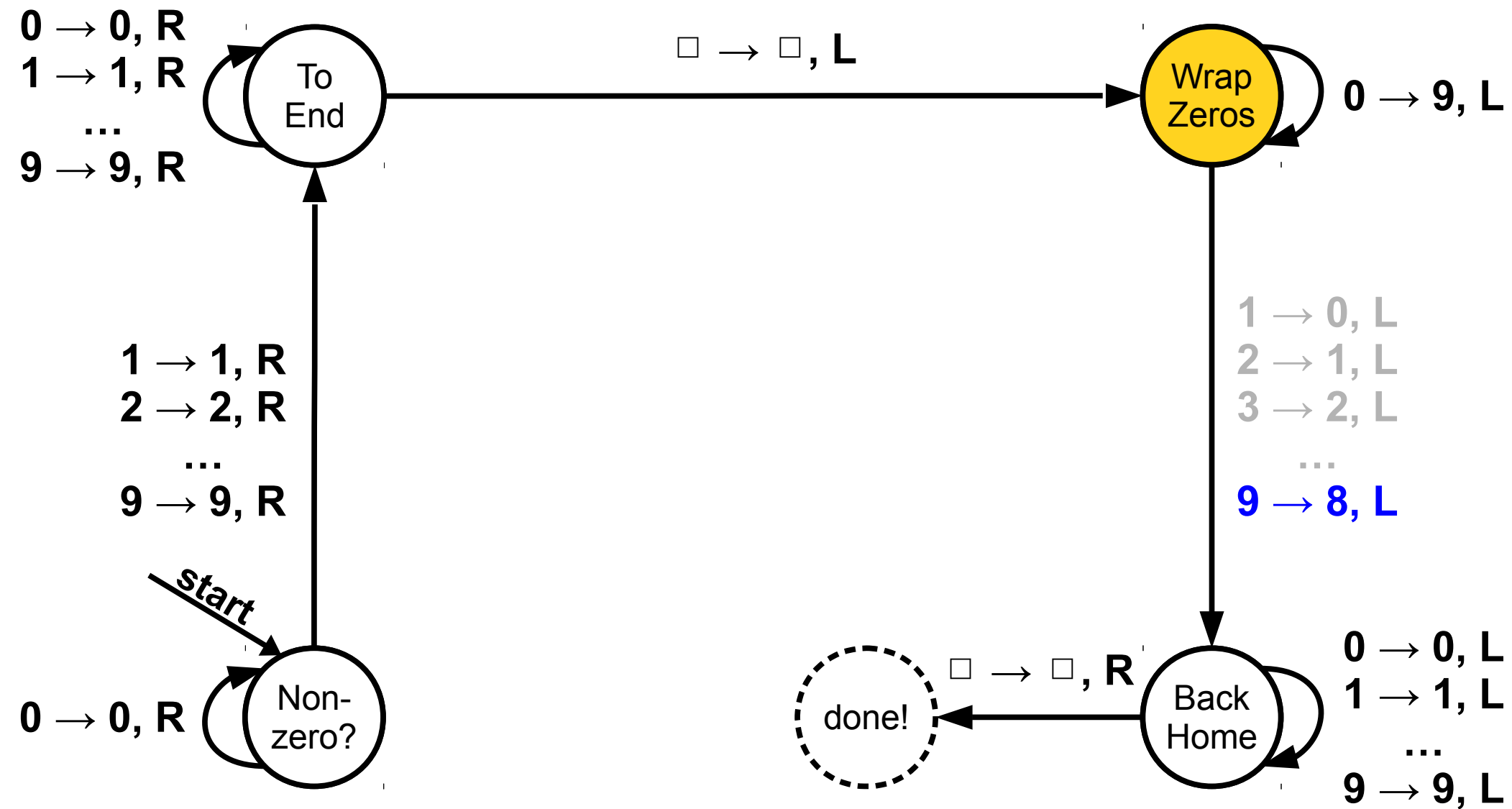


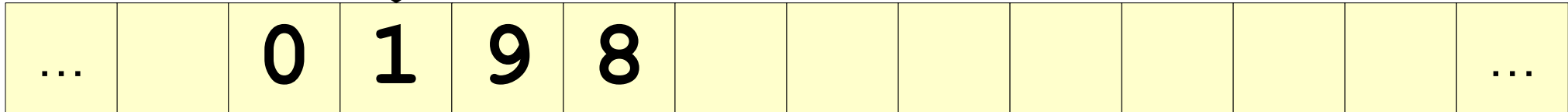
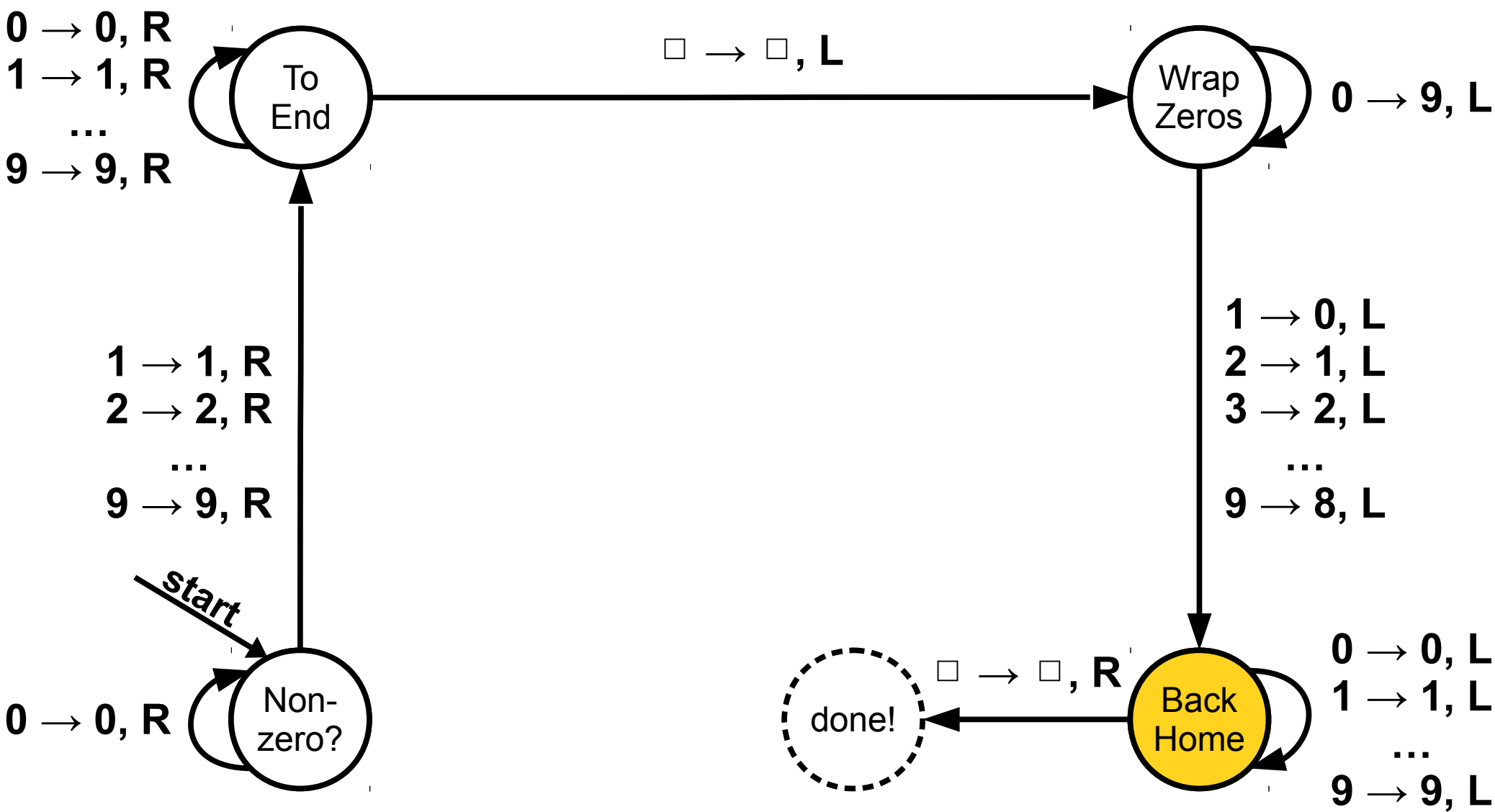


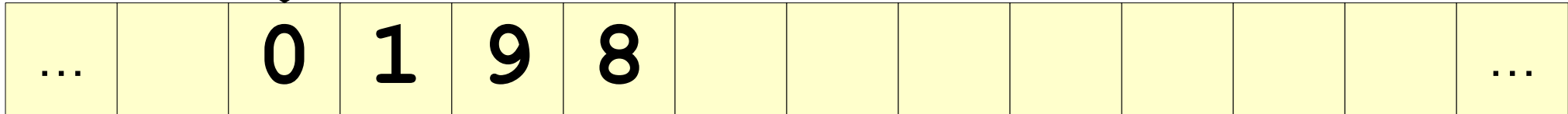
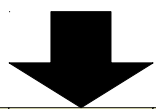
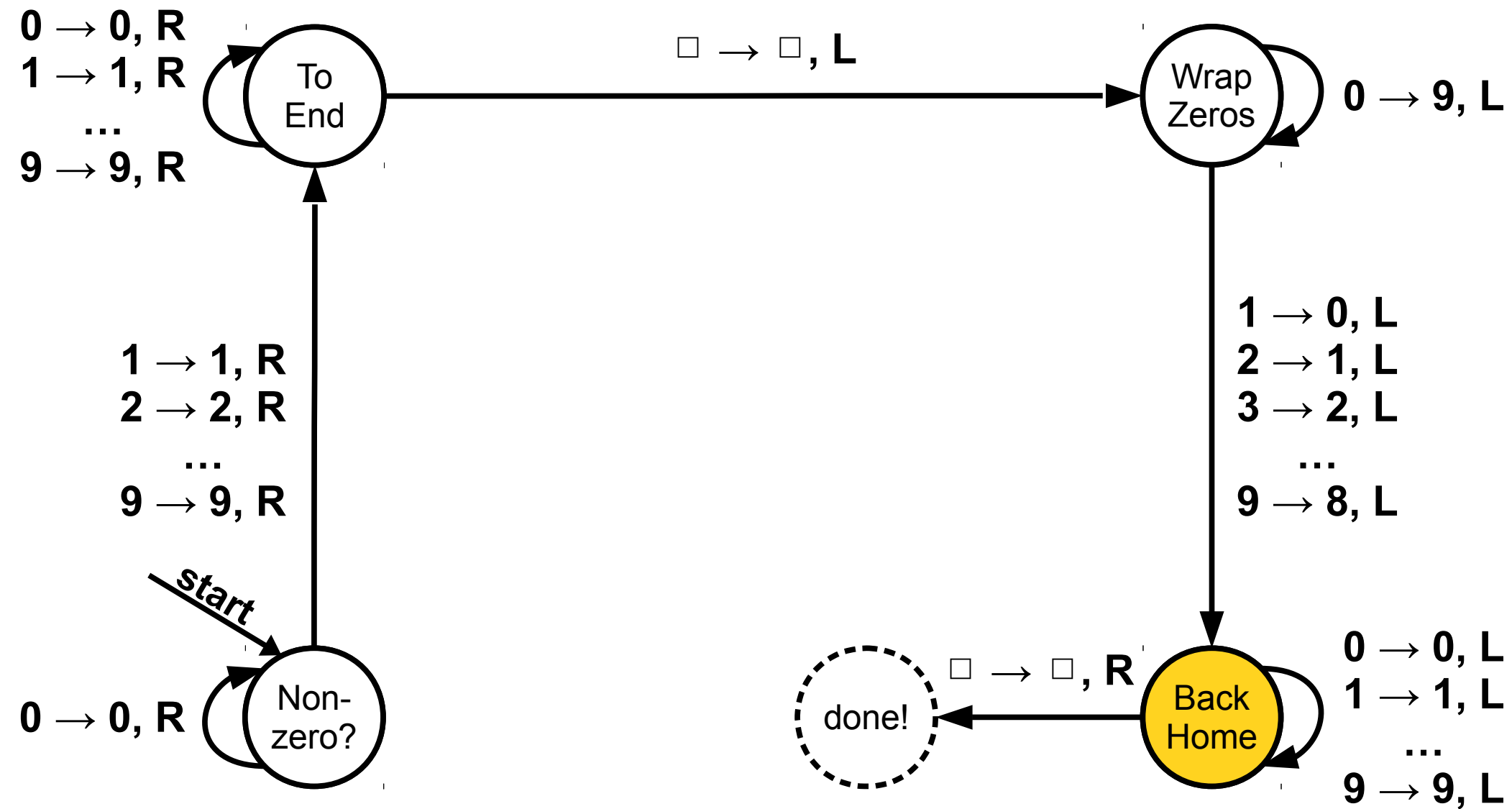


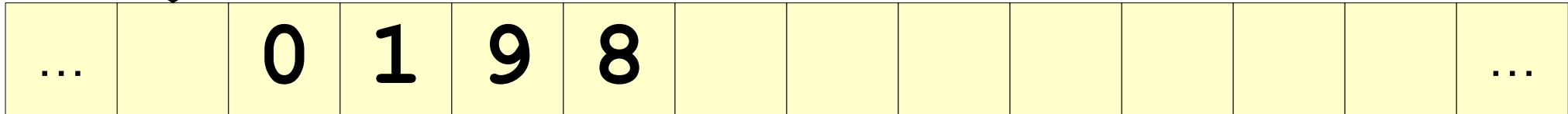
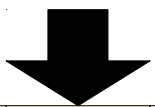
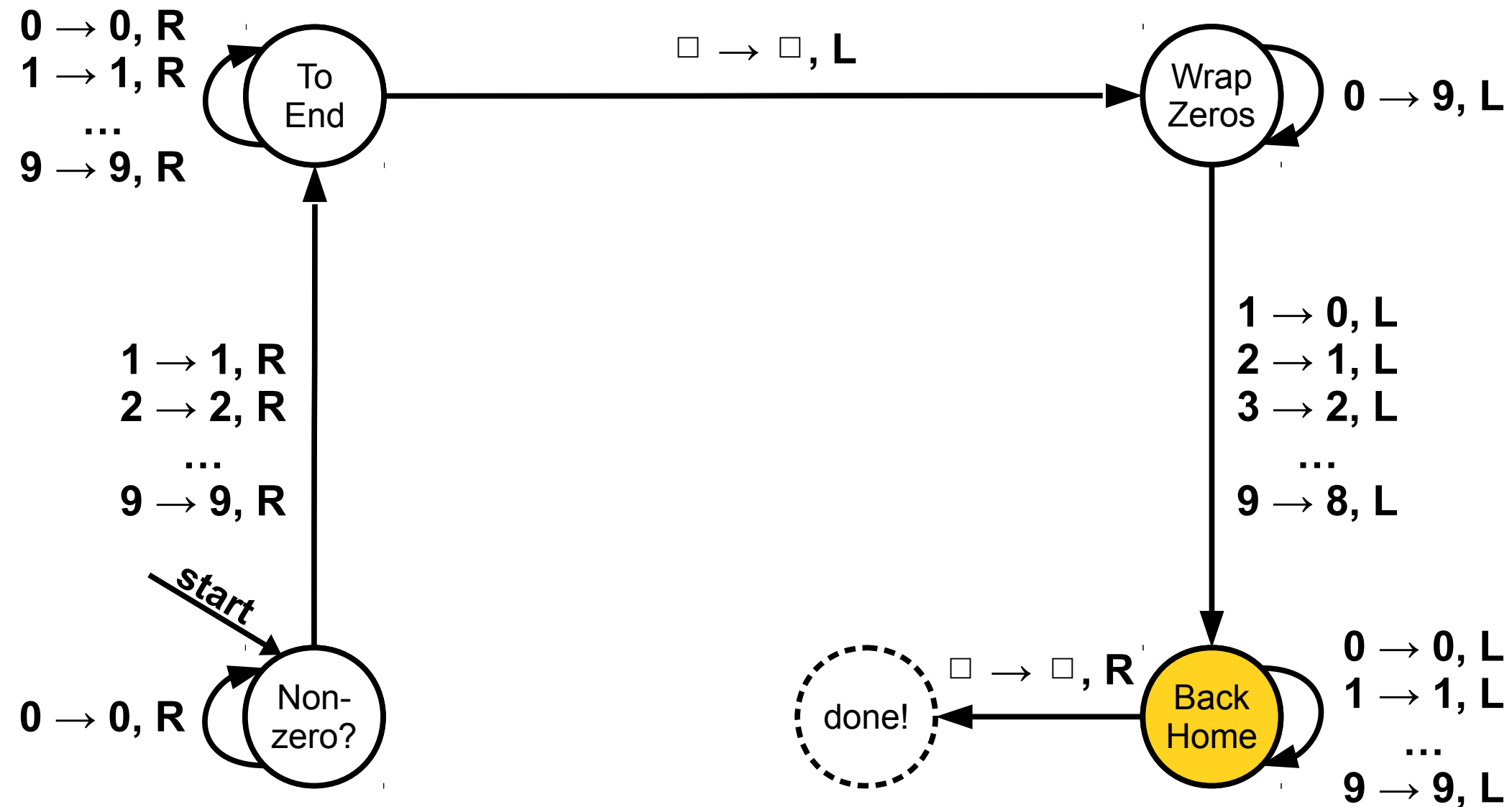


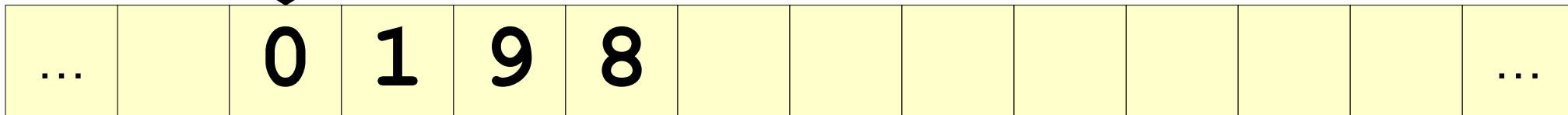
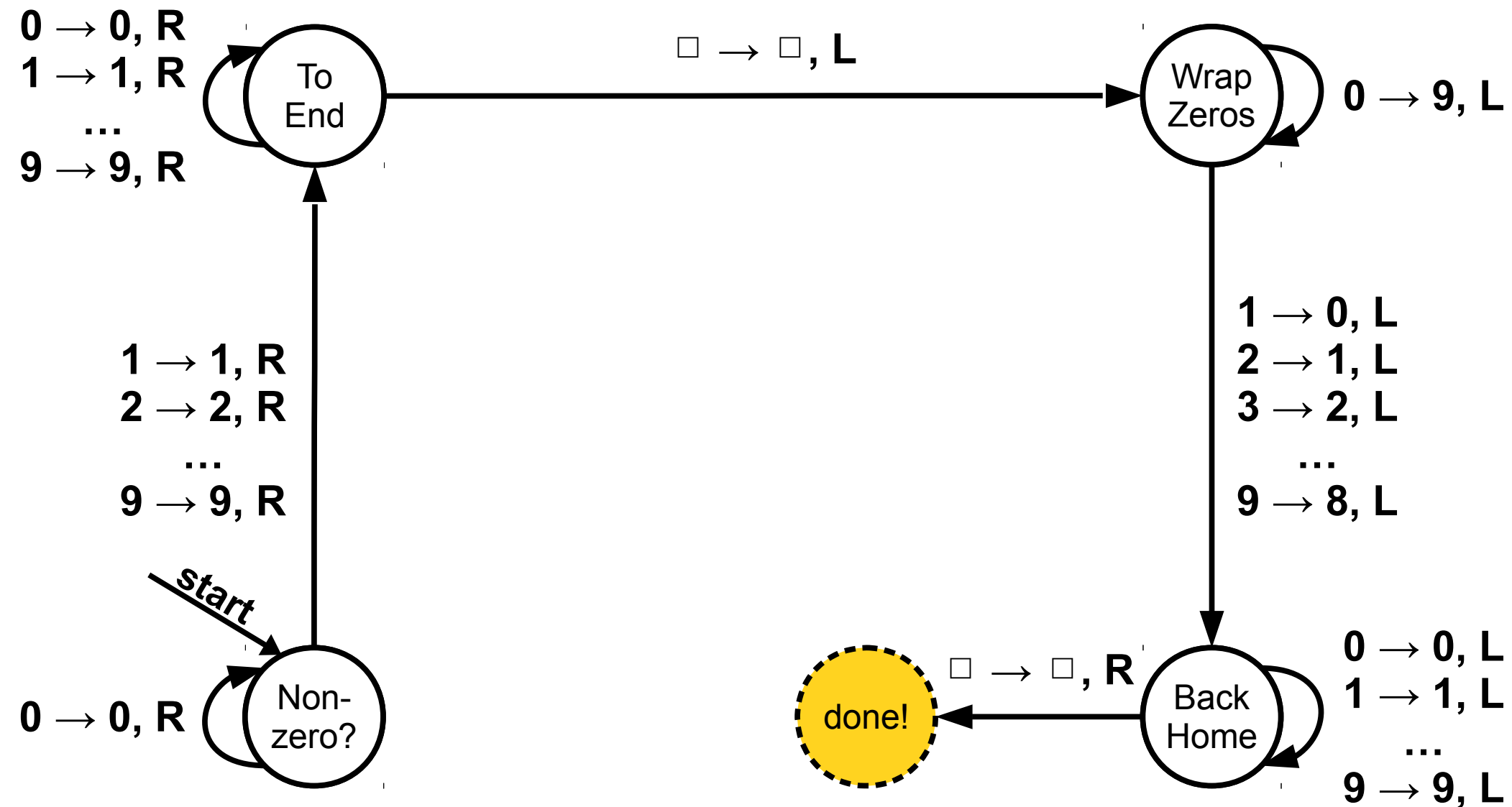


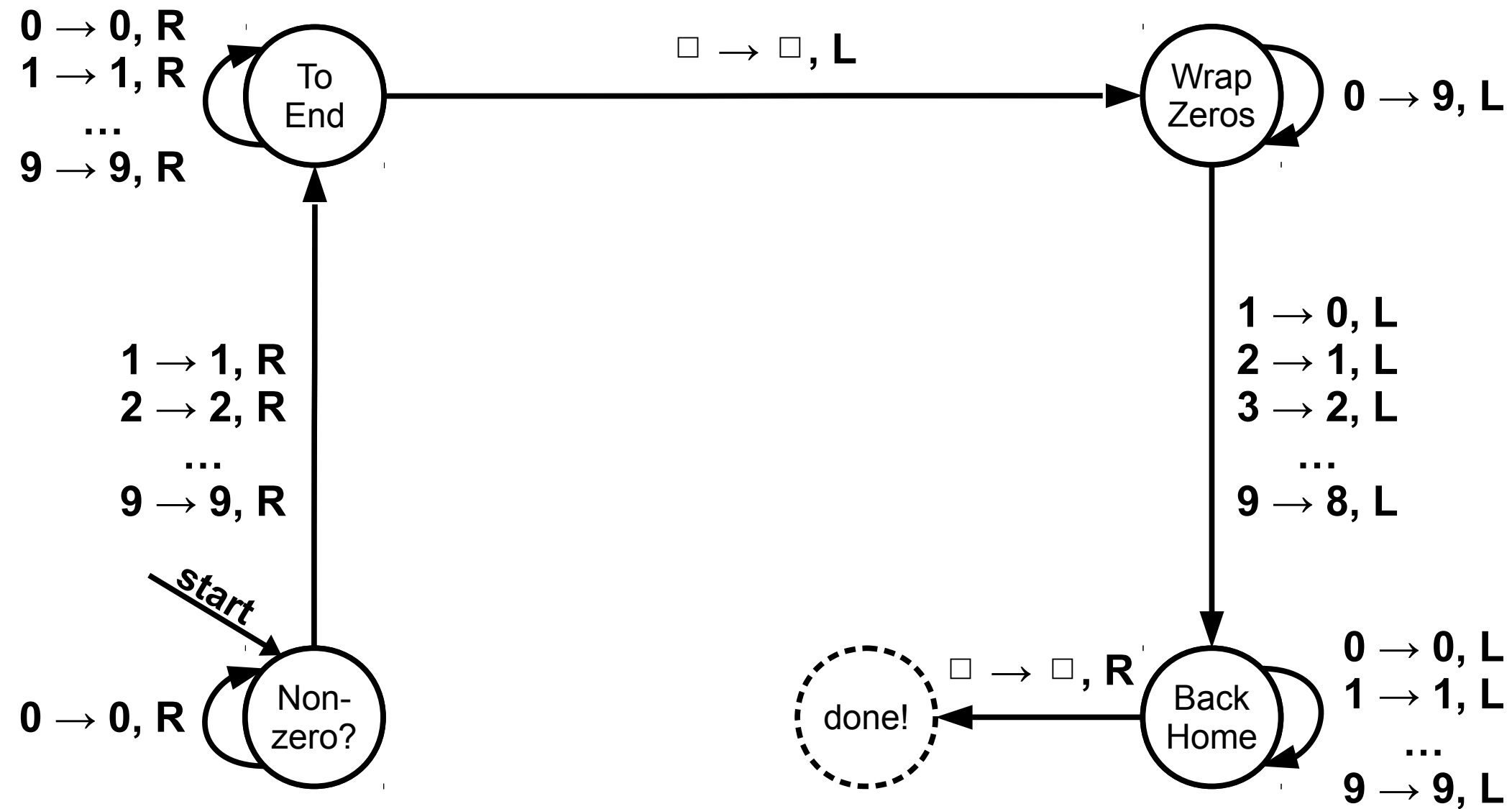


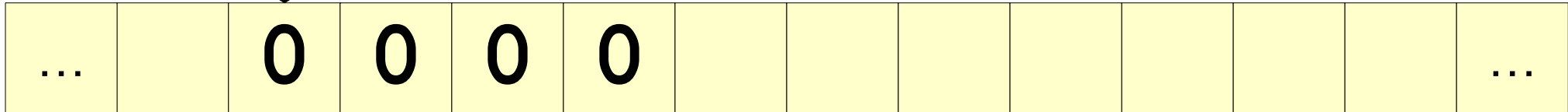
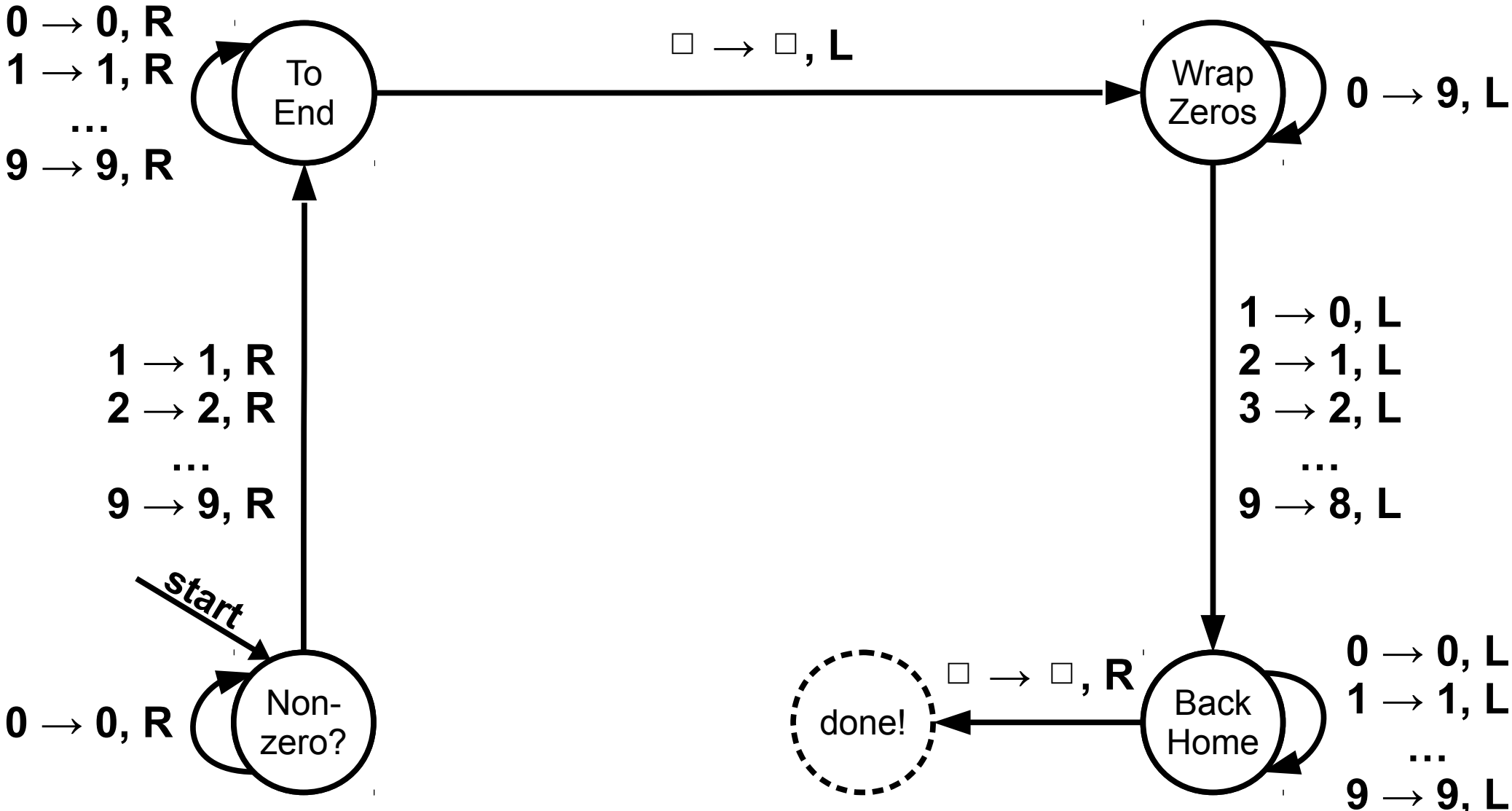


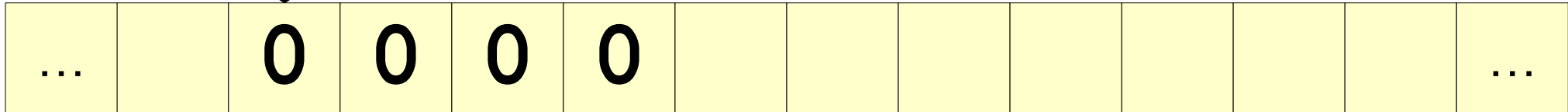
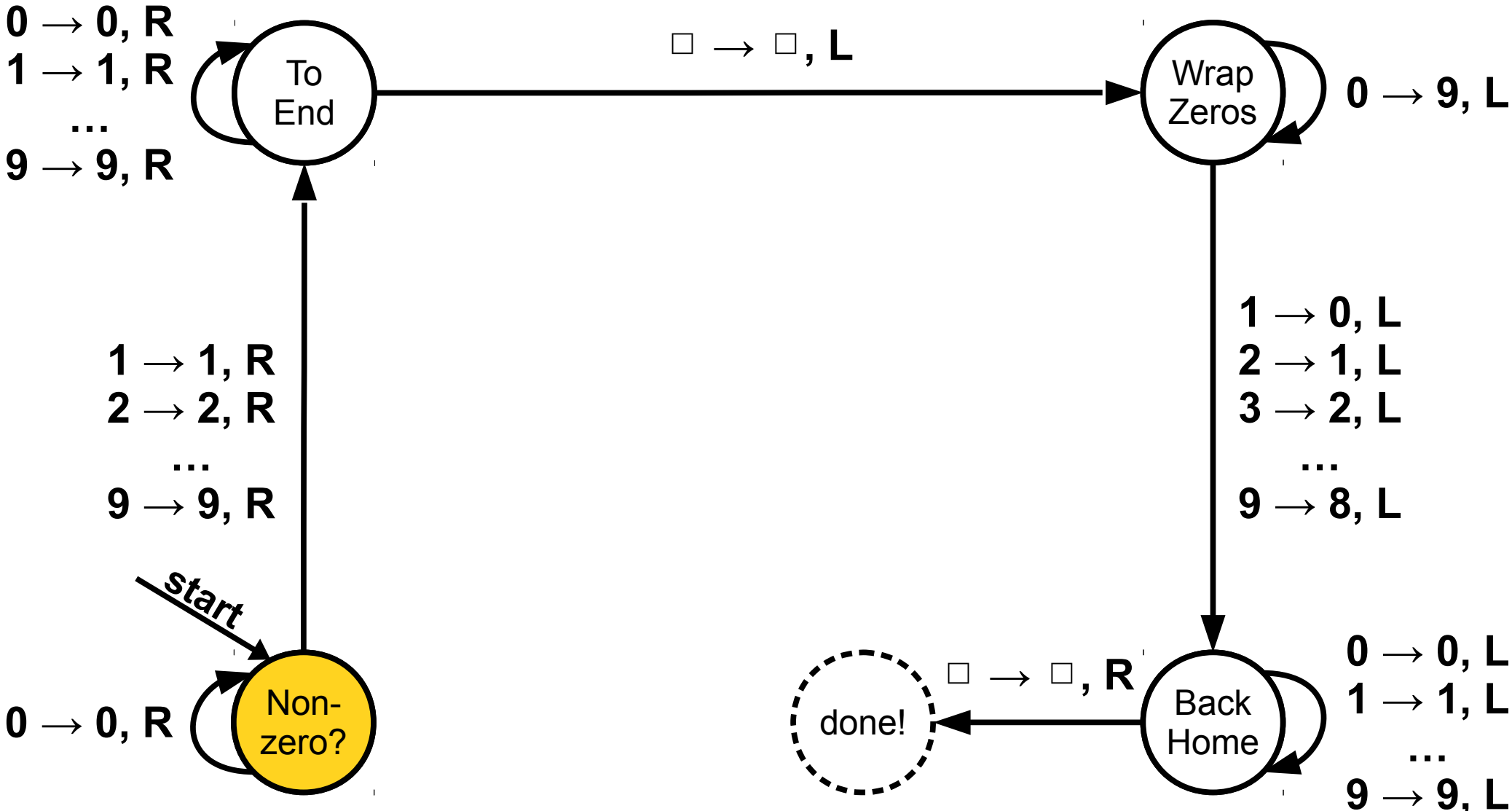


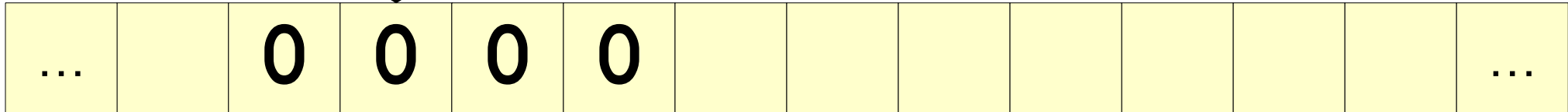
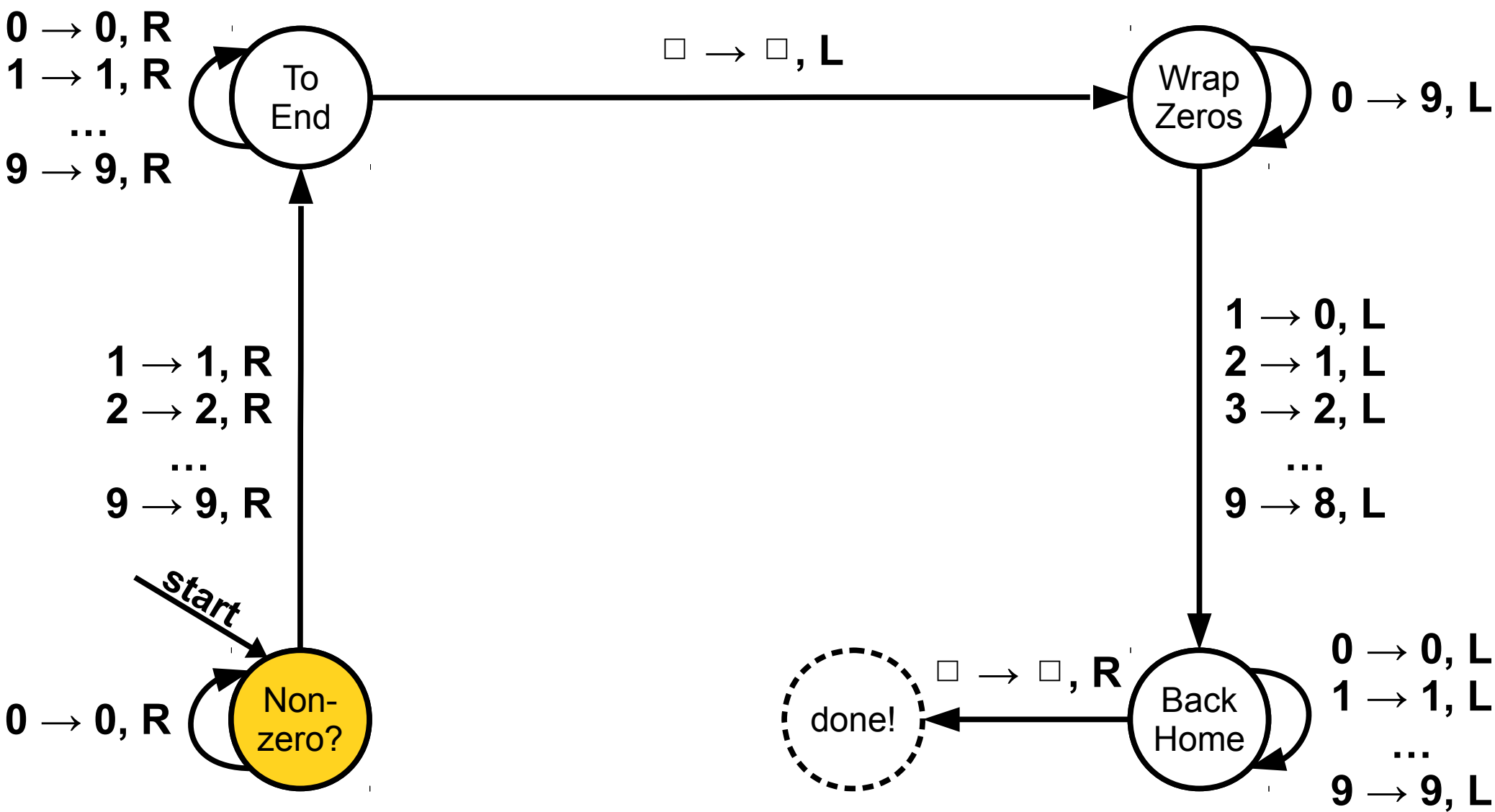


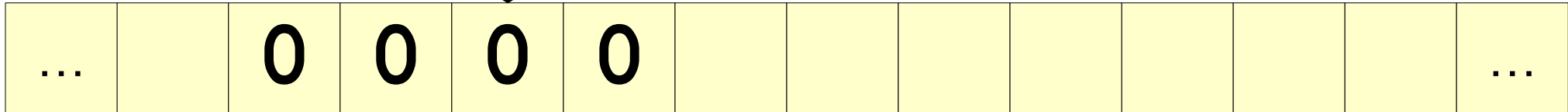
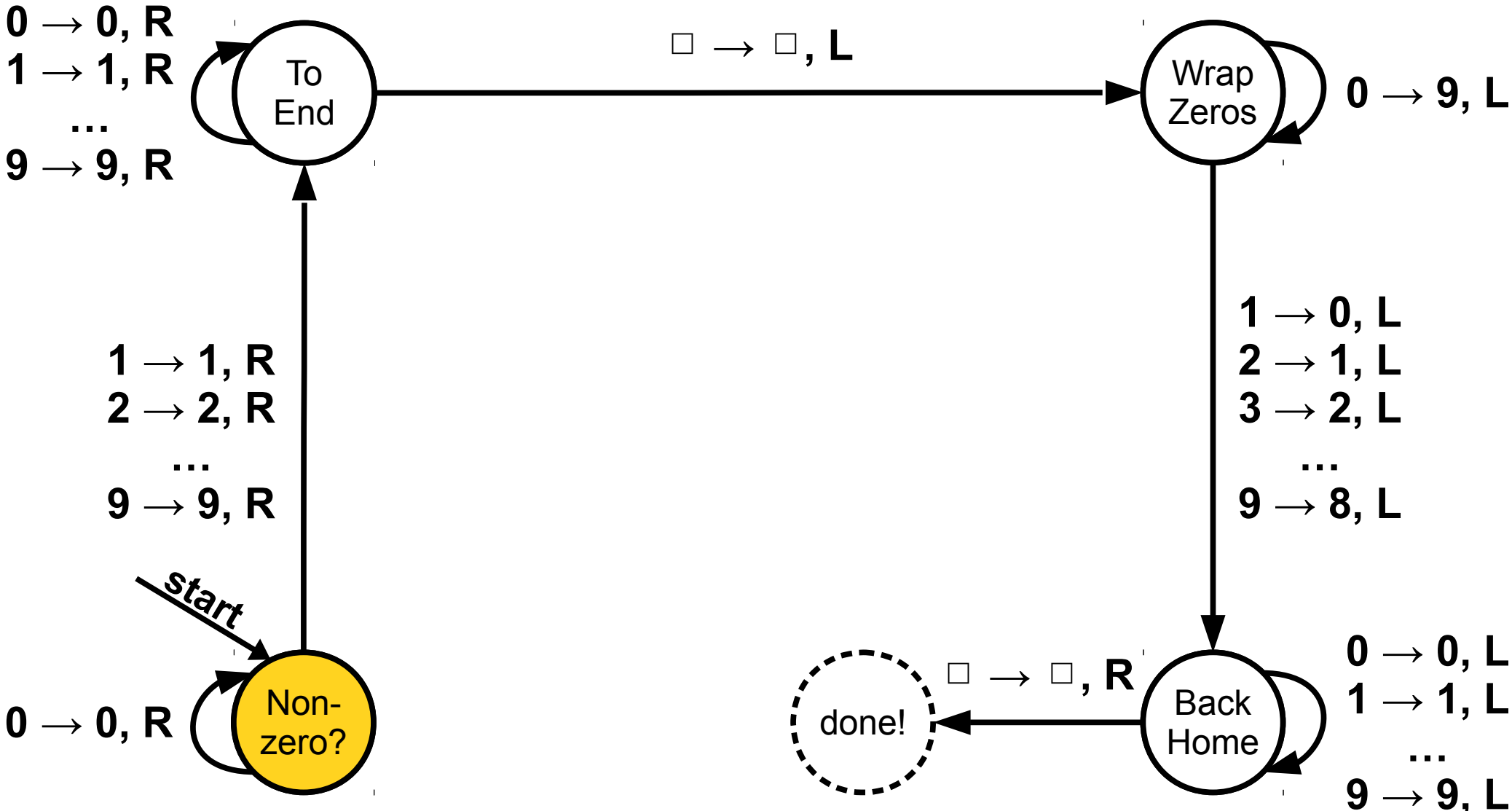


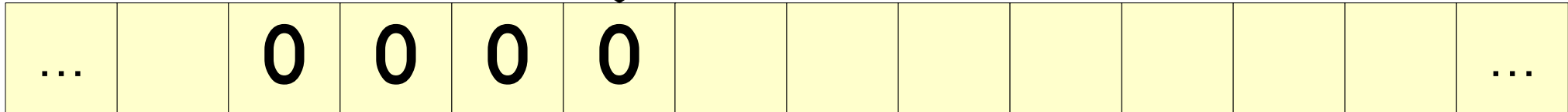
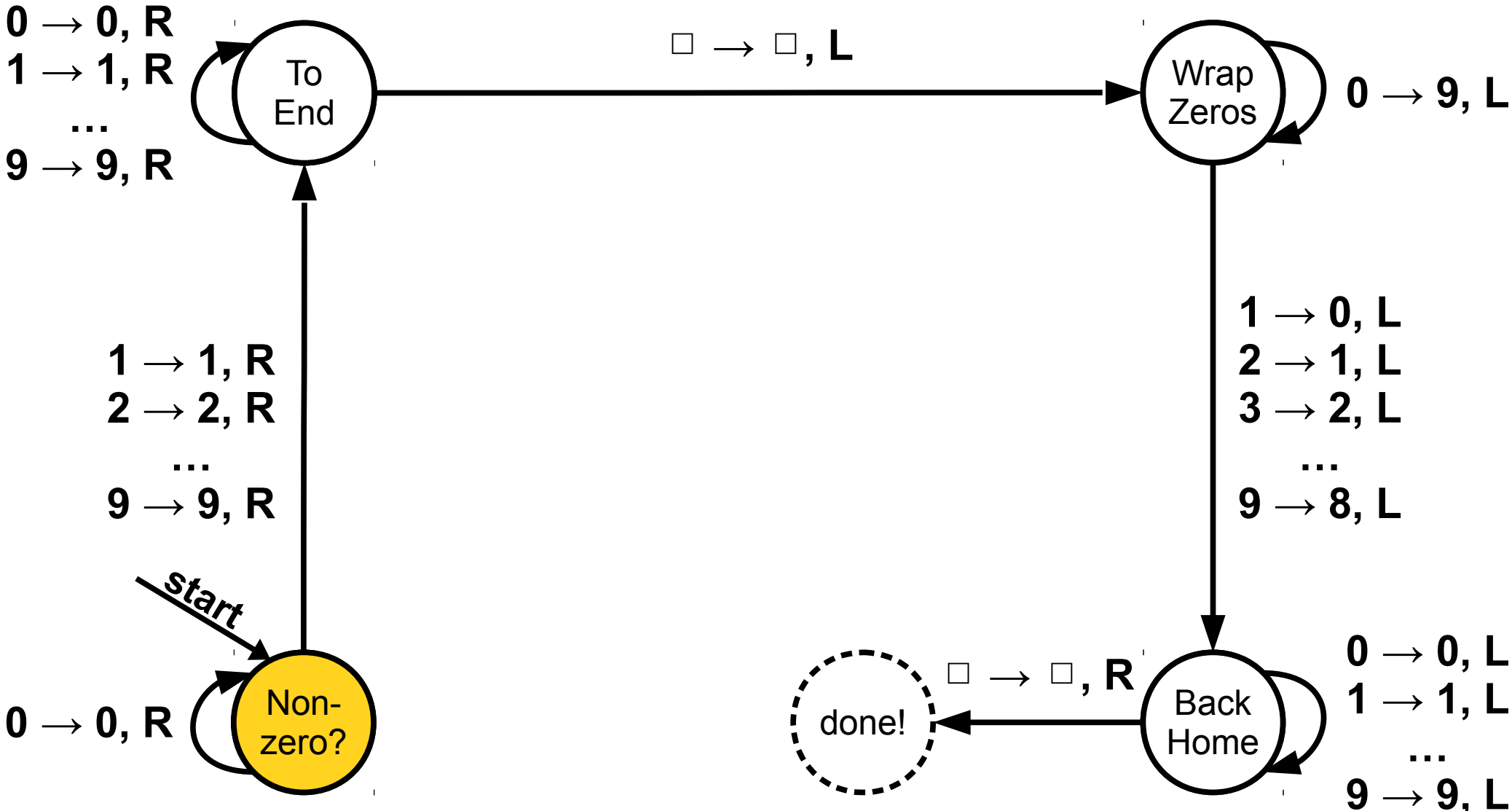


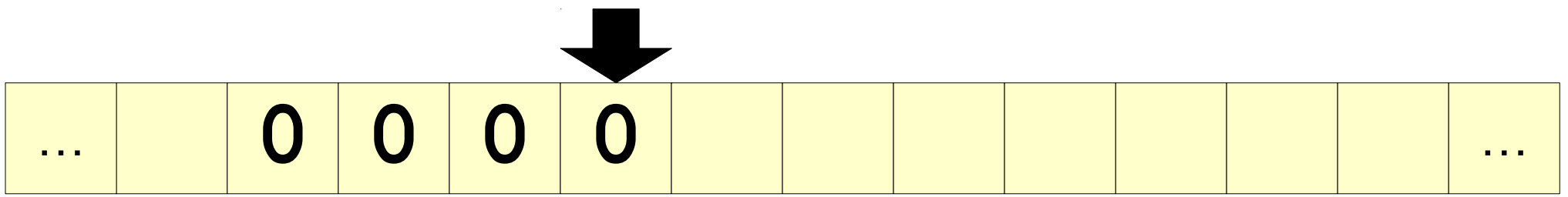
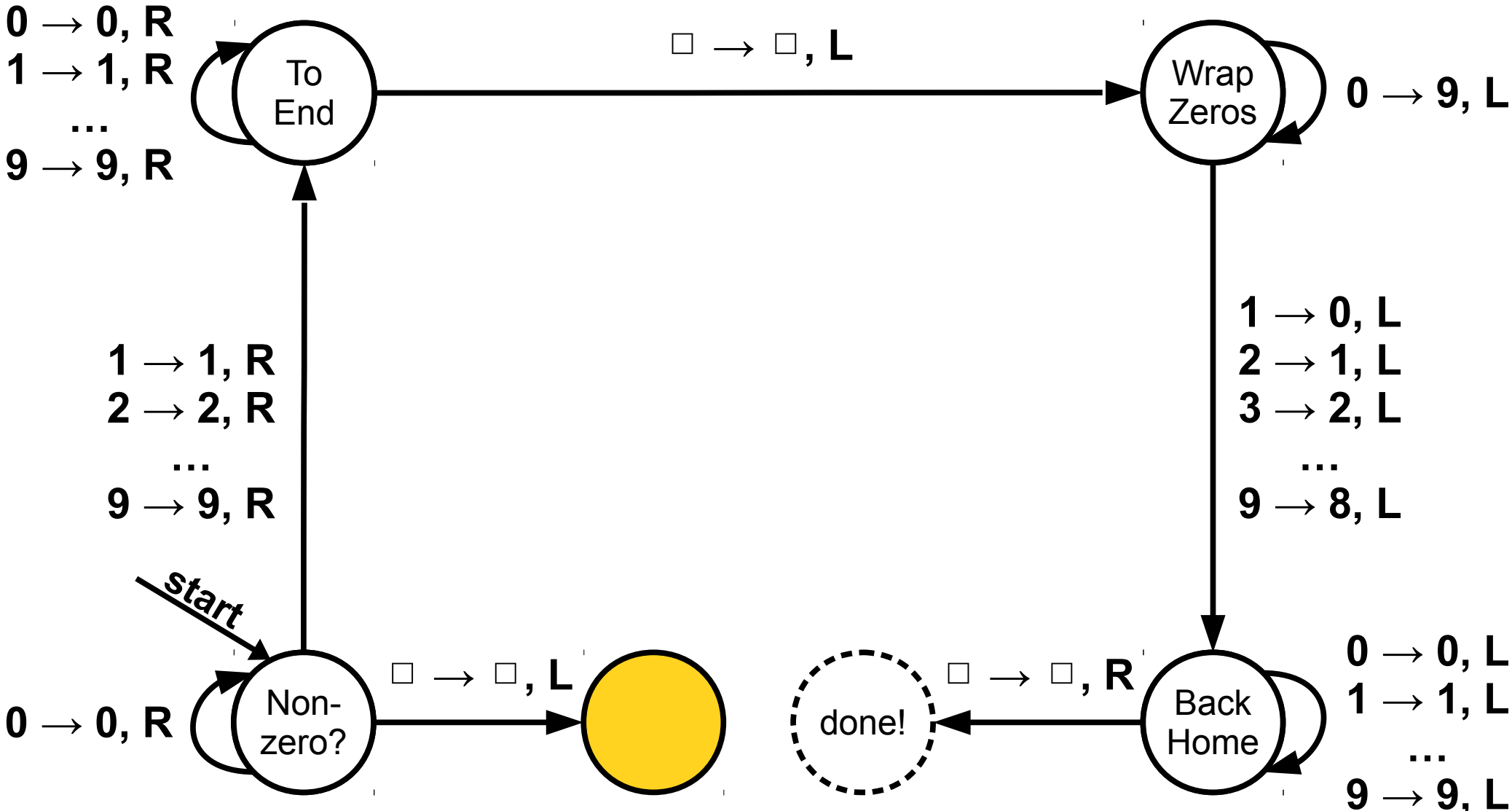


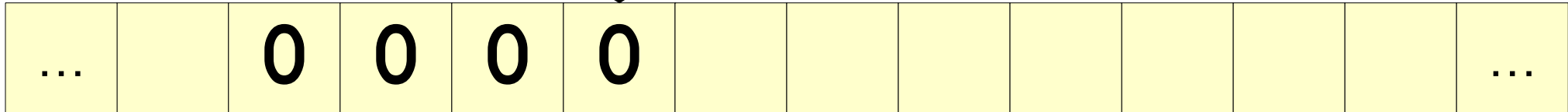
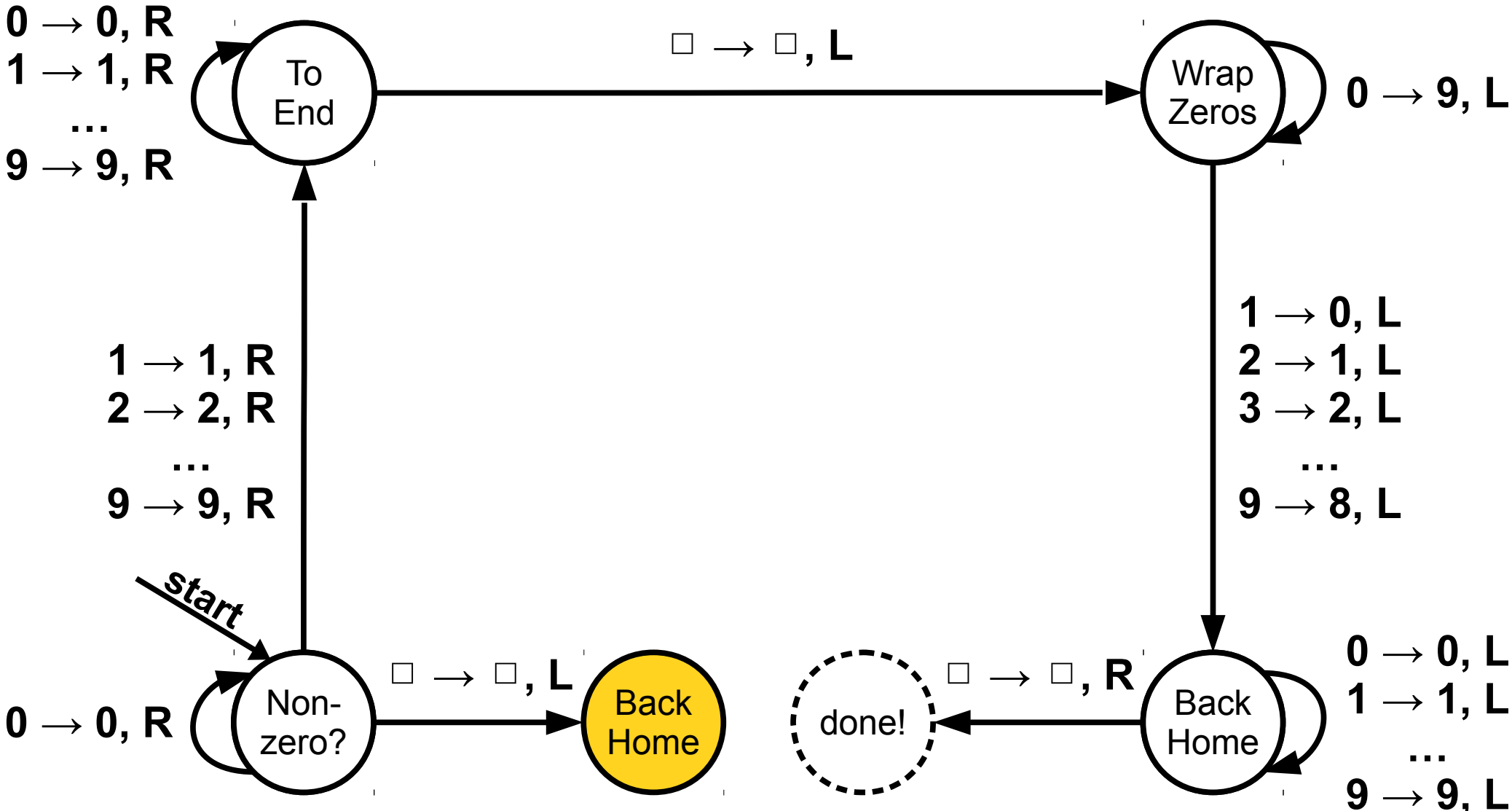


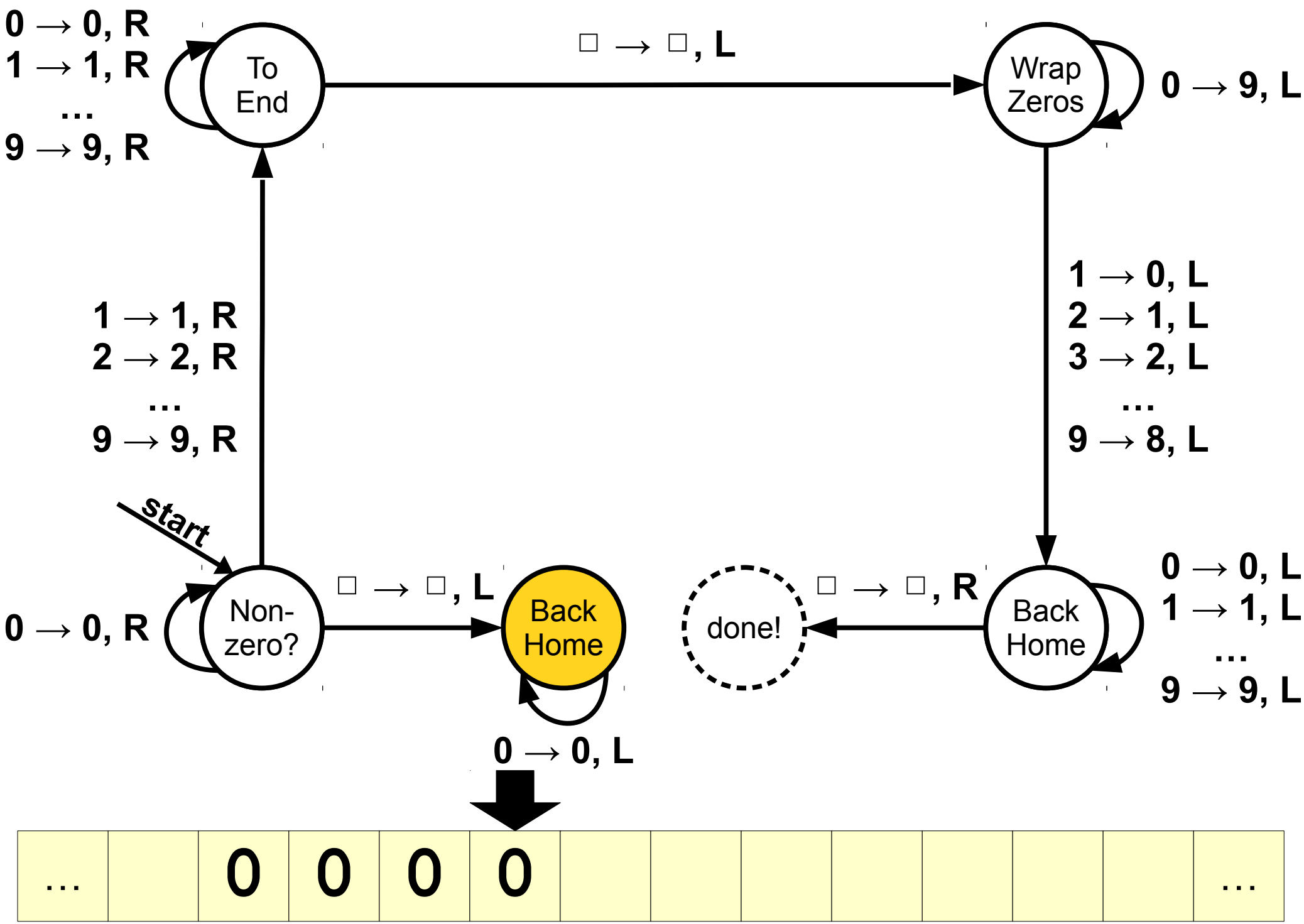


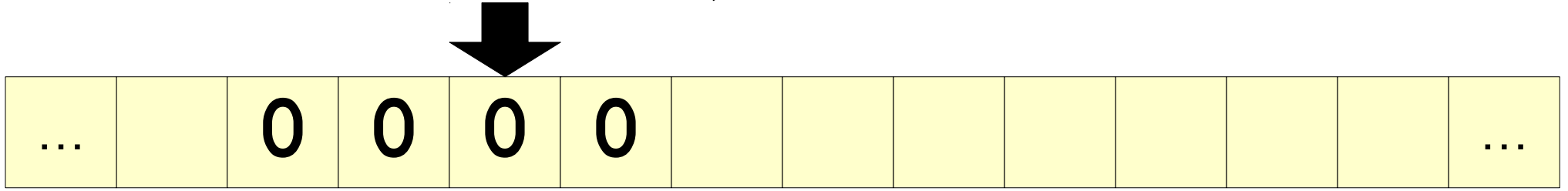
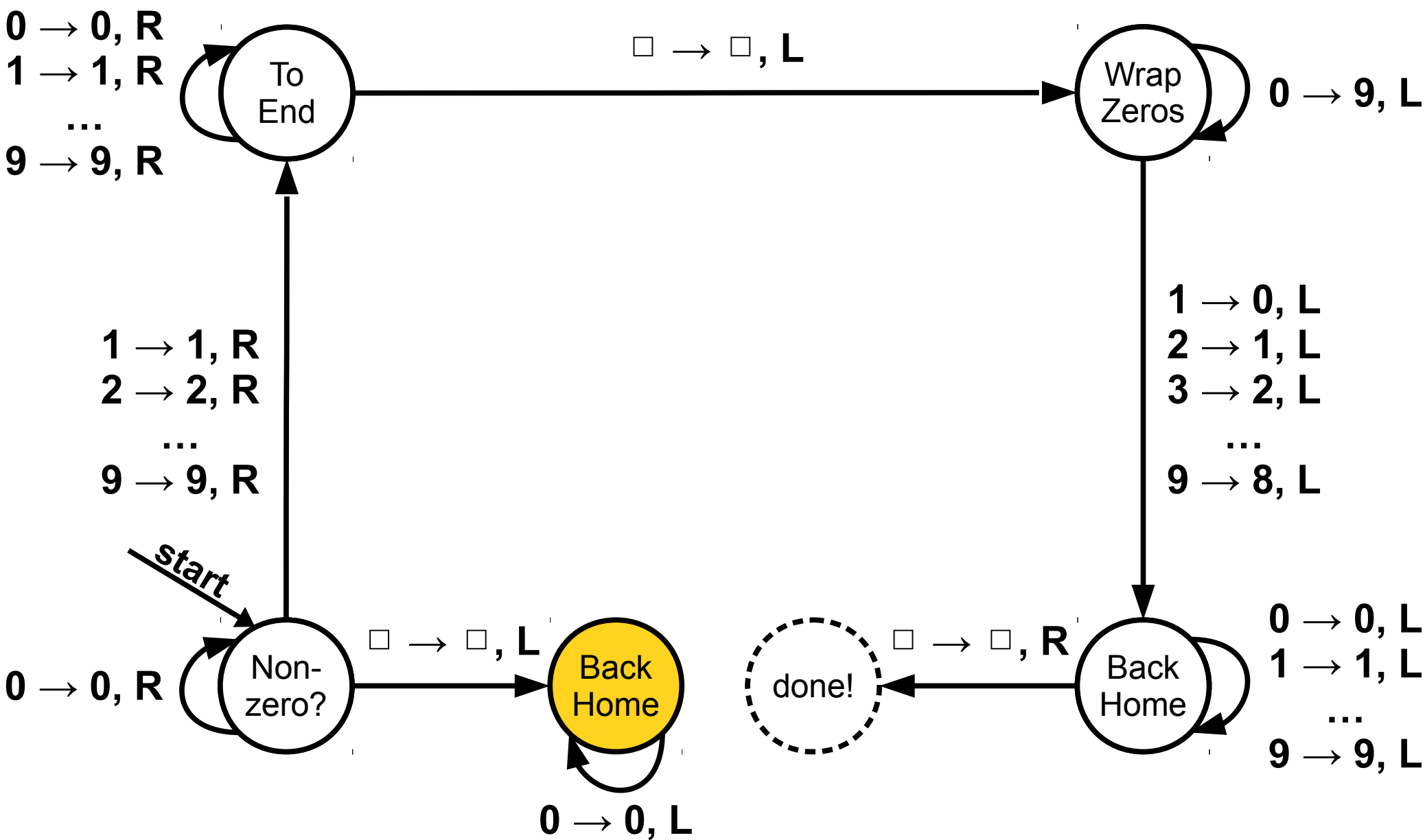


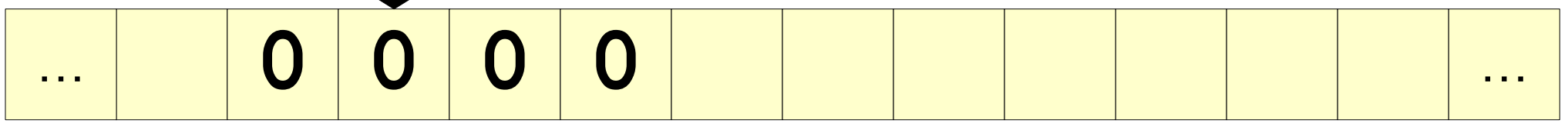
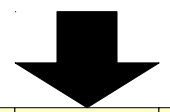
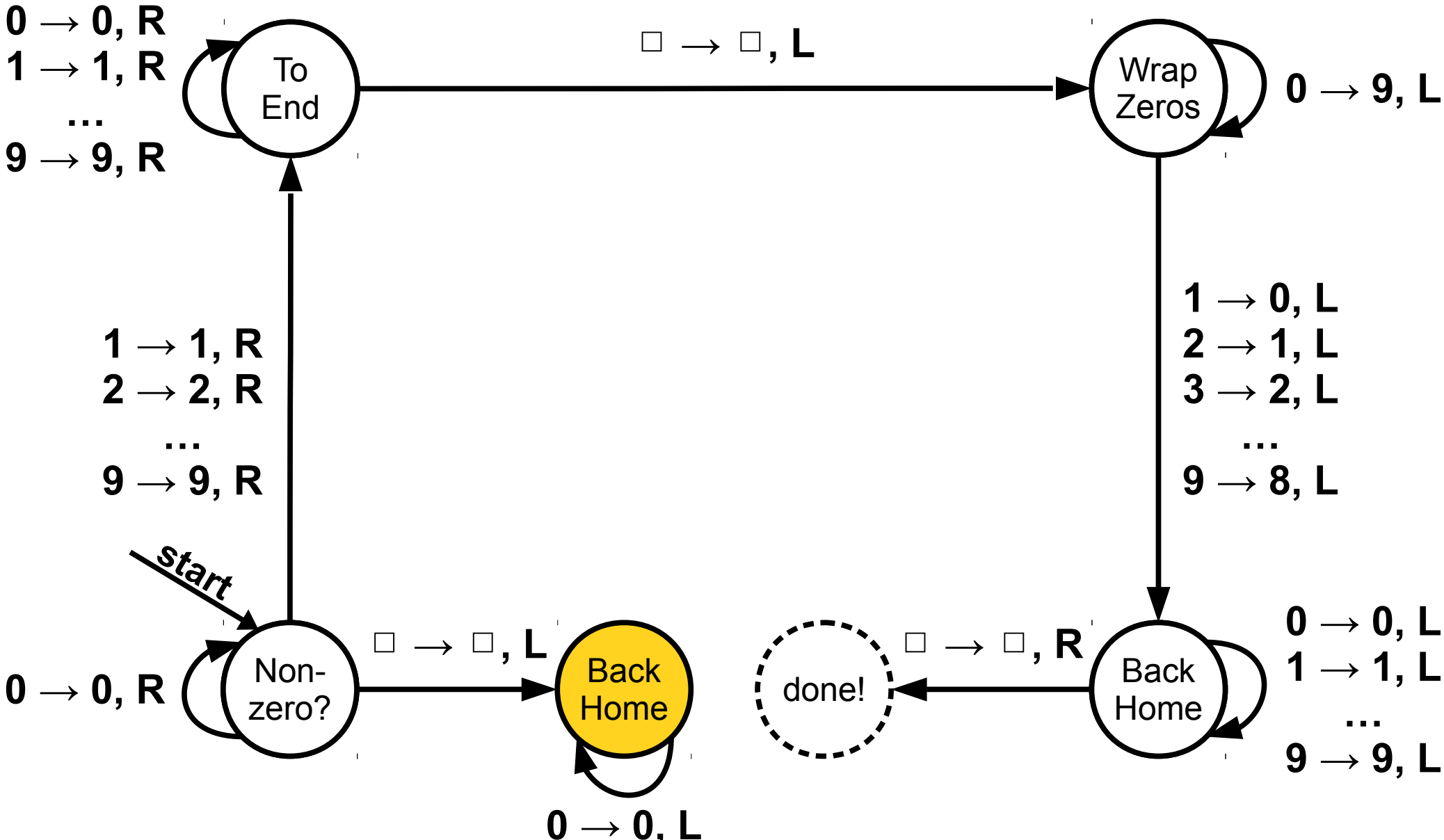


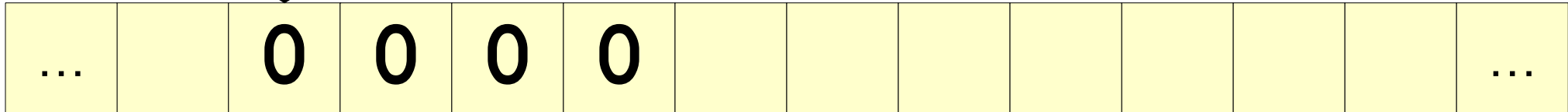
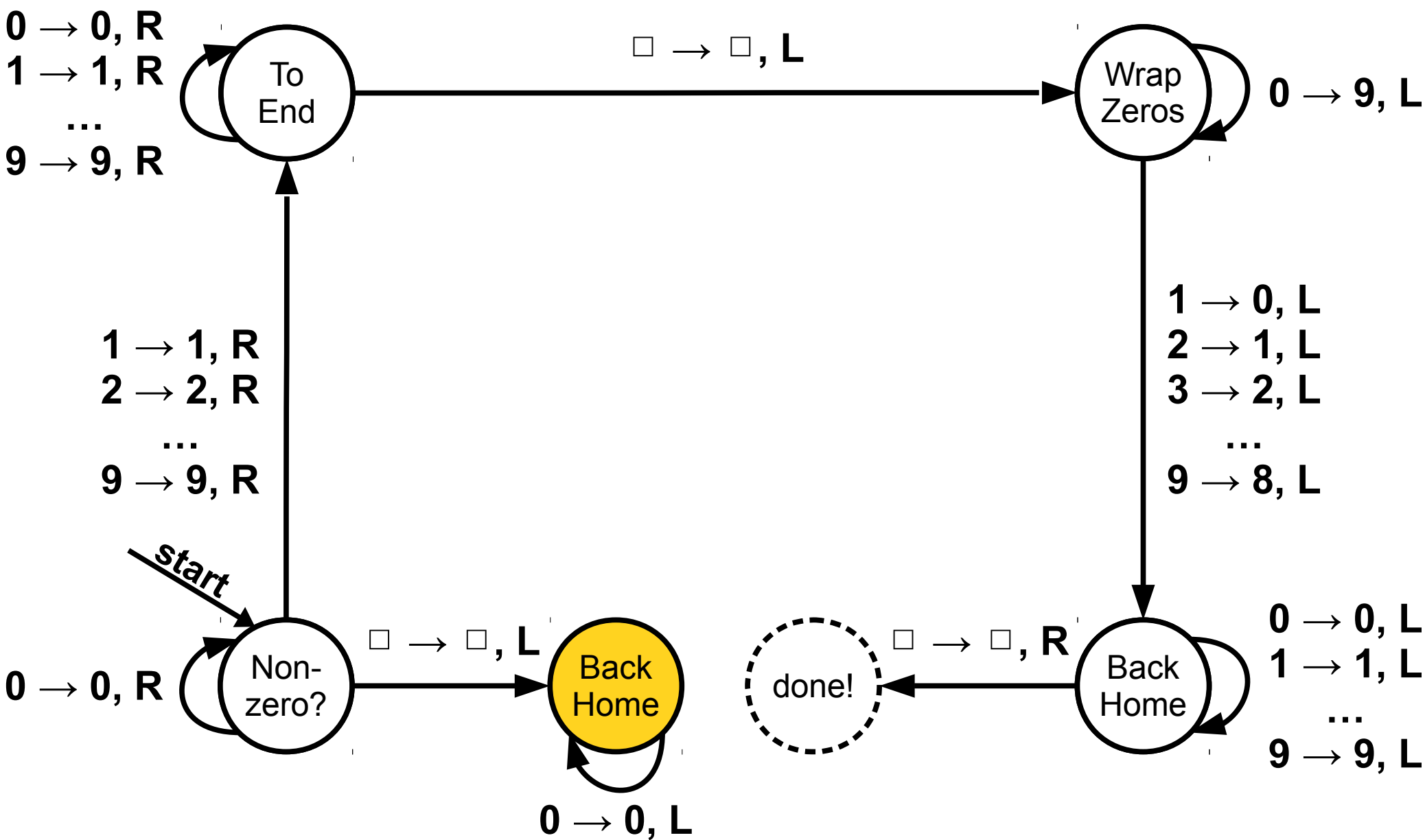


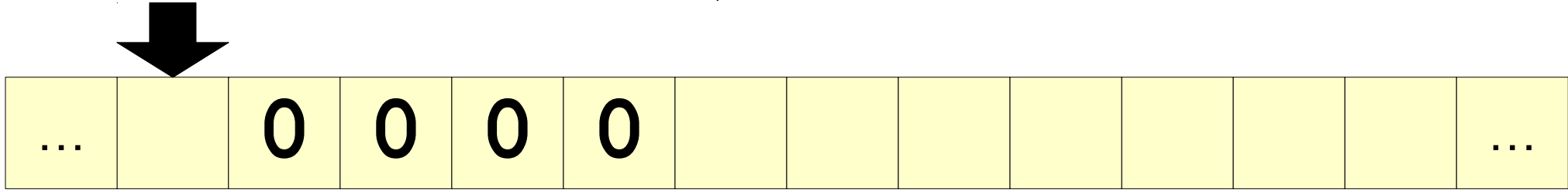
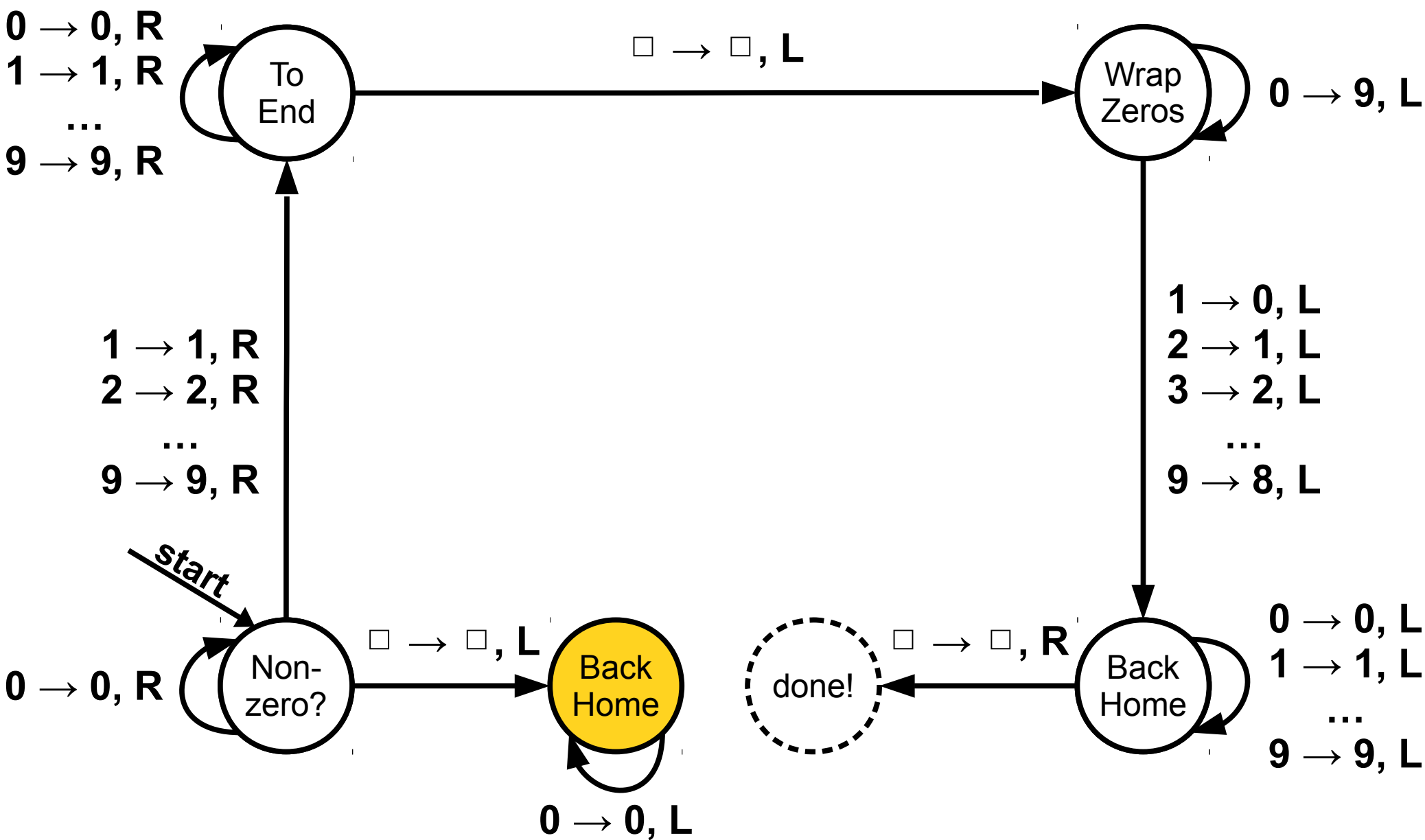


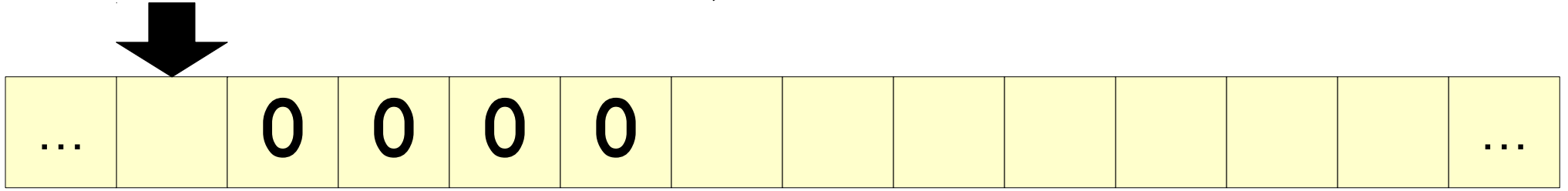
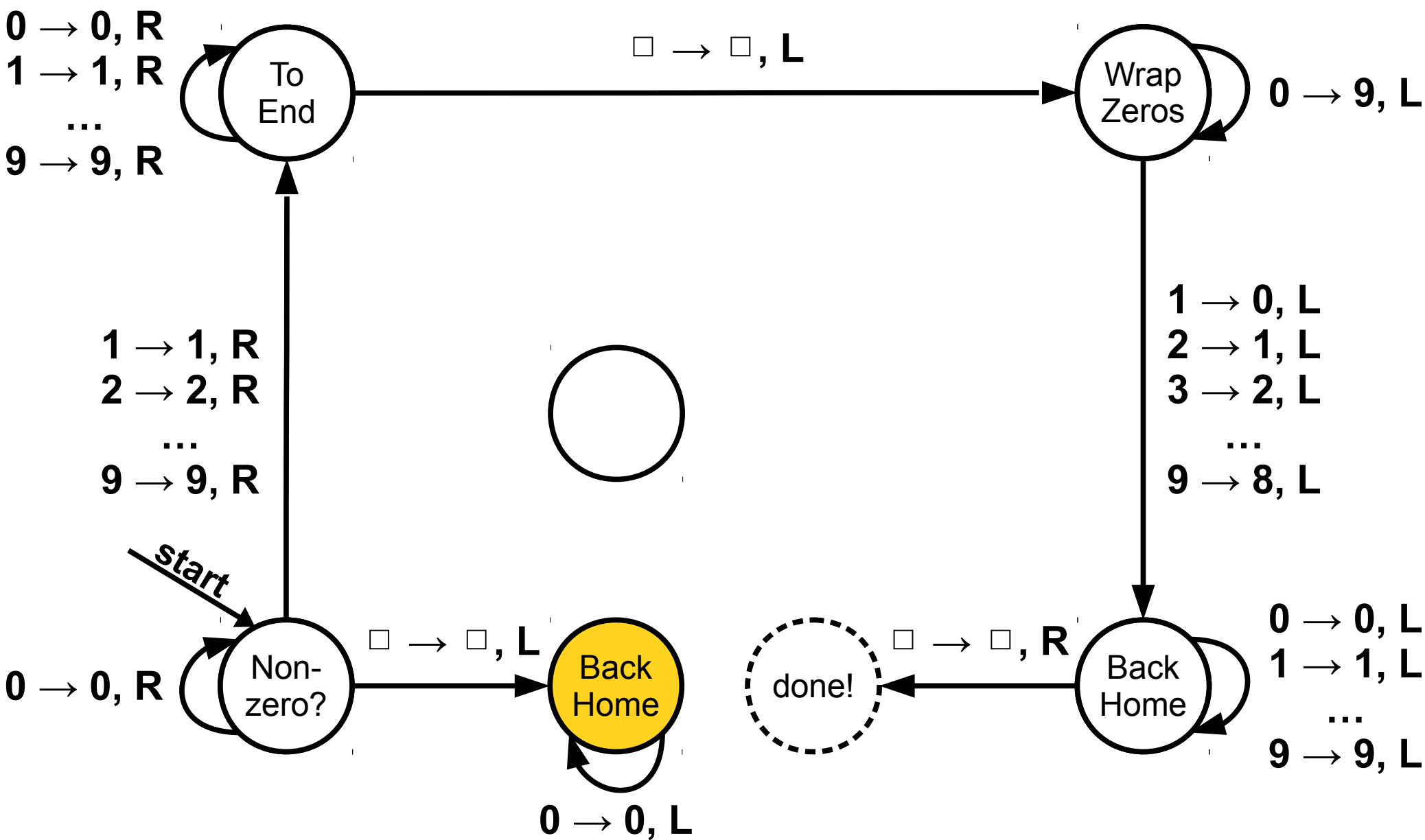


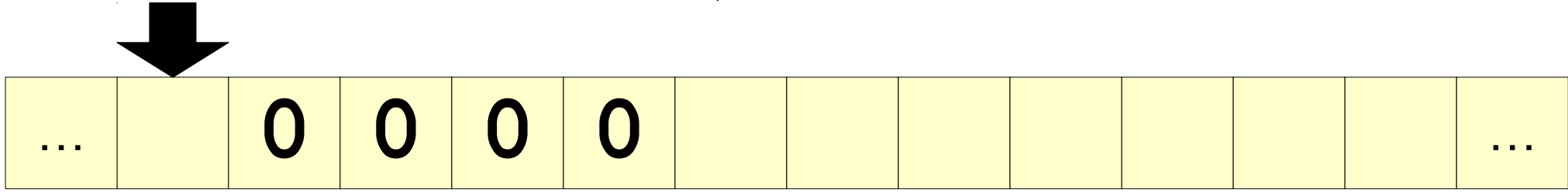
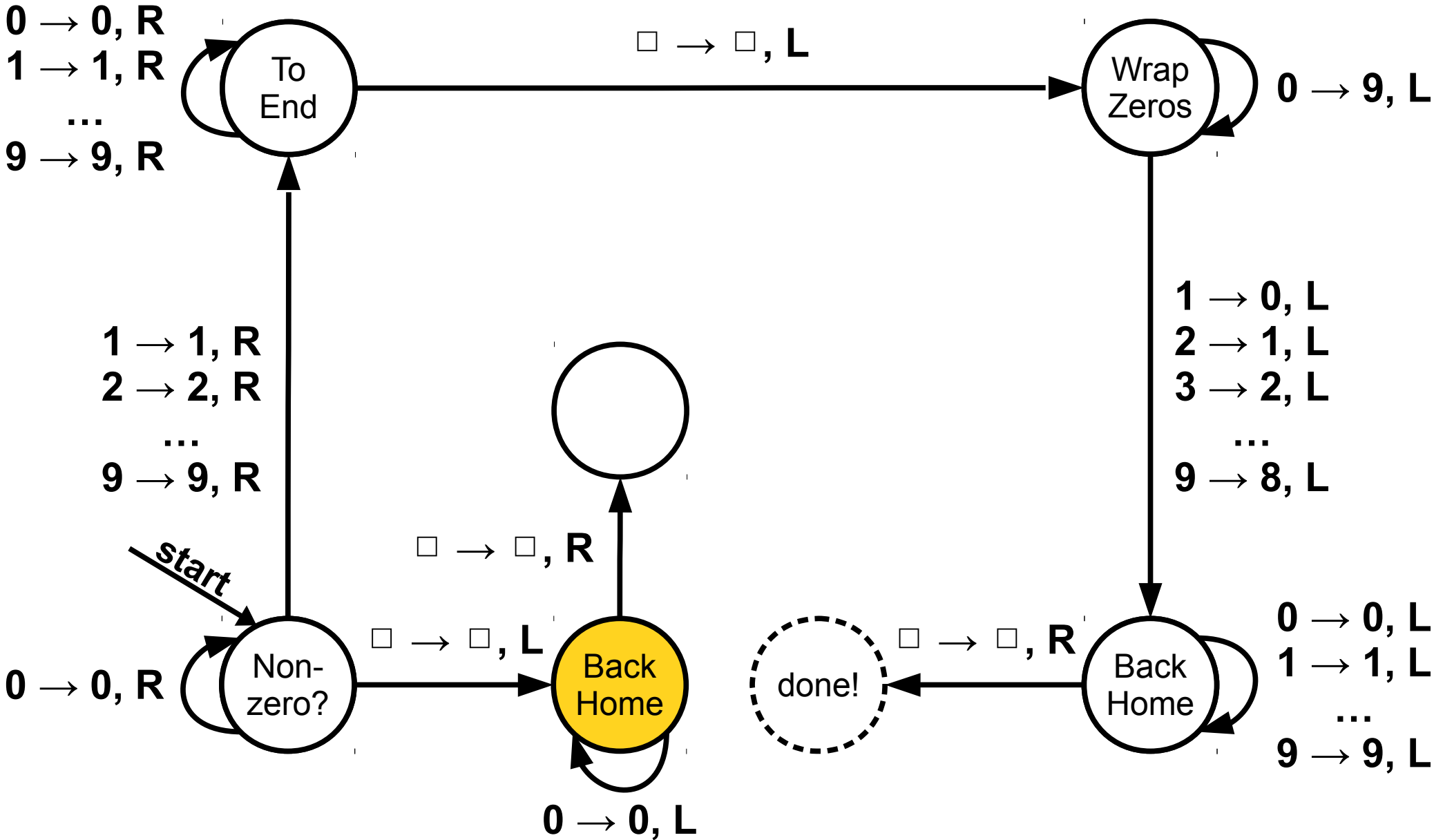


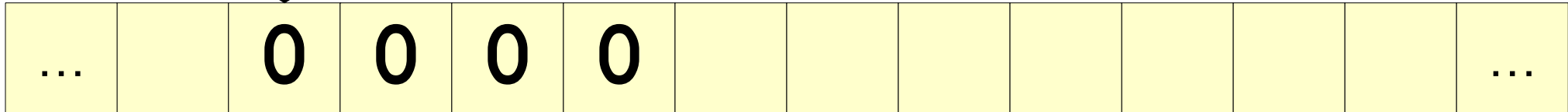
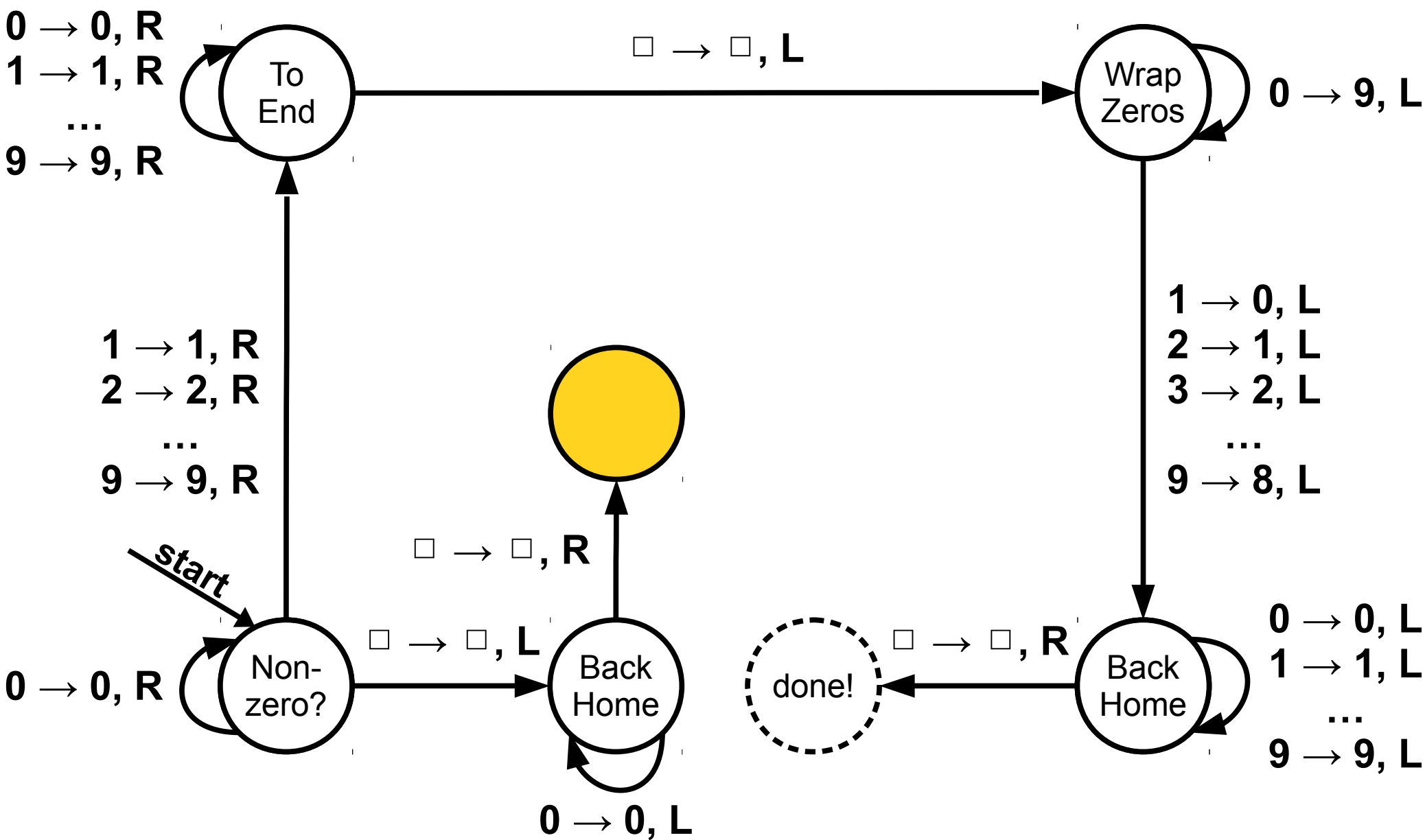


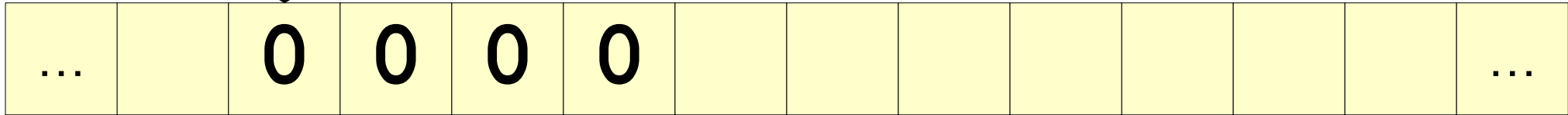
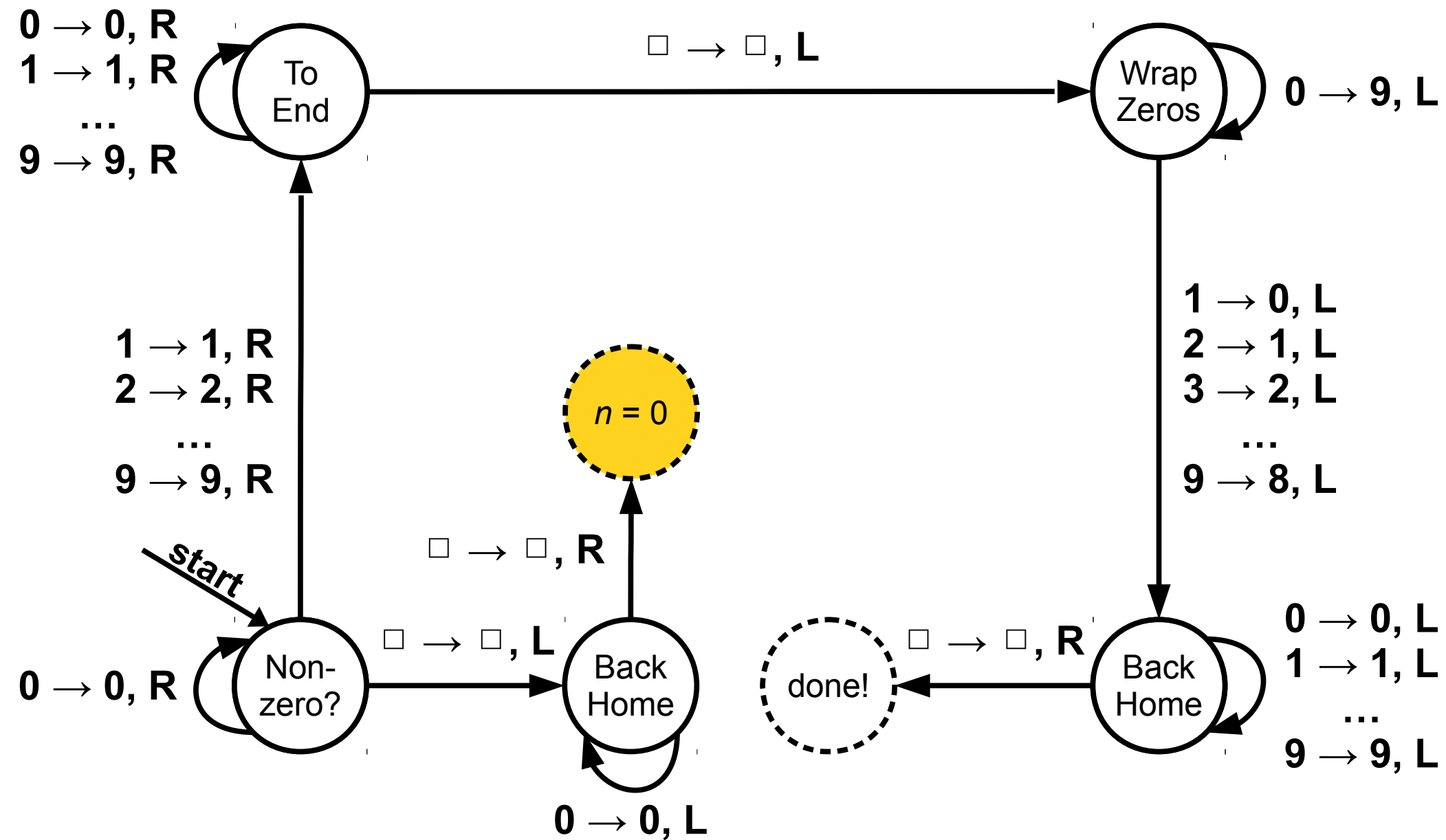


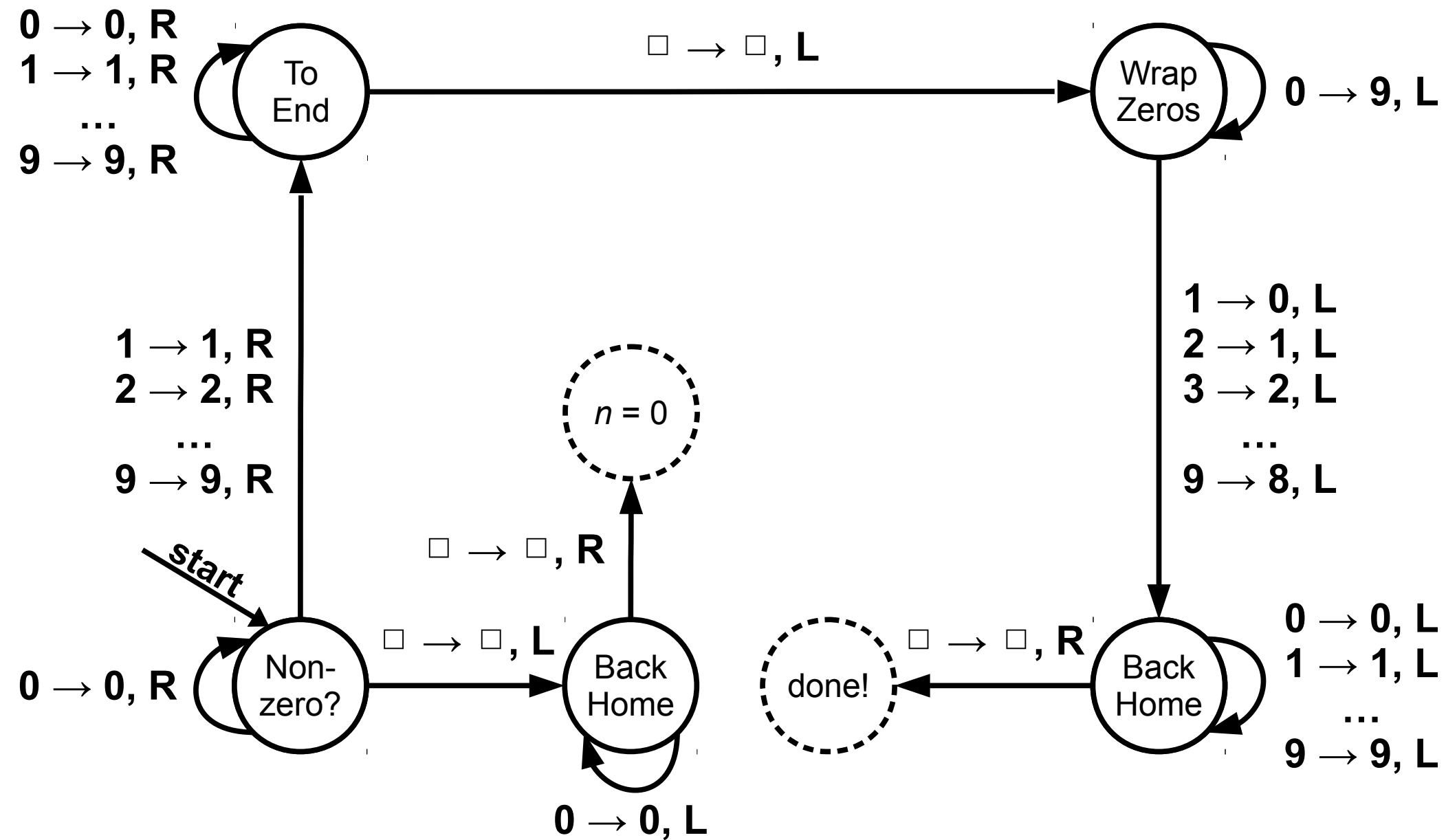










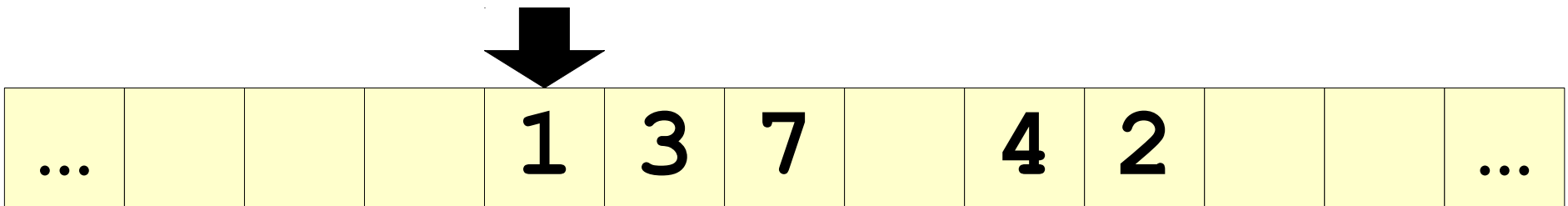


TM Subroutines

- Sometimes, a subroutine needs to report back some information about what happened.
- Just as a function can return multiple different values, we'll allow subroutines to have different “done” states.
- Each state can then be wired to a different state, so a TM using the subroutine can control what happens next.

Putting it All Together

- Our goal is to build a TM that, given two numbers, adds those numbers together.
- Before:



- After:

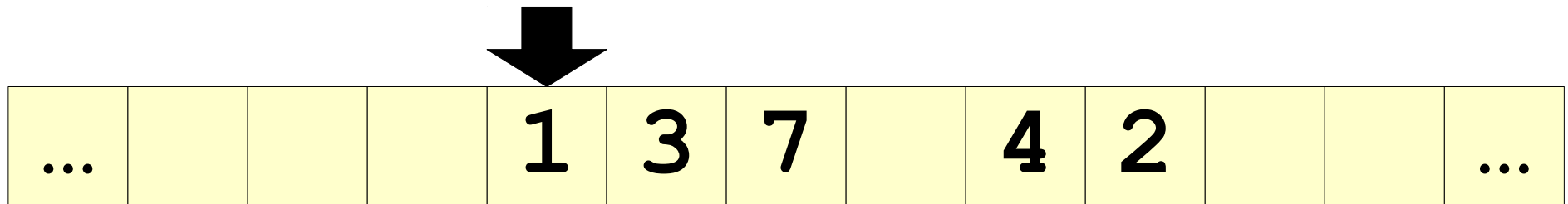


TM Arithmetic

```
add(num1, num2) {  
    while (num2 > 0) {  
        increment(num1);  
        decrement(num2);  
    }  
}
```

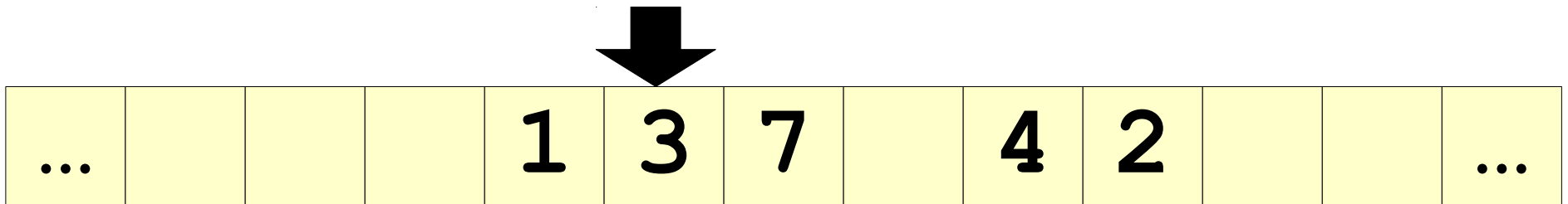
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



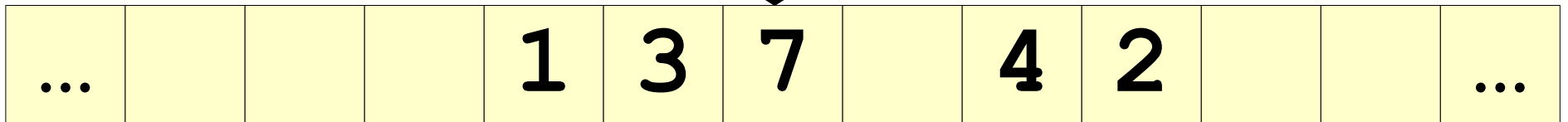
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



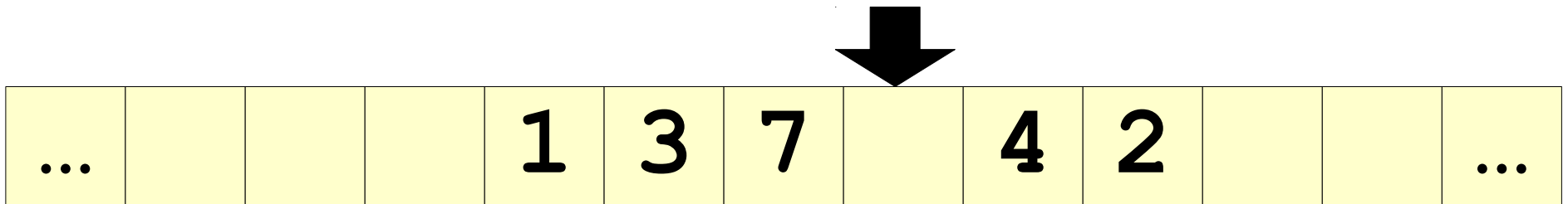
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



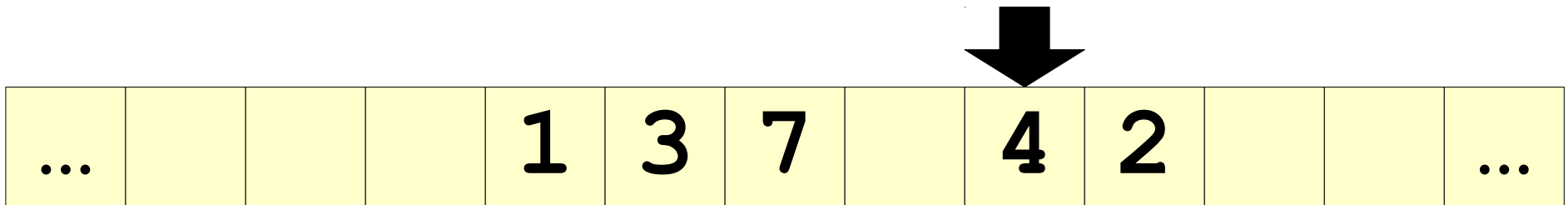
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



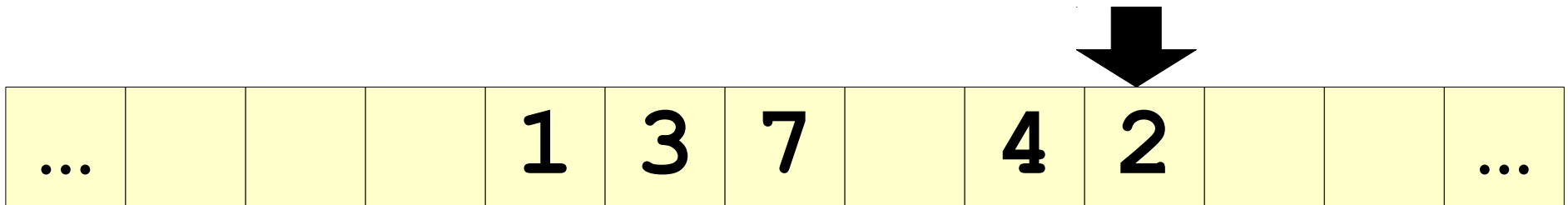
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



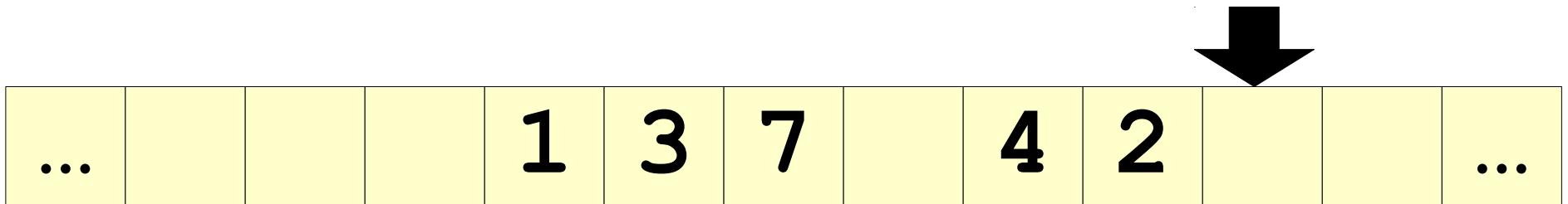
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



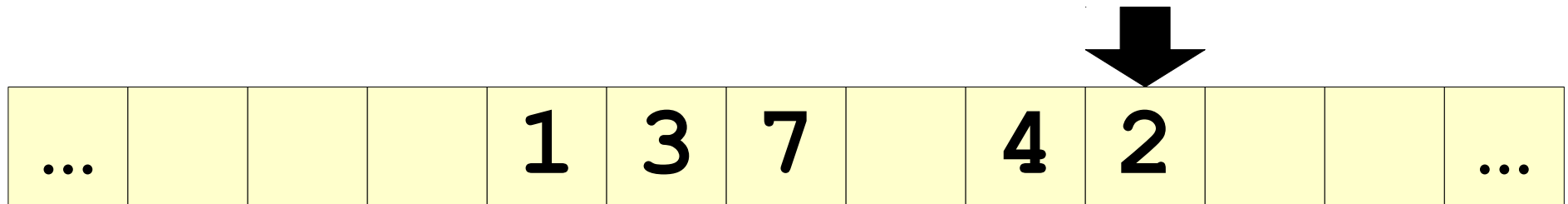
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



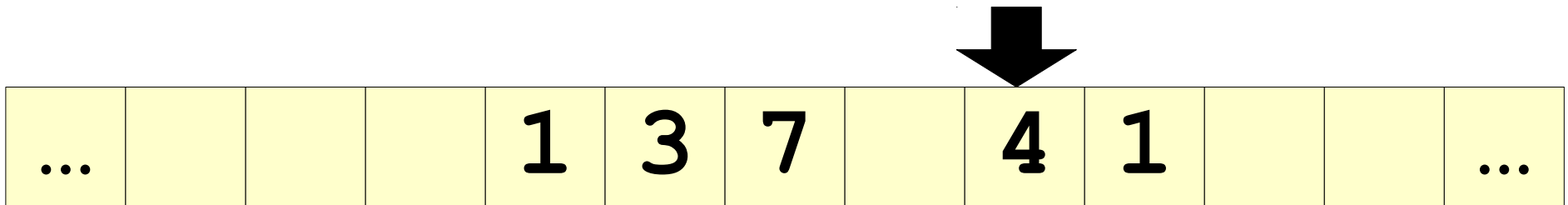
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



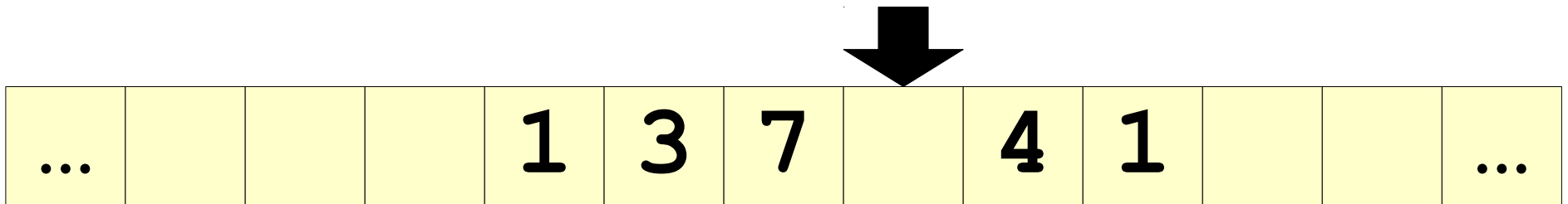
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



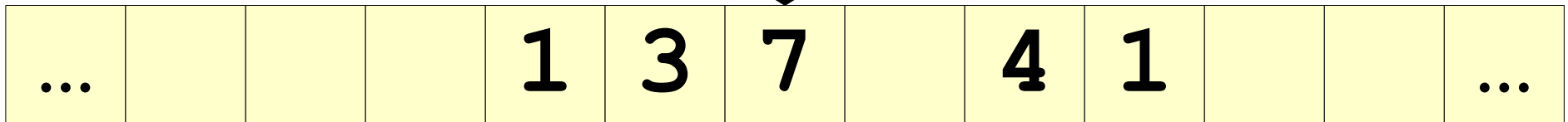
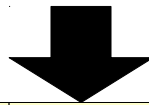
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



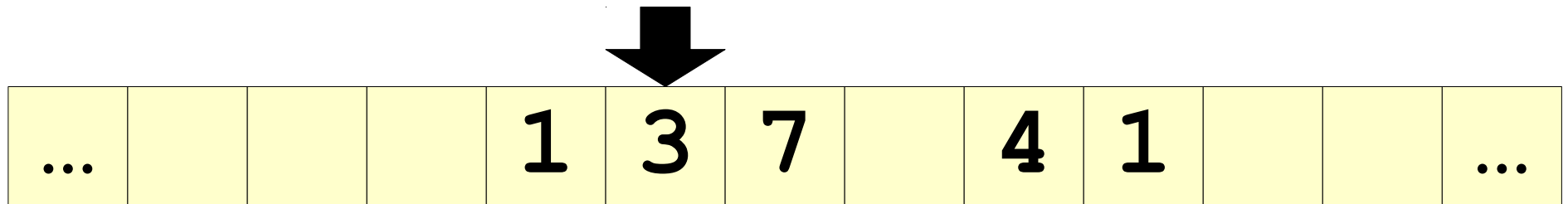
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



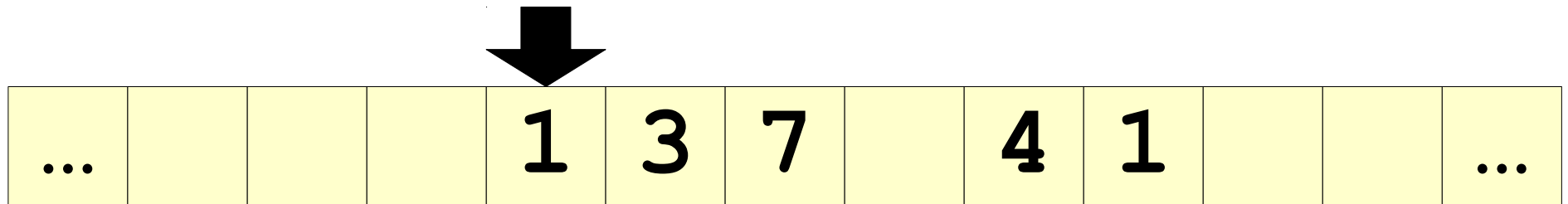
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



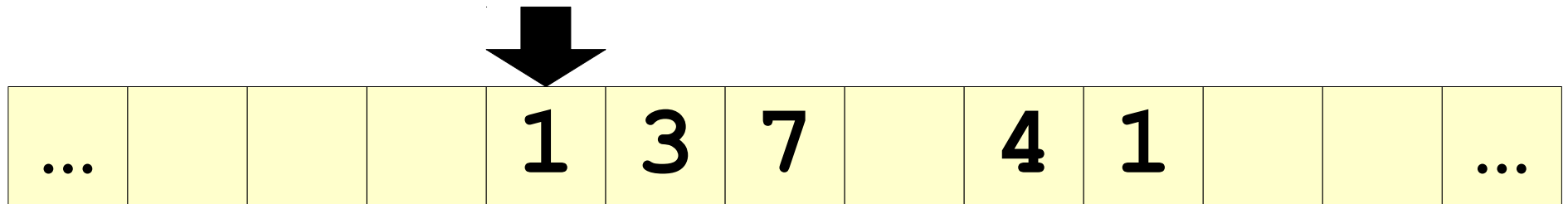
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



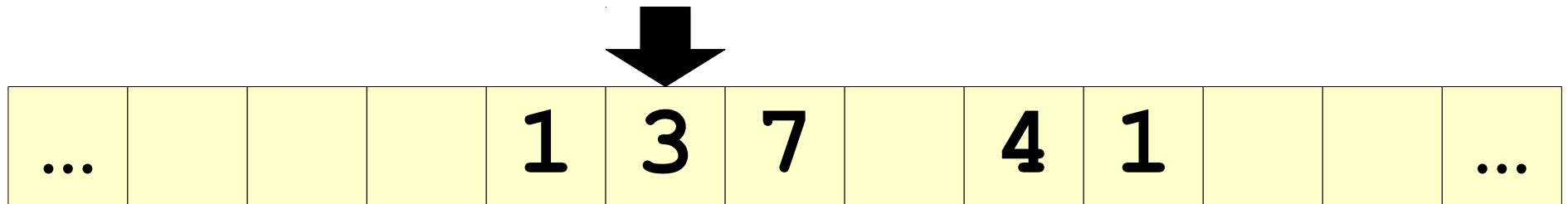
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



Using Our Subroutines

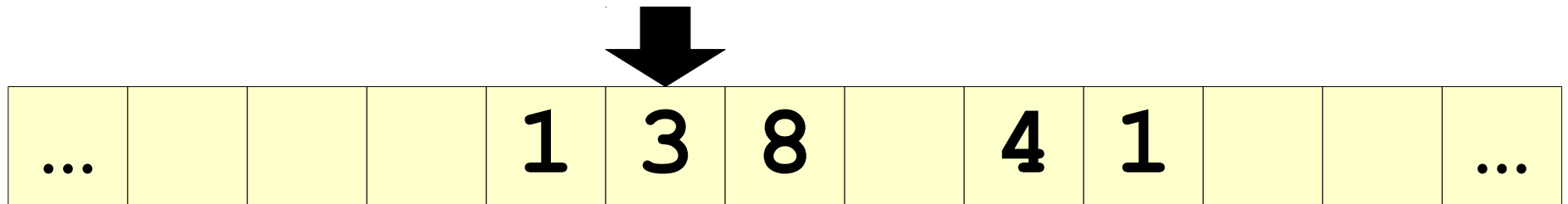
- We'll build our new machine using our existing increment and decrement subroutines:



...				1	3	7		4	1			...
-----	--	--	--	---	---	---	--	---	---	--	--	-----

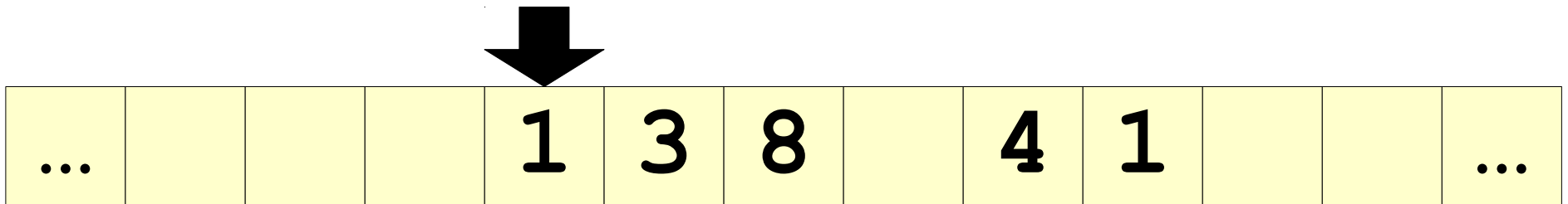
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



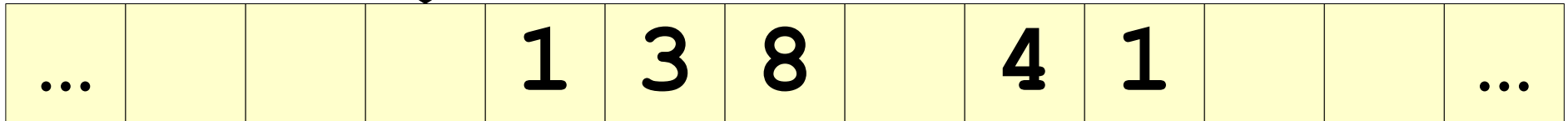
Using Our Subroutines

- We'll build our new machine using our existing increment and decrement subroutines:



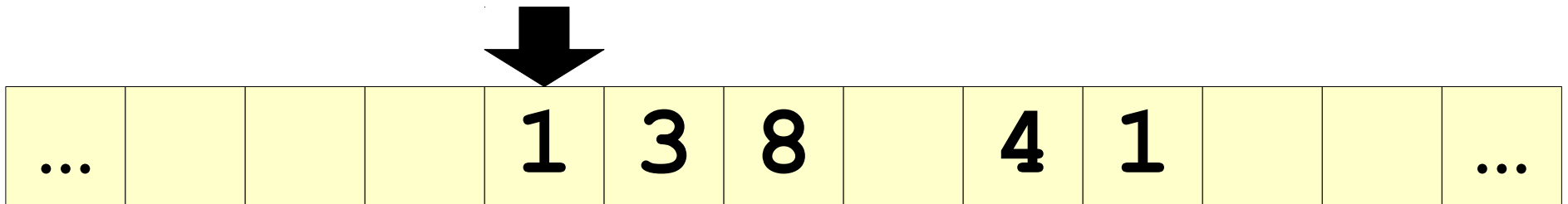
Using Our Subroutines

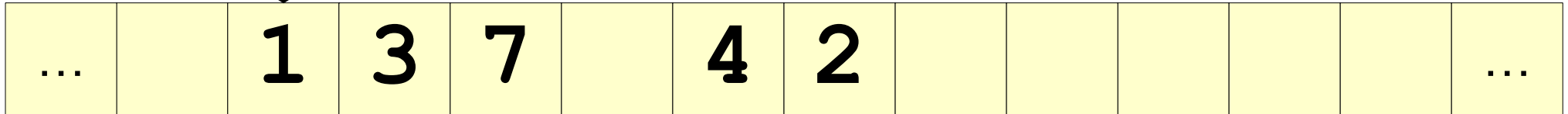
- We'll build our new machine using our existing increment and decrement subroutines:

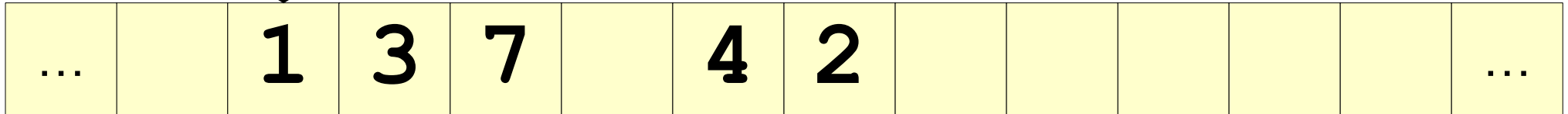
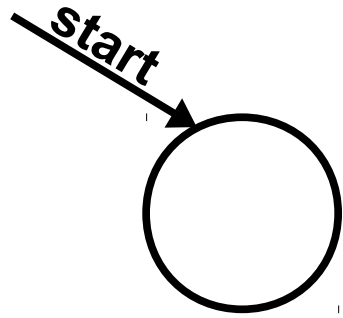


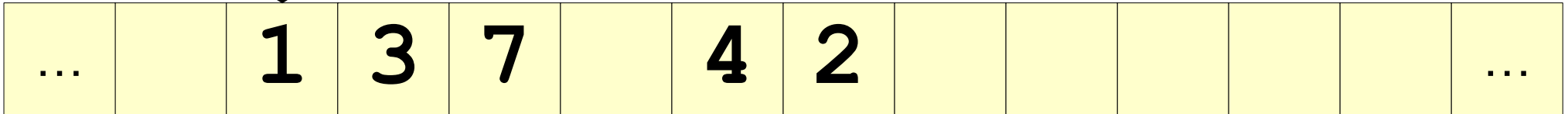
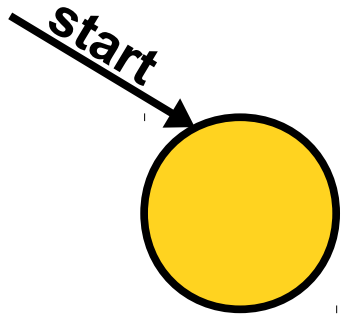
Using Our Subroutines

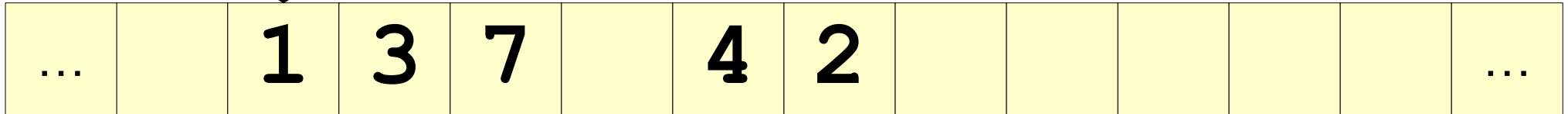
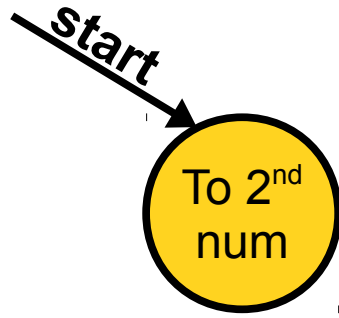
- We'll build our new machine using our existing increment and decrement subroutines:

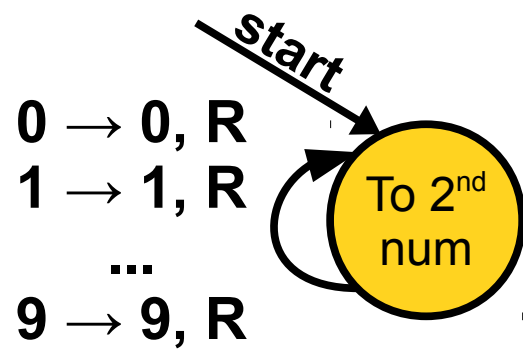




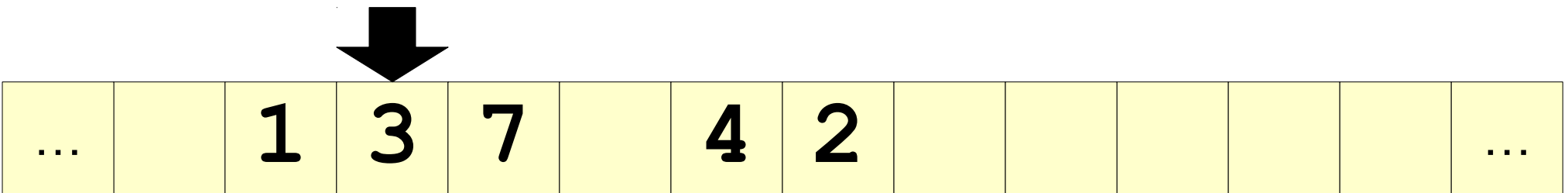
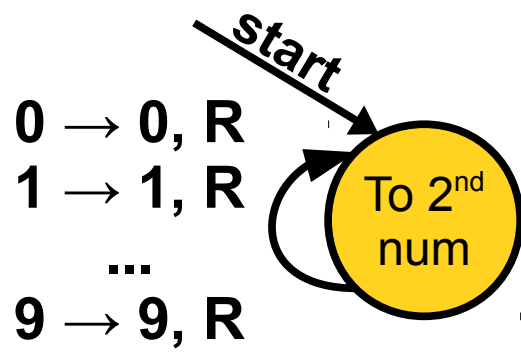


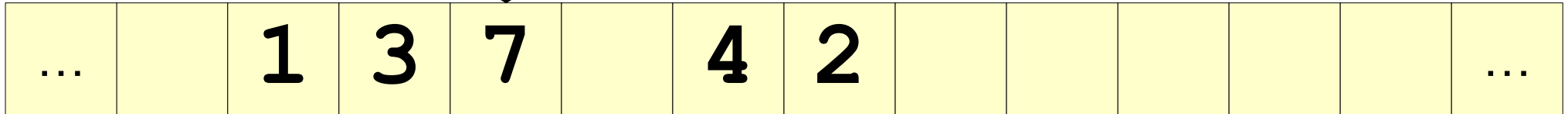
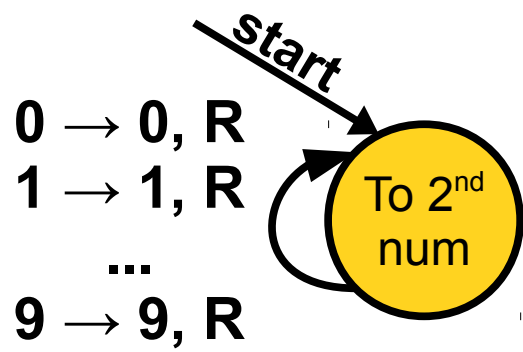


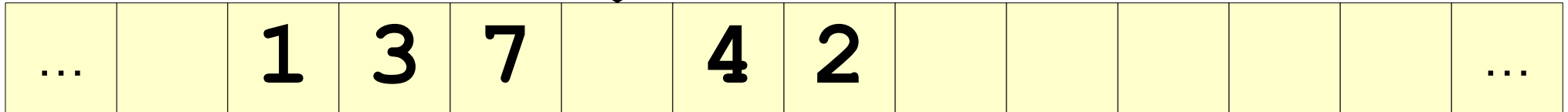
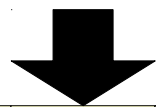
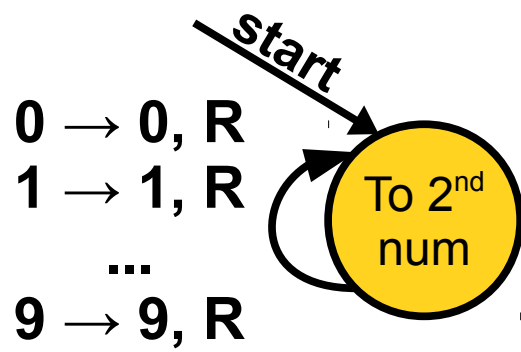


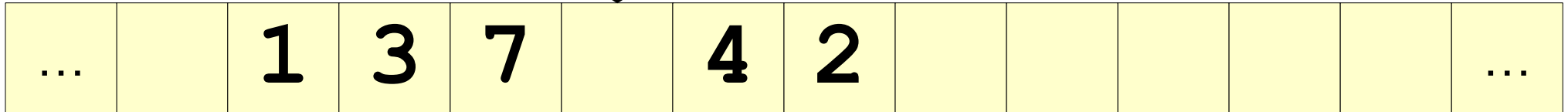
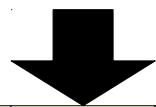
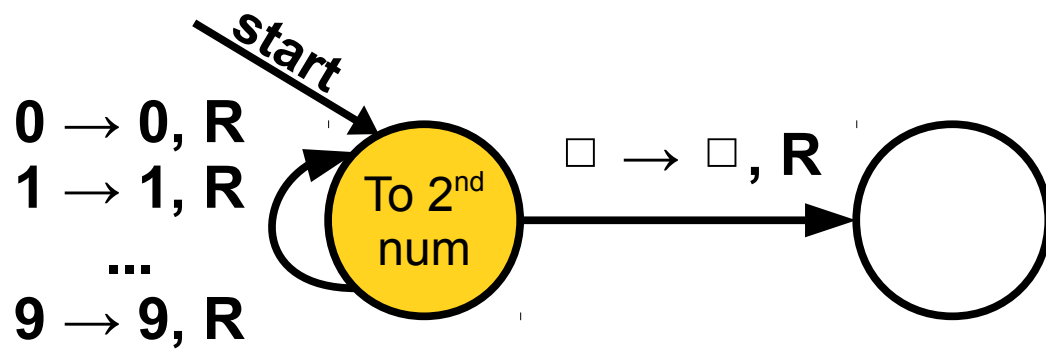


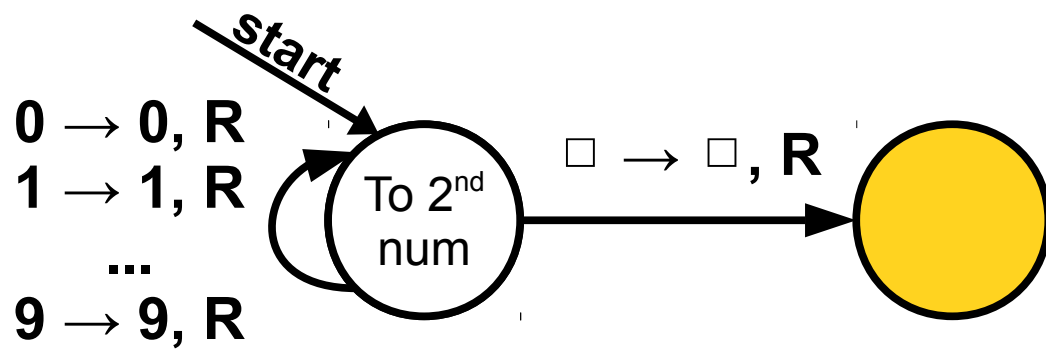
...		1	3	7		4	2						...
-----	--	---	---	---	--	---	---	--	--	--	--	--	-----

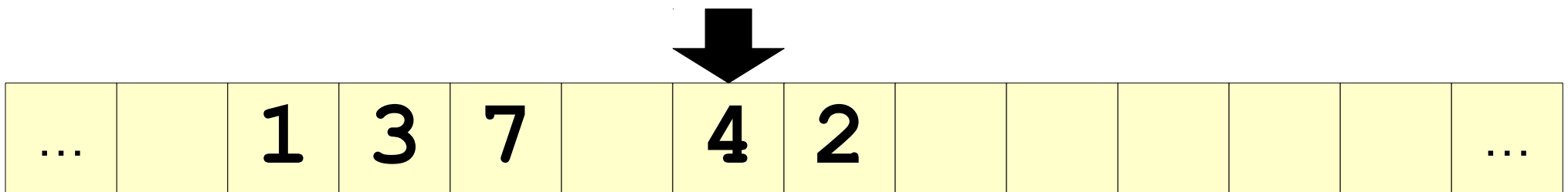
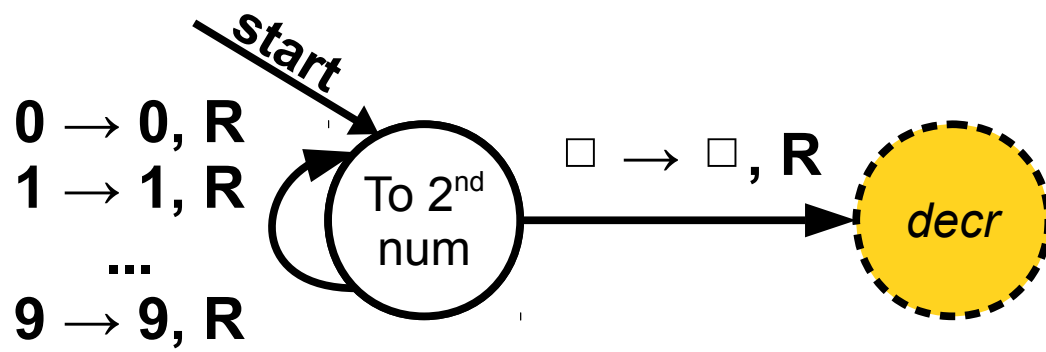


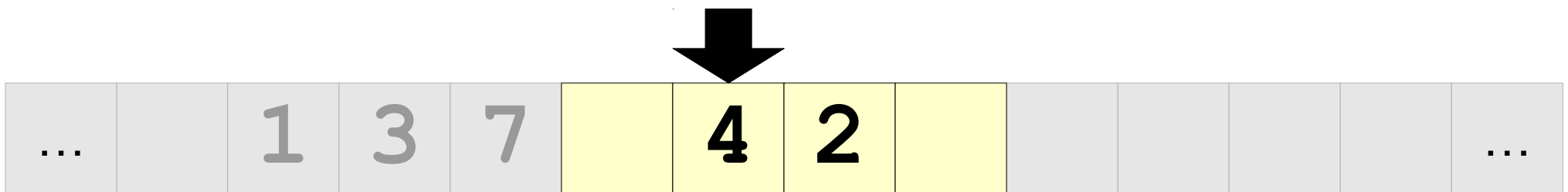
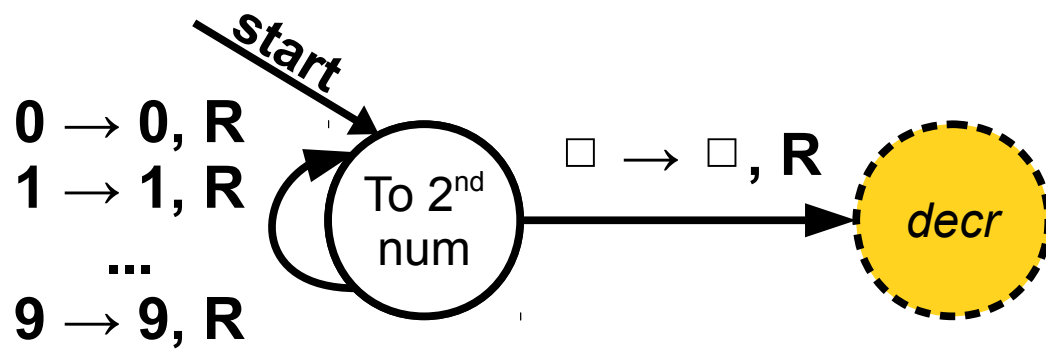


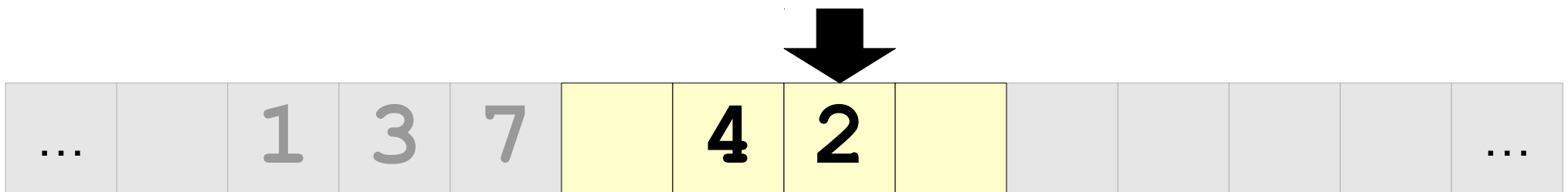
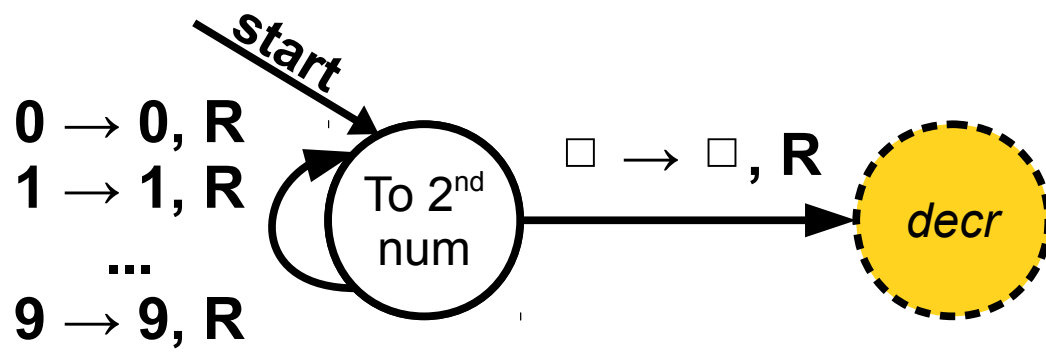


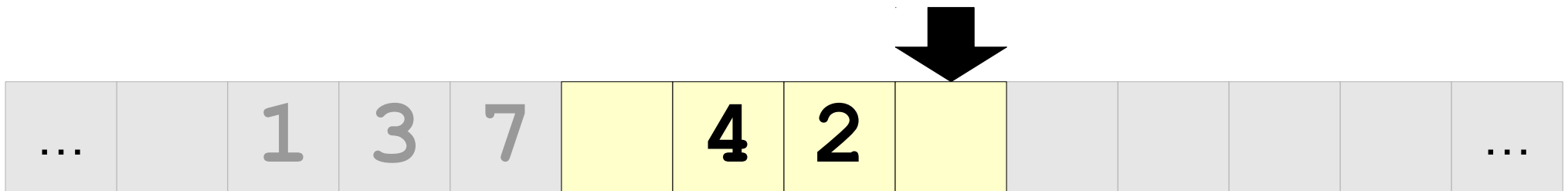
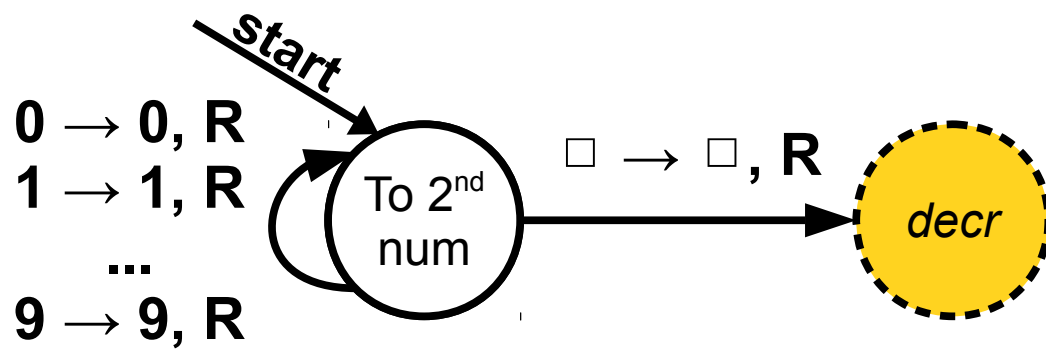


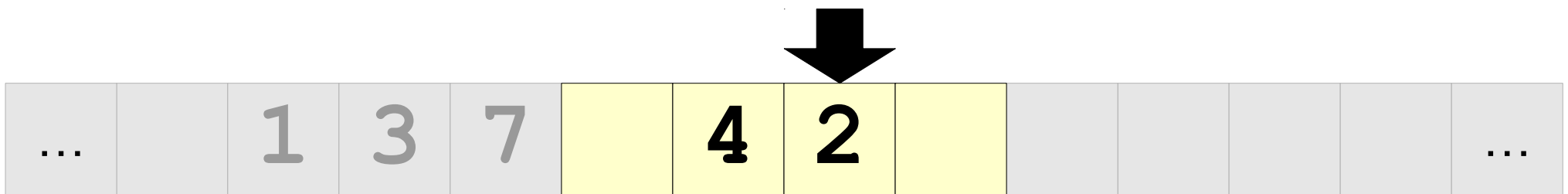
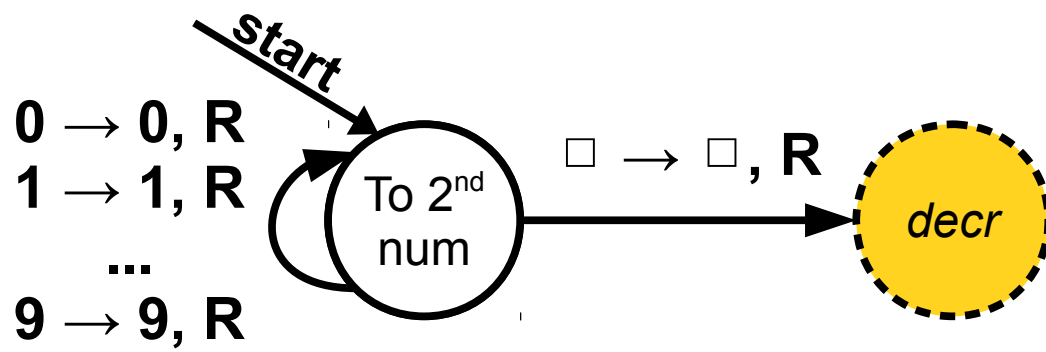


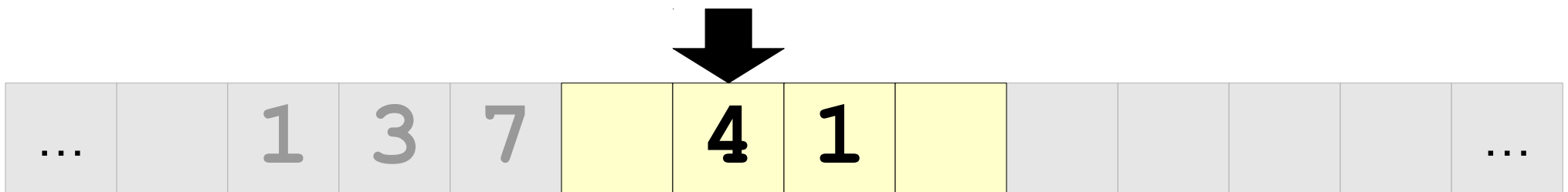
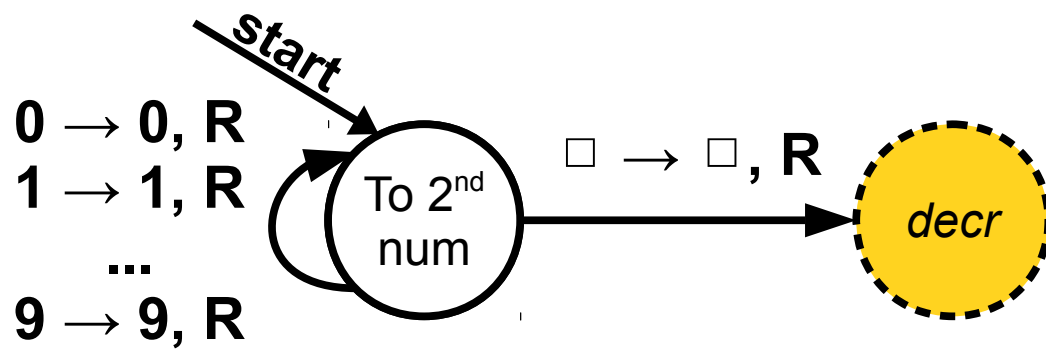


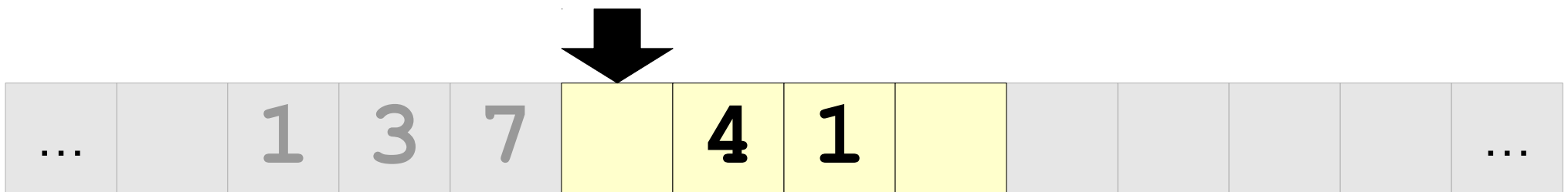
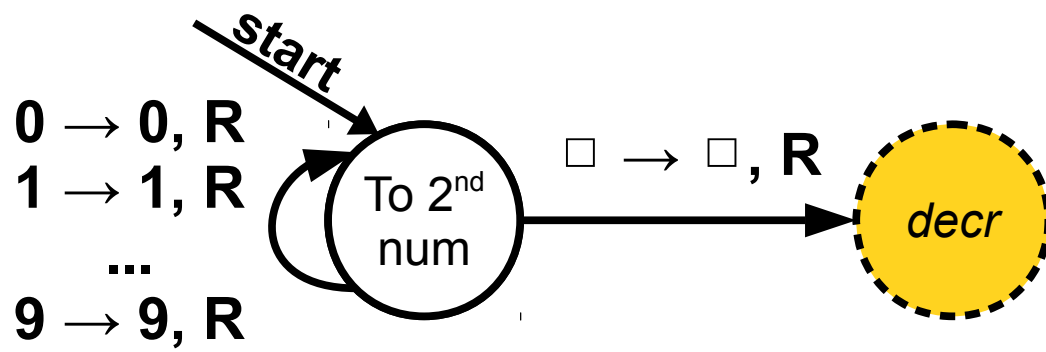


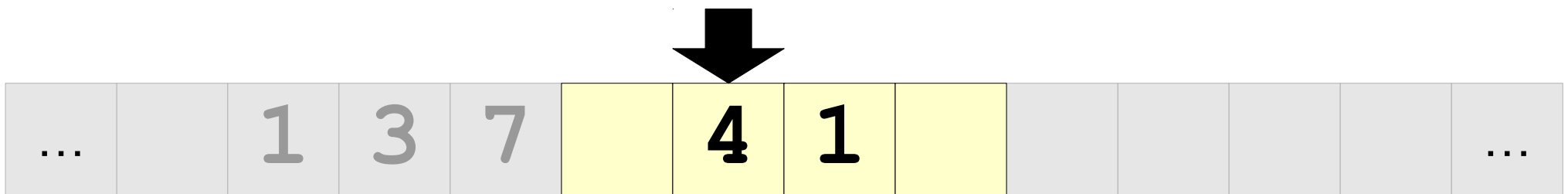
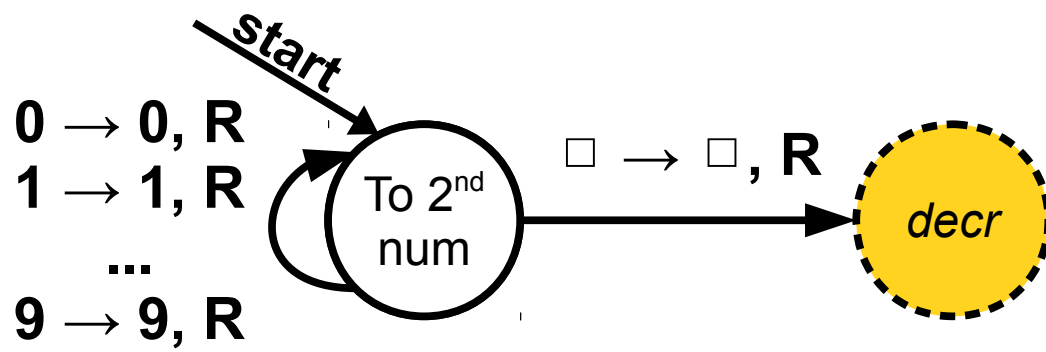


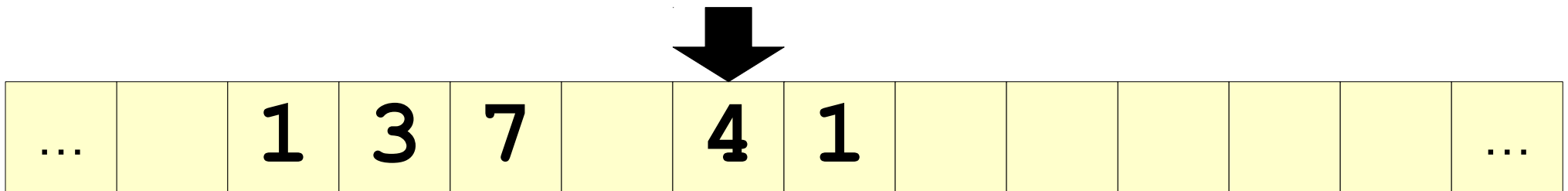
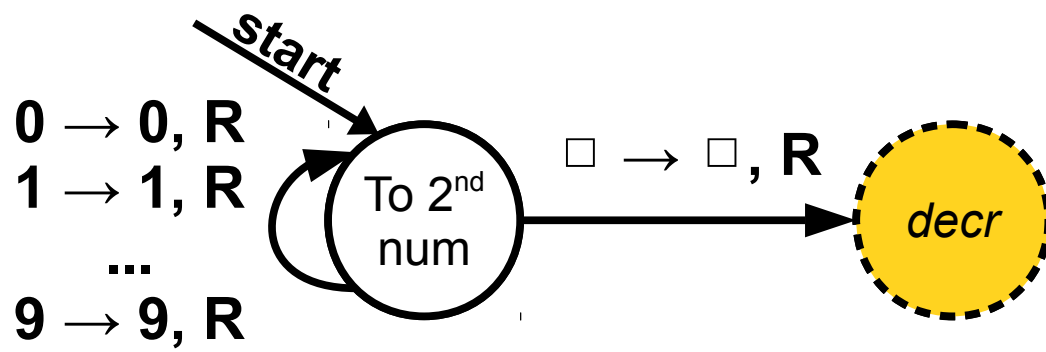


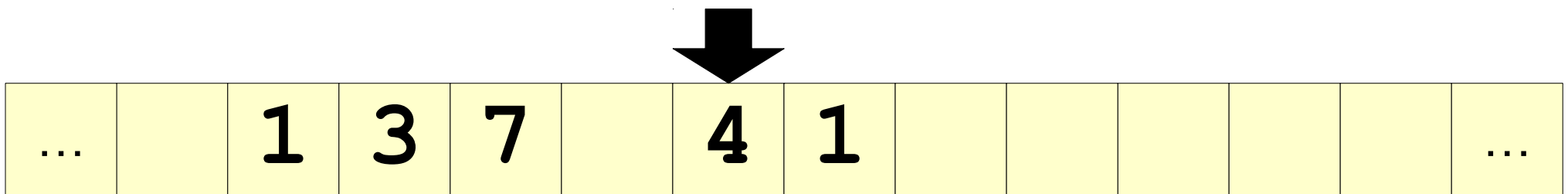
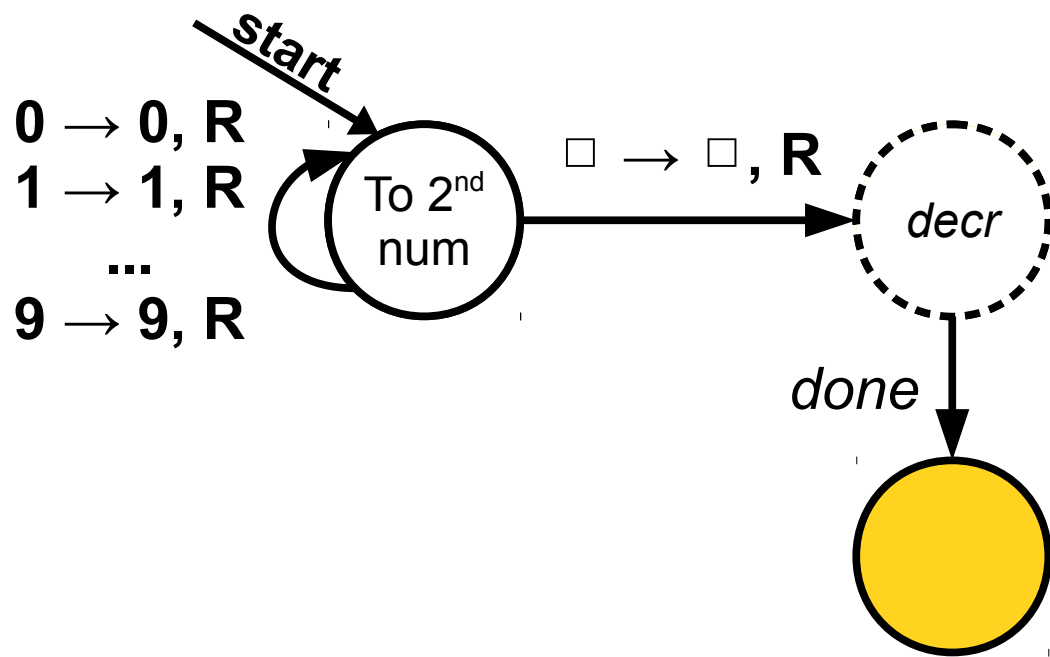


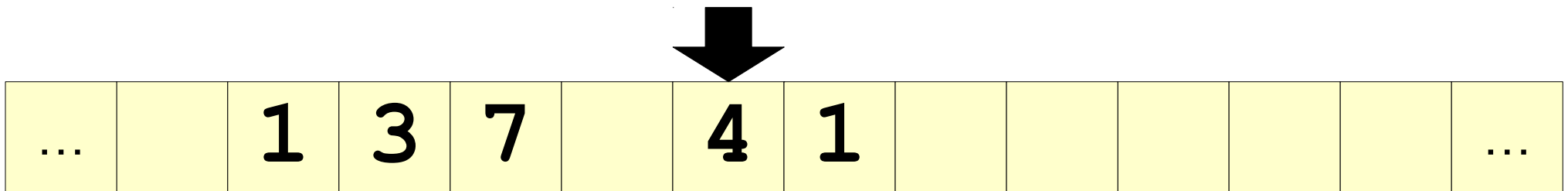
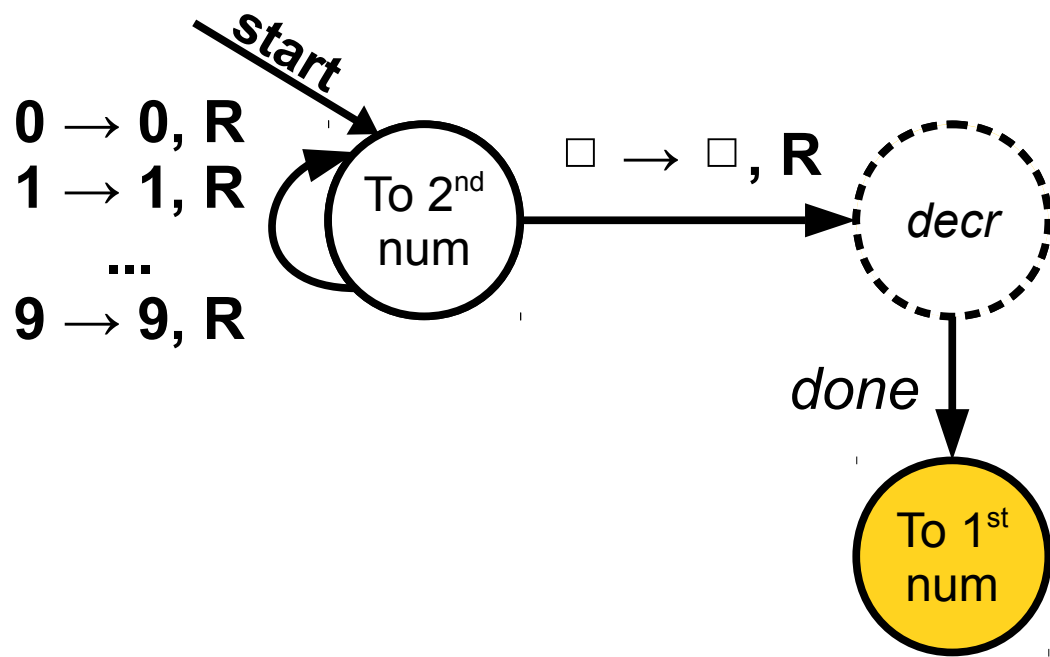


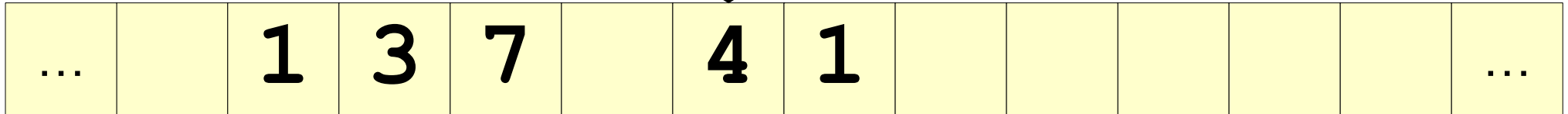
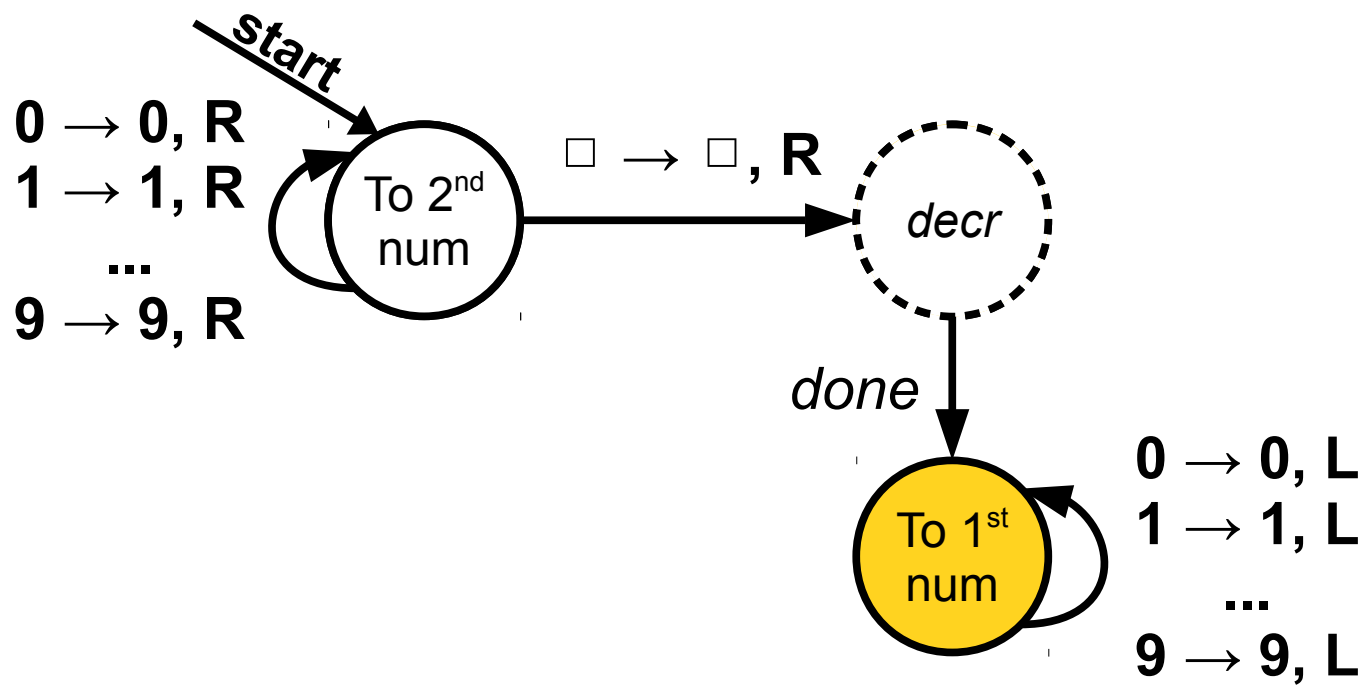


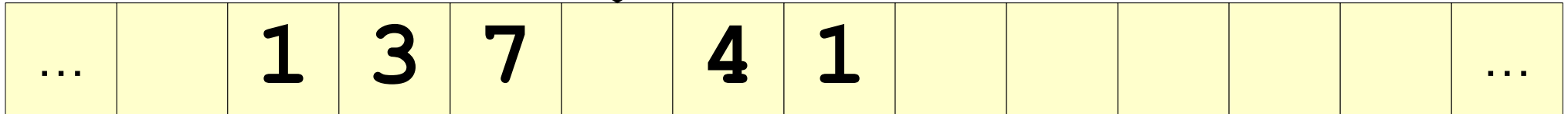
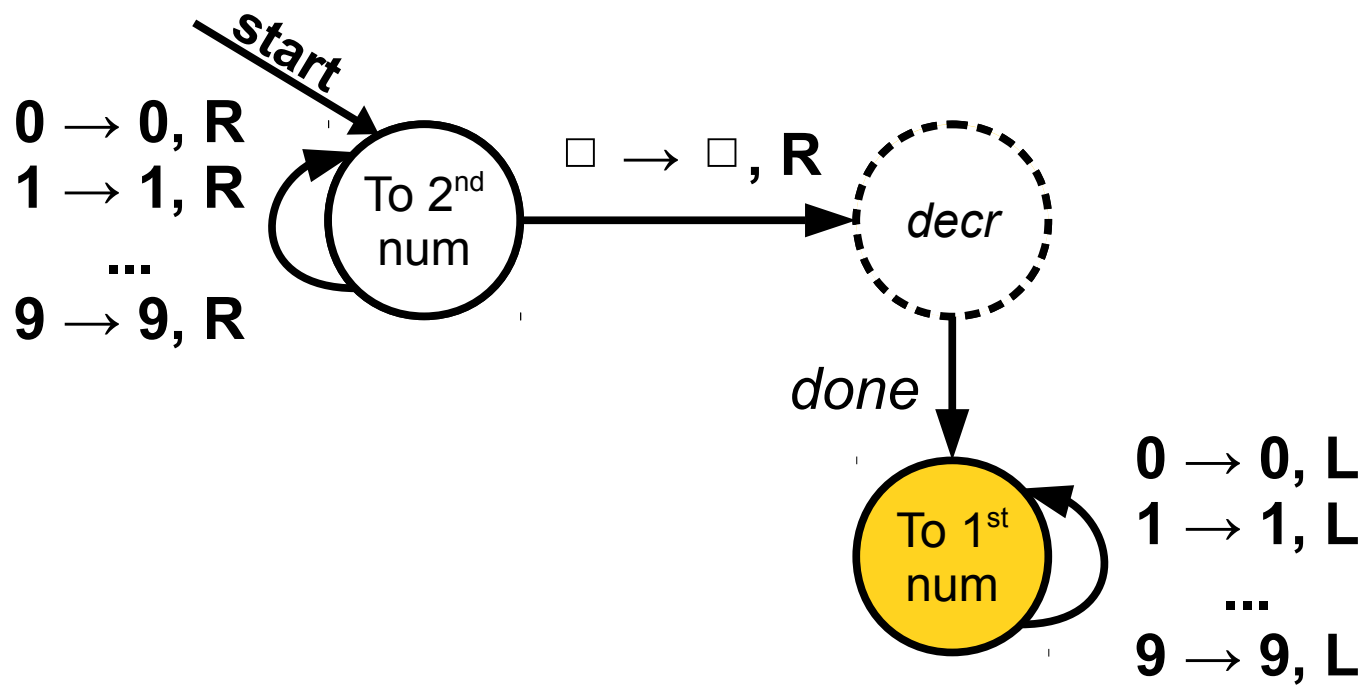


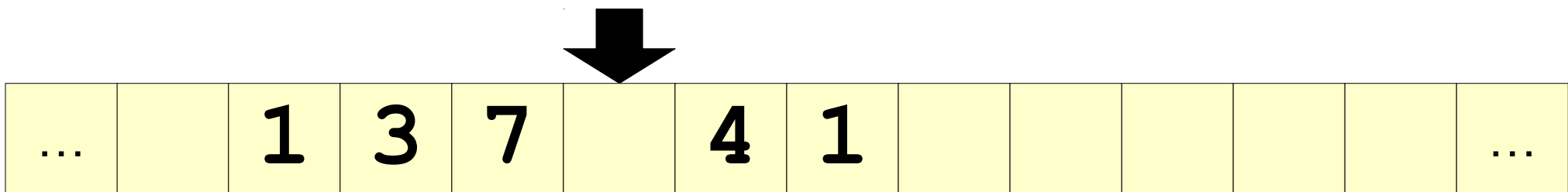
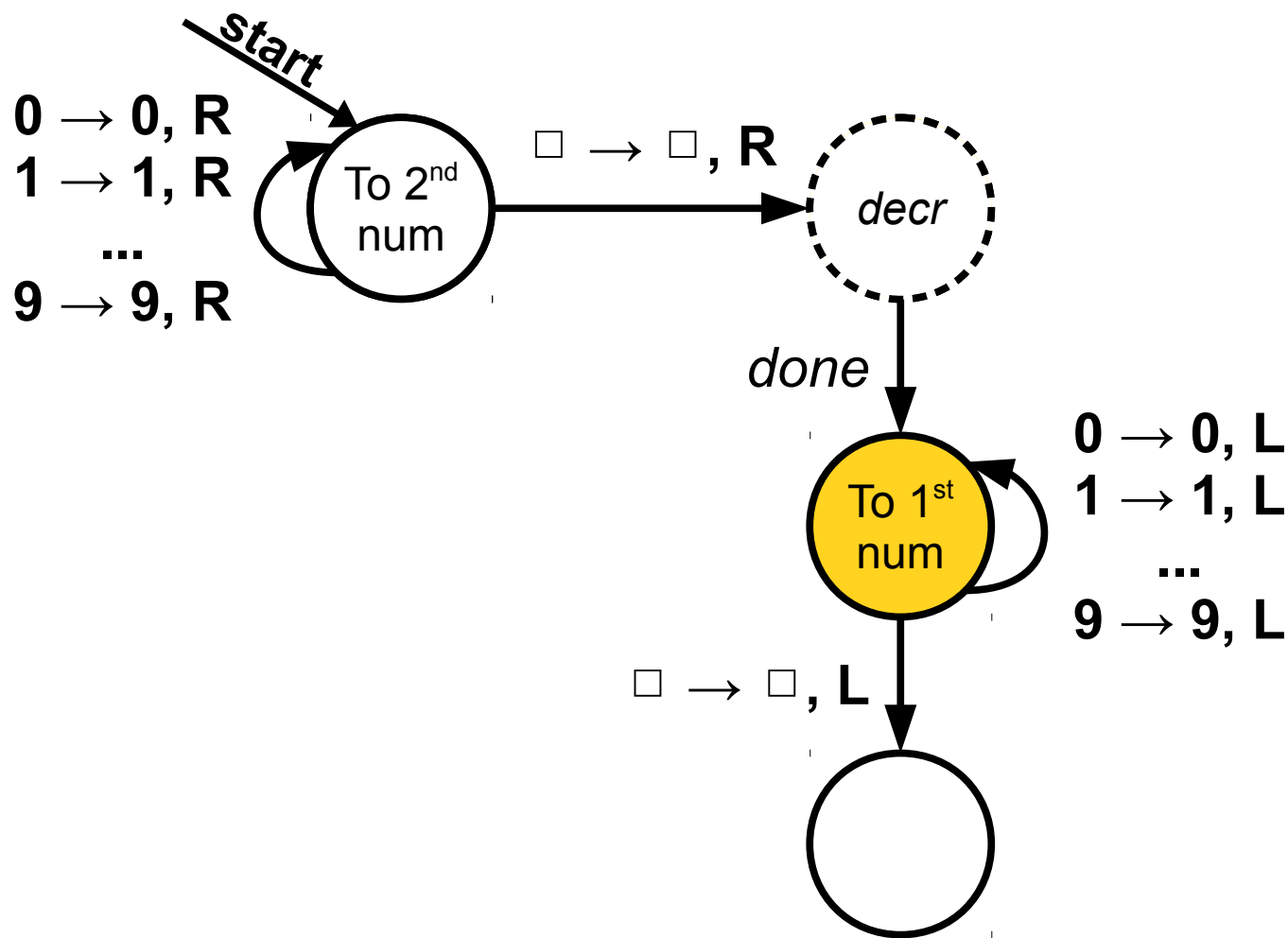


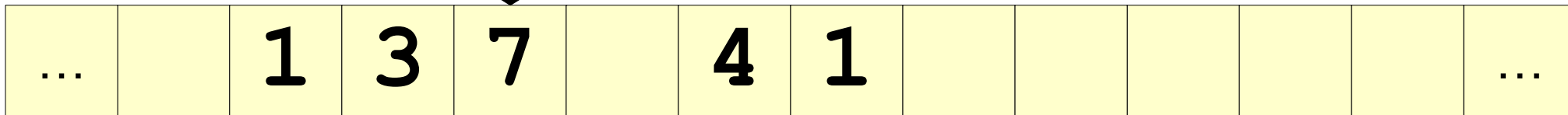
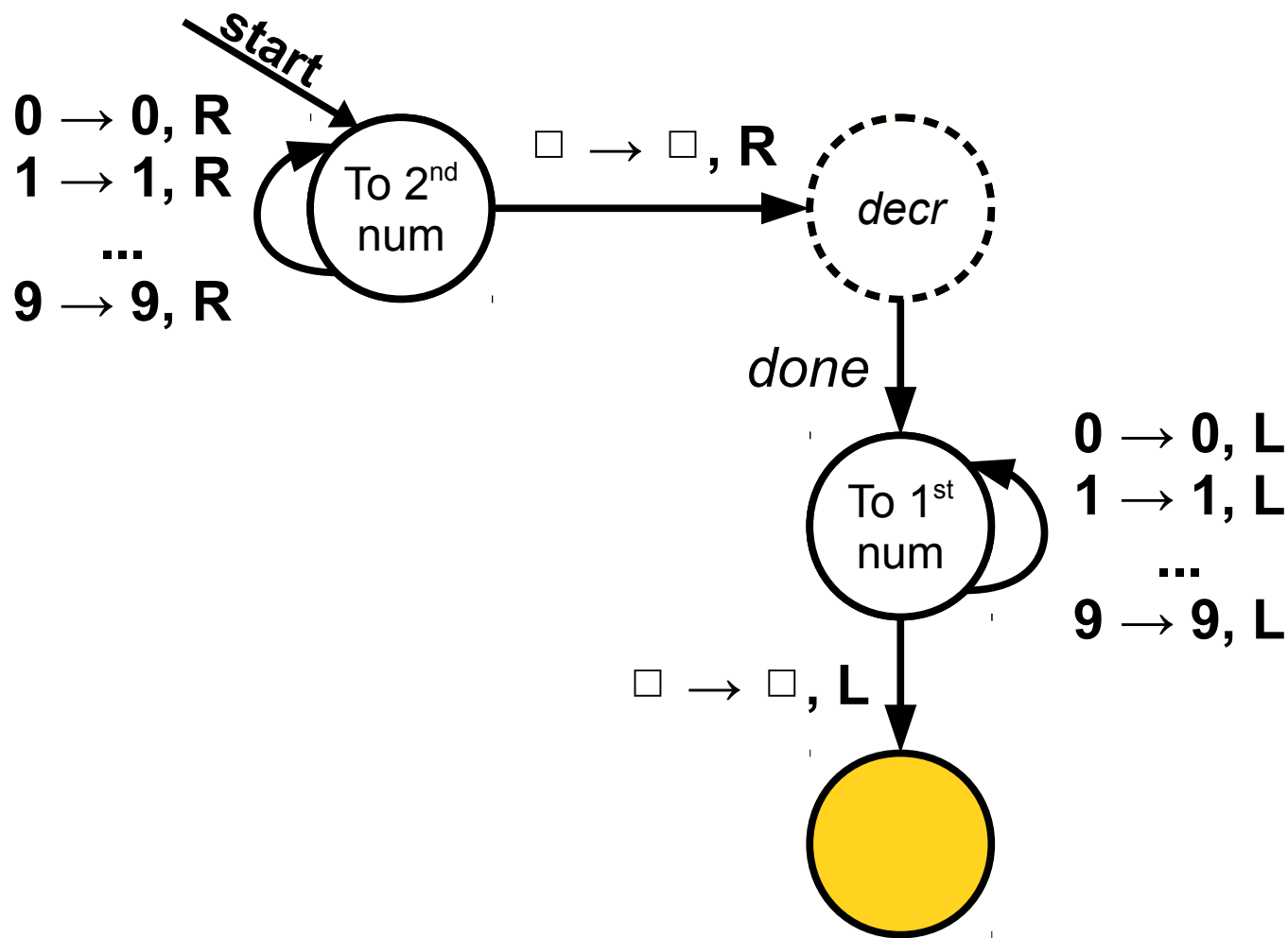


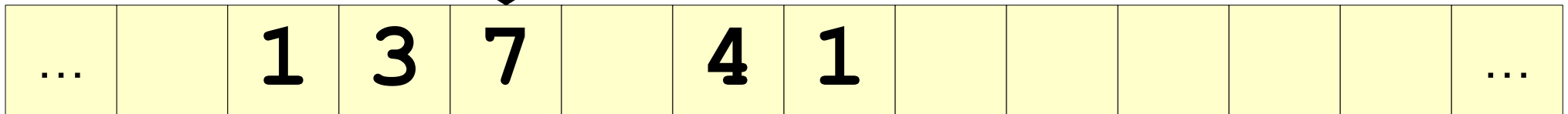
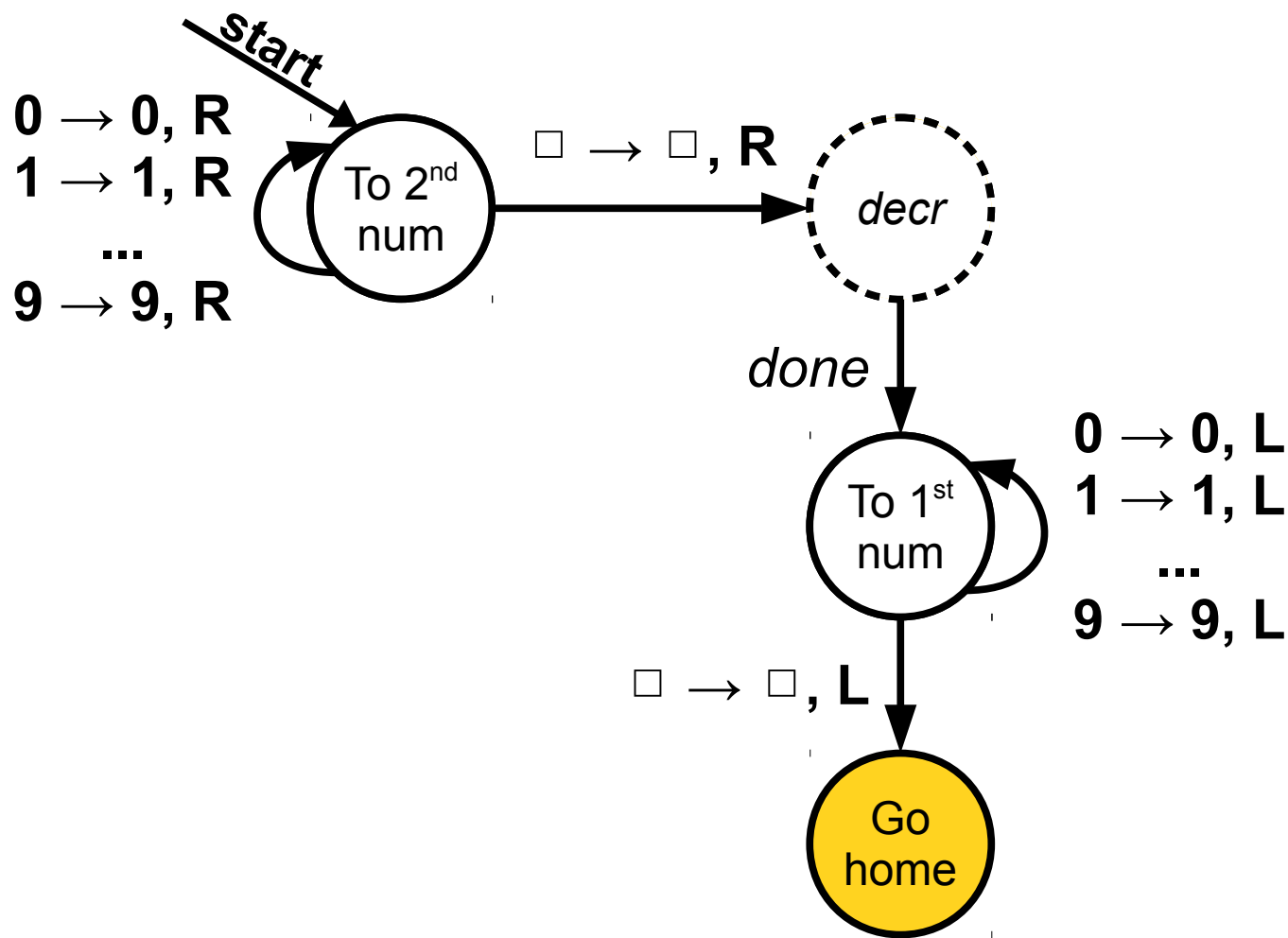


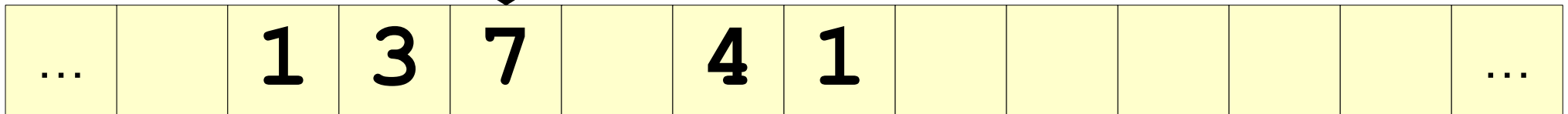
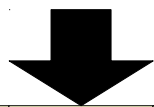
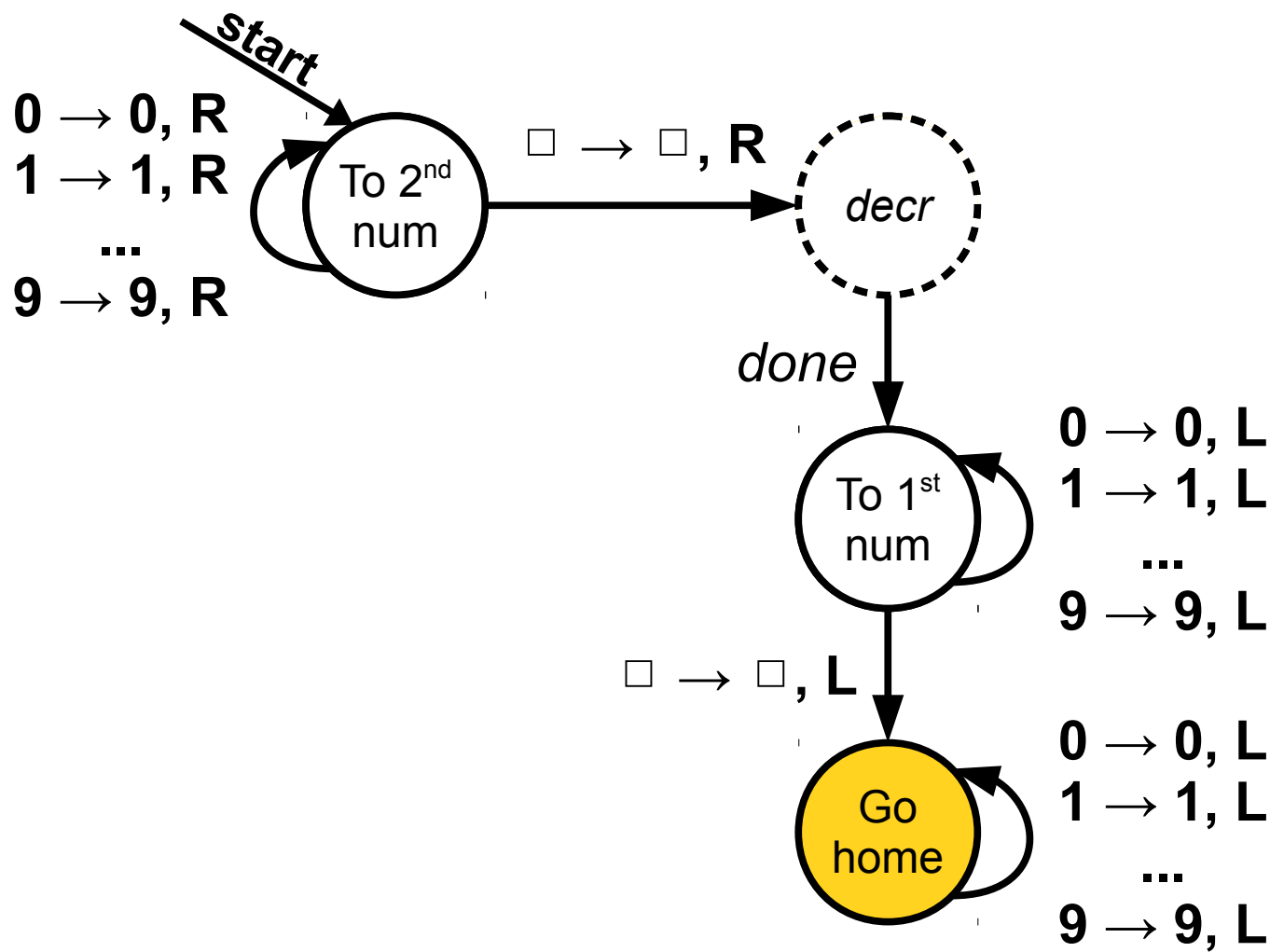


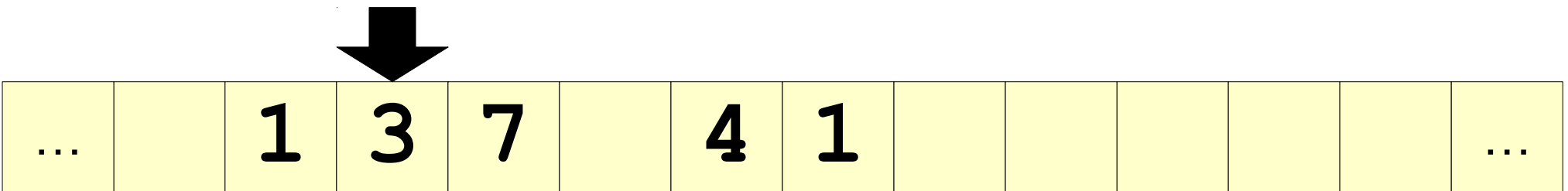
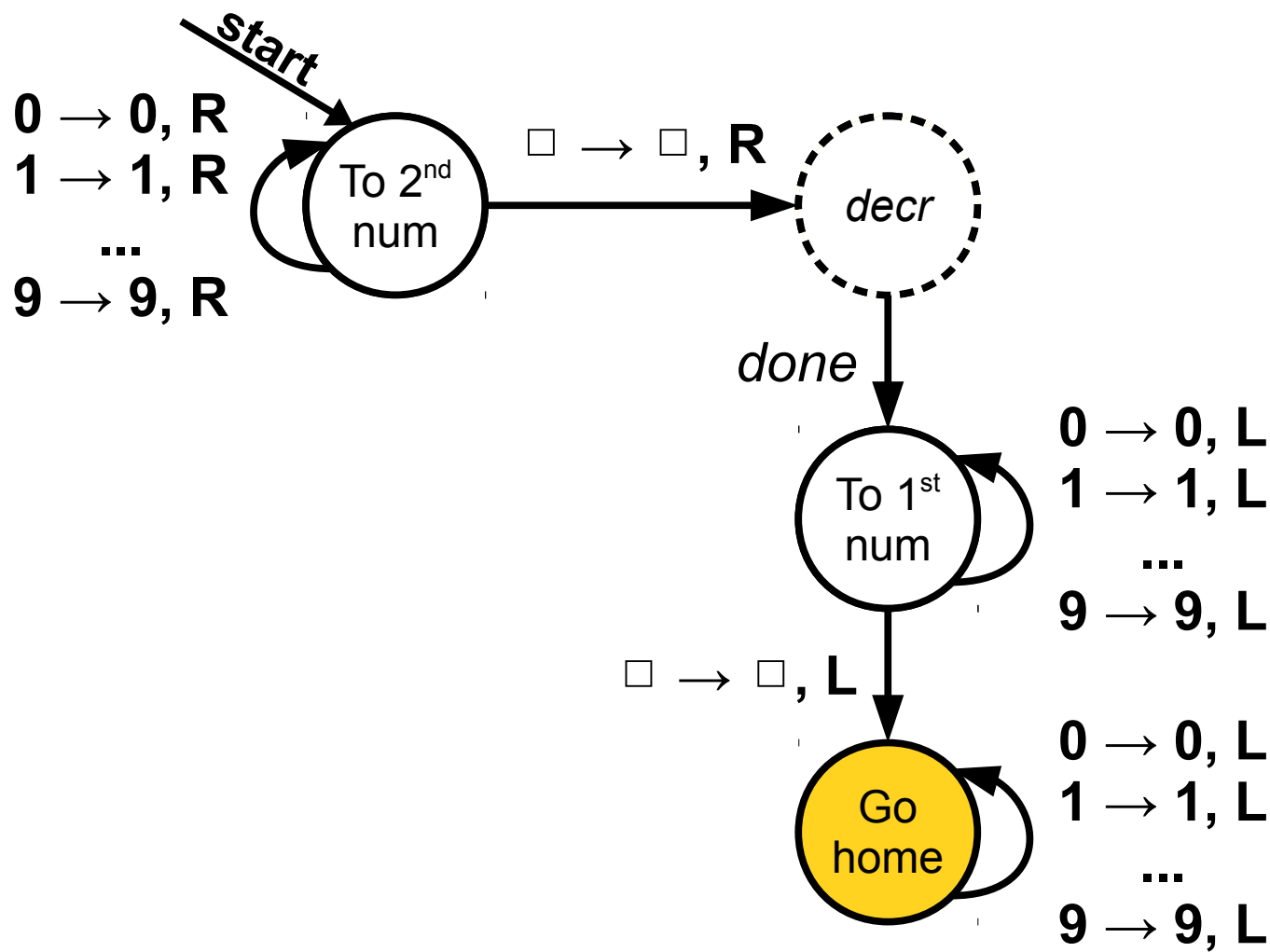


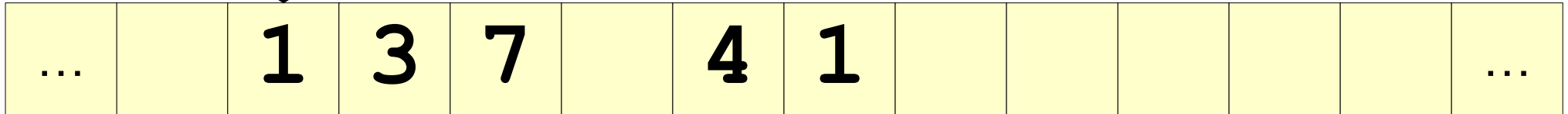
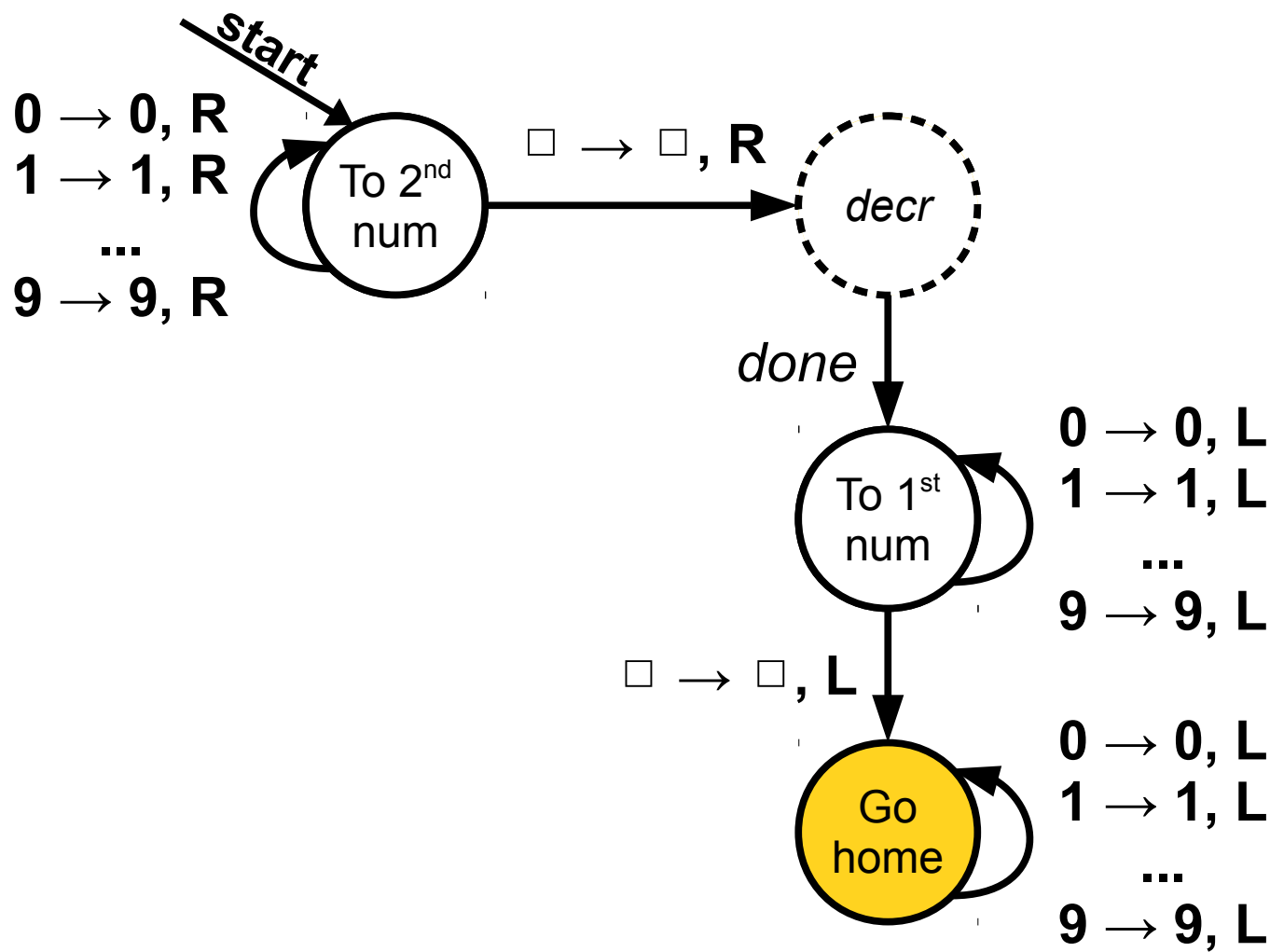


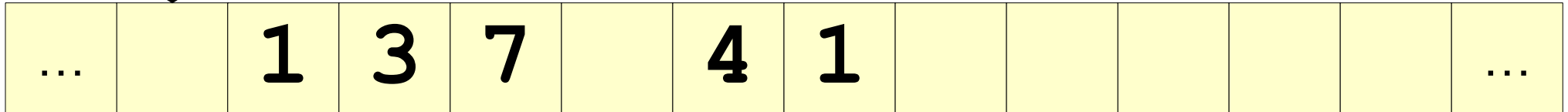
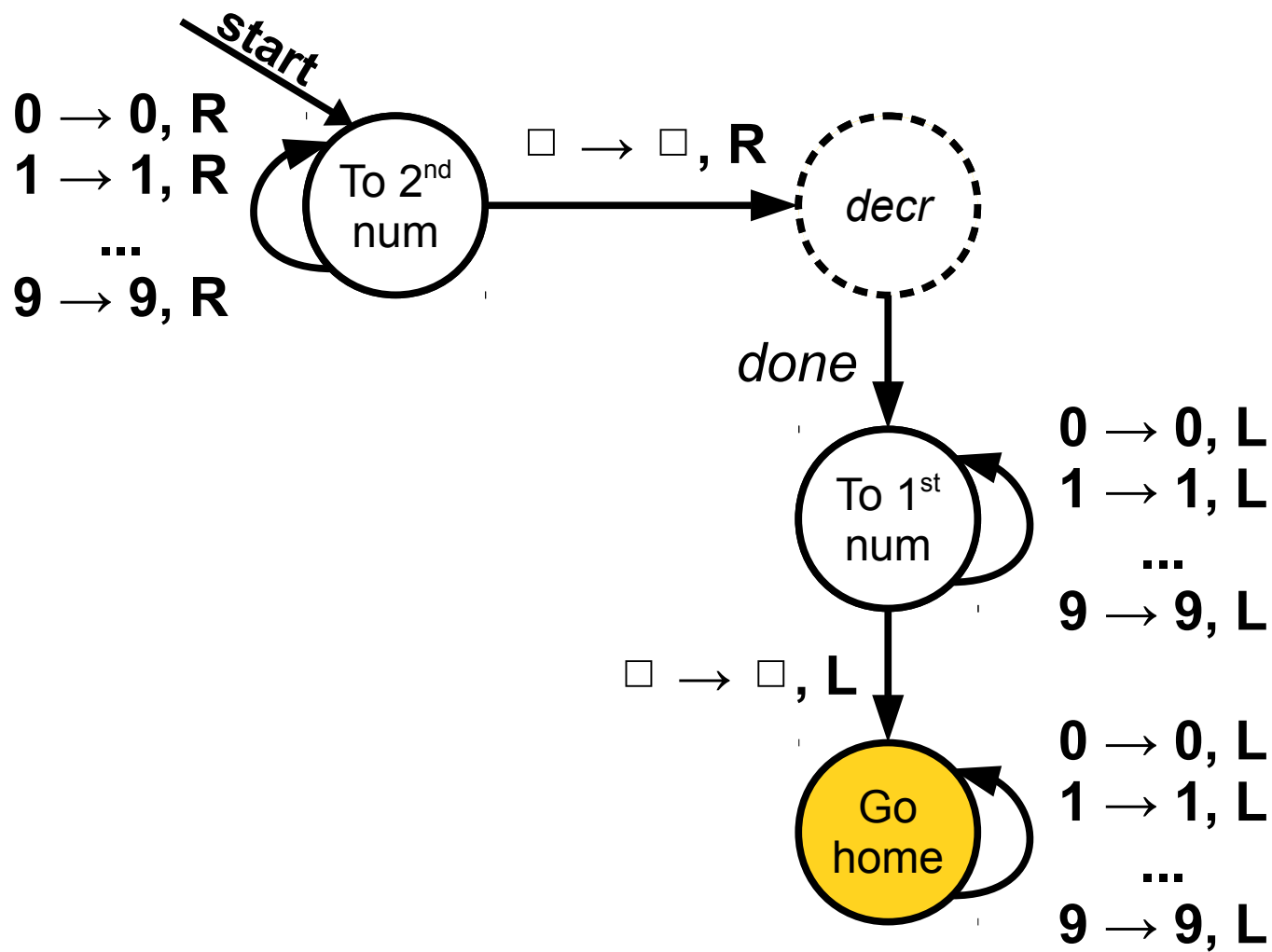


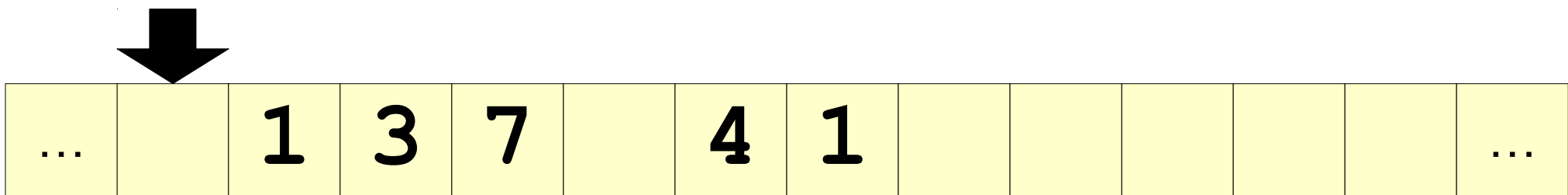
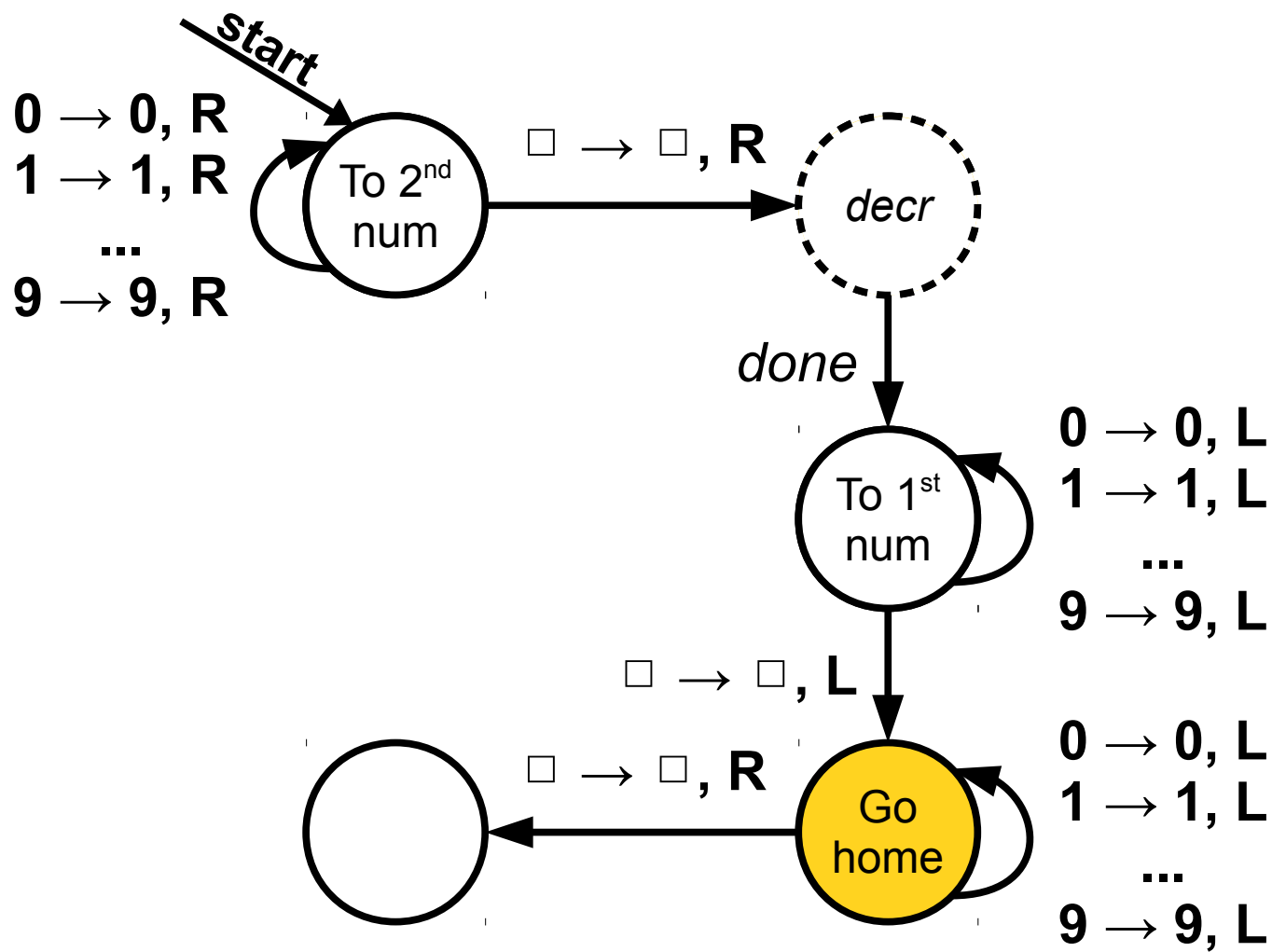


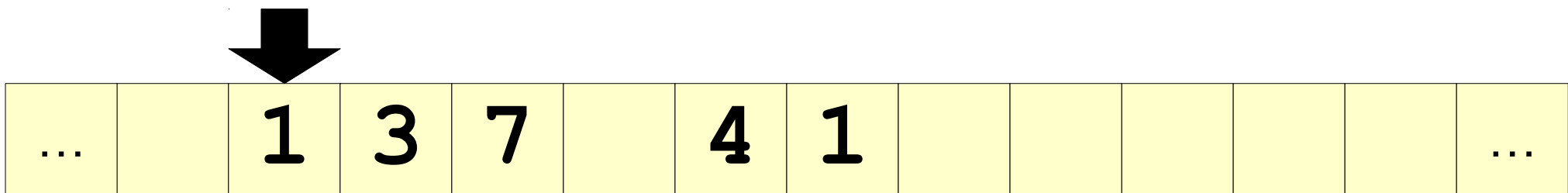
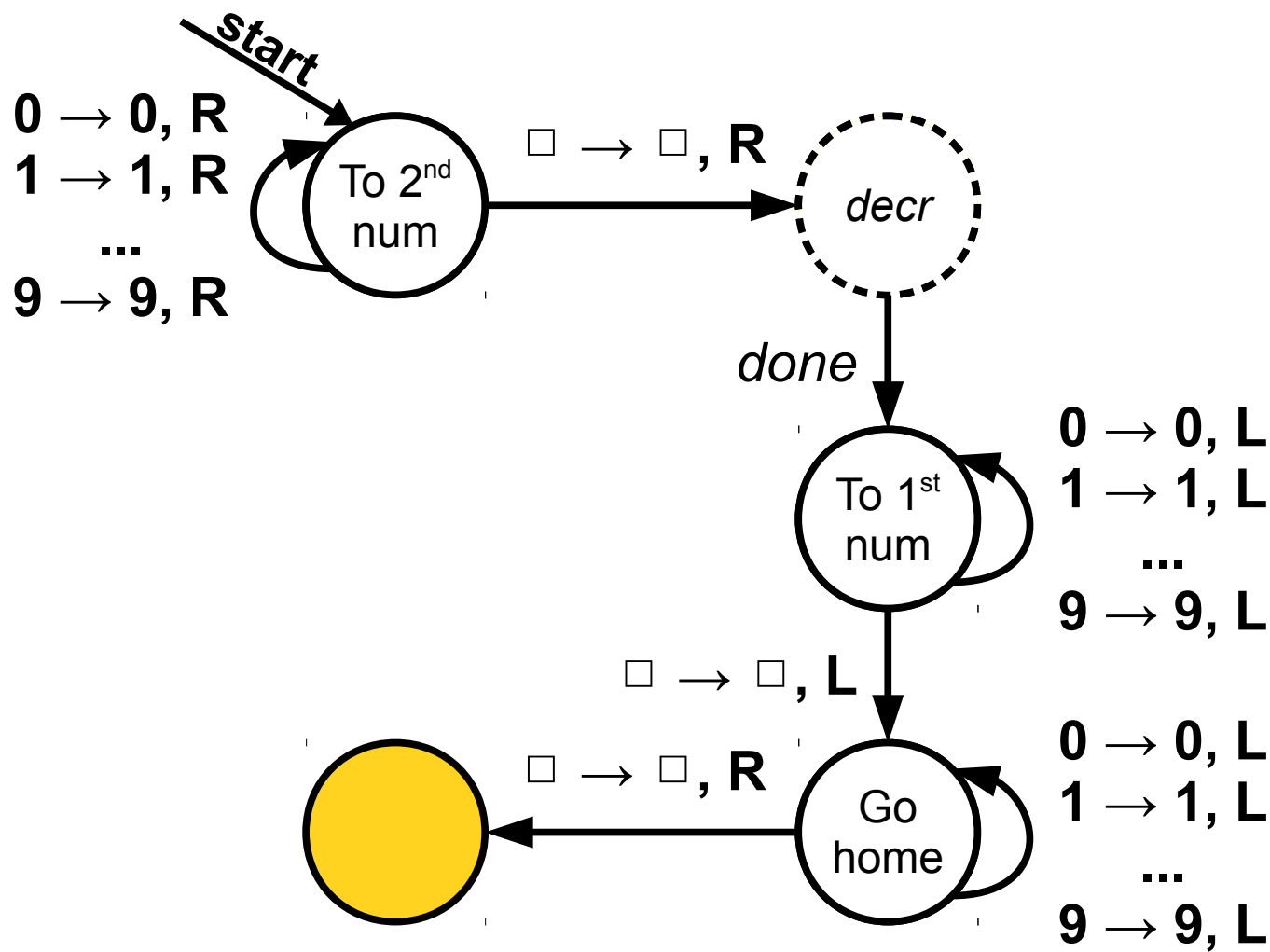


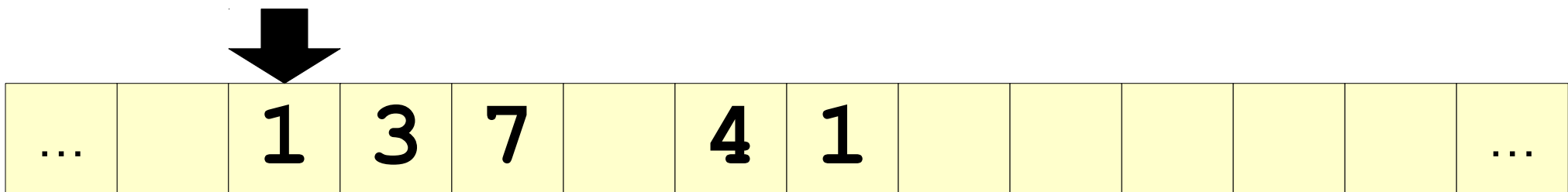
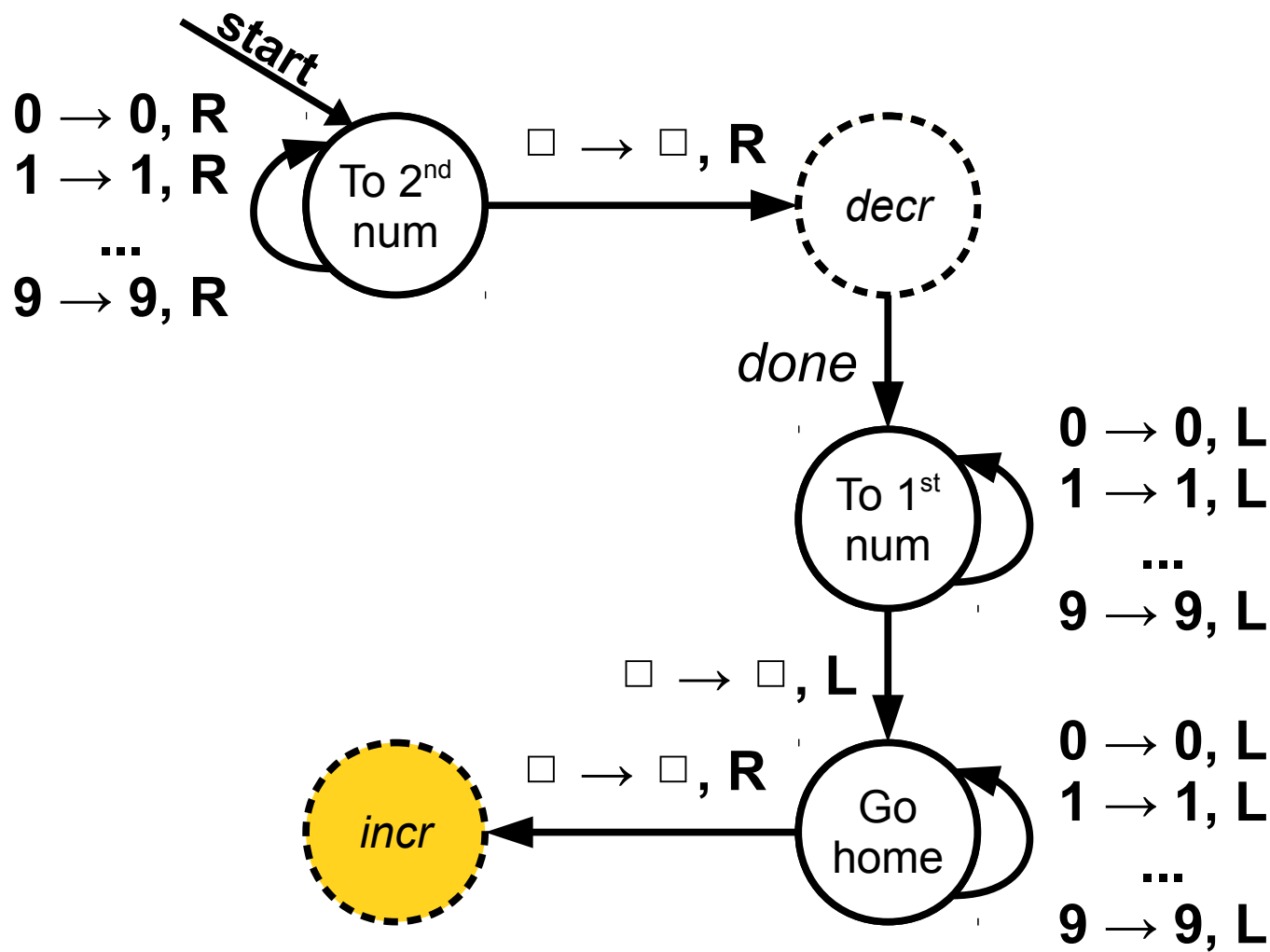


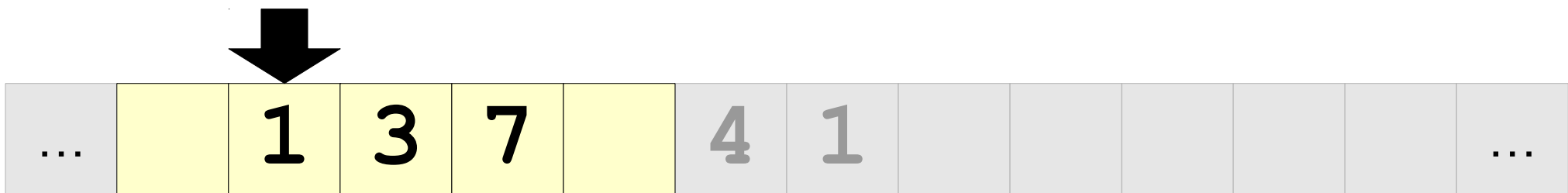
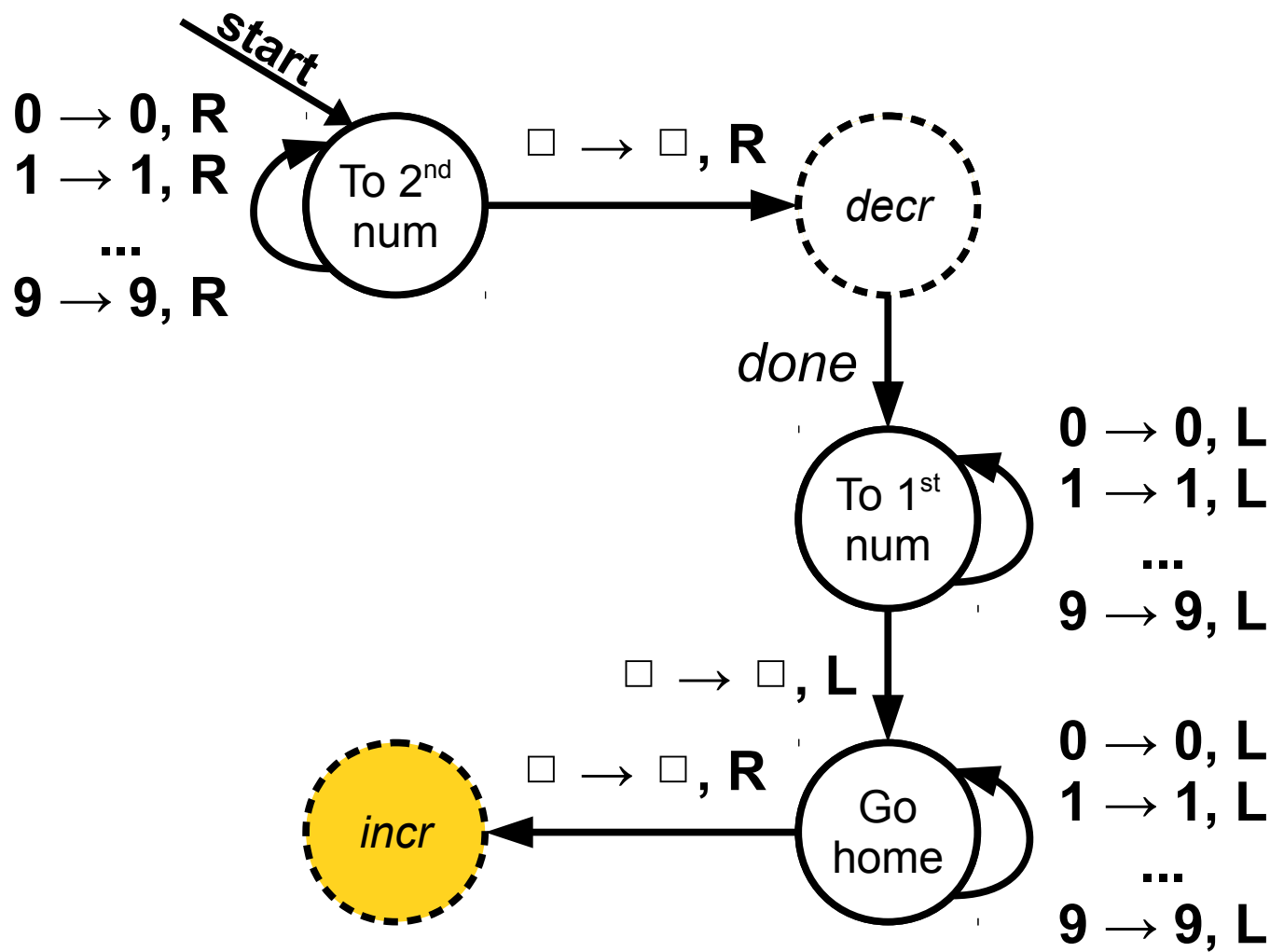


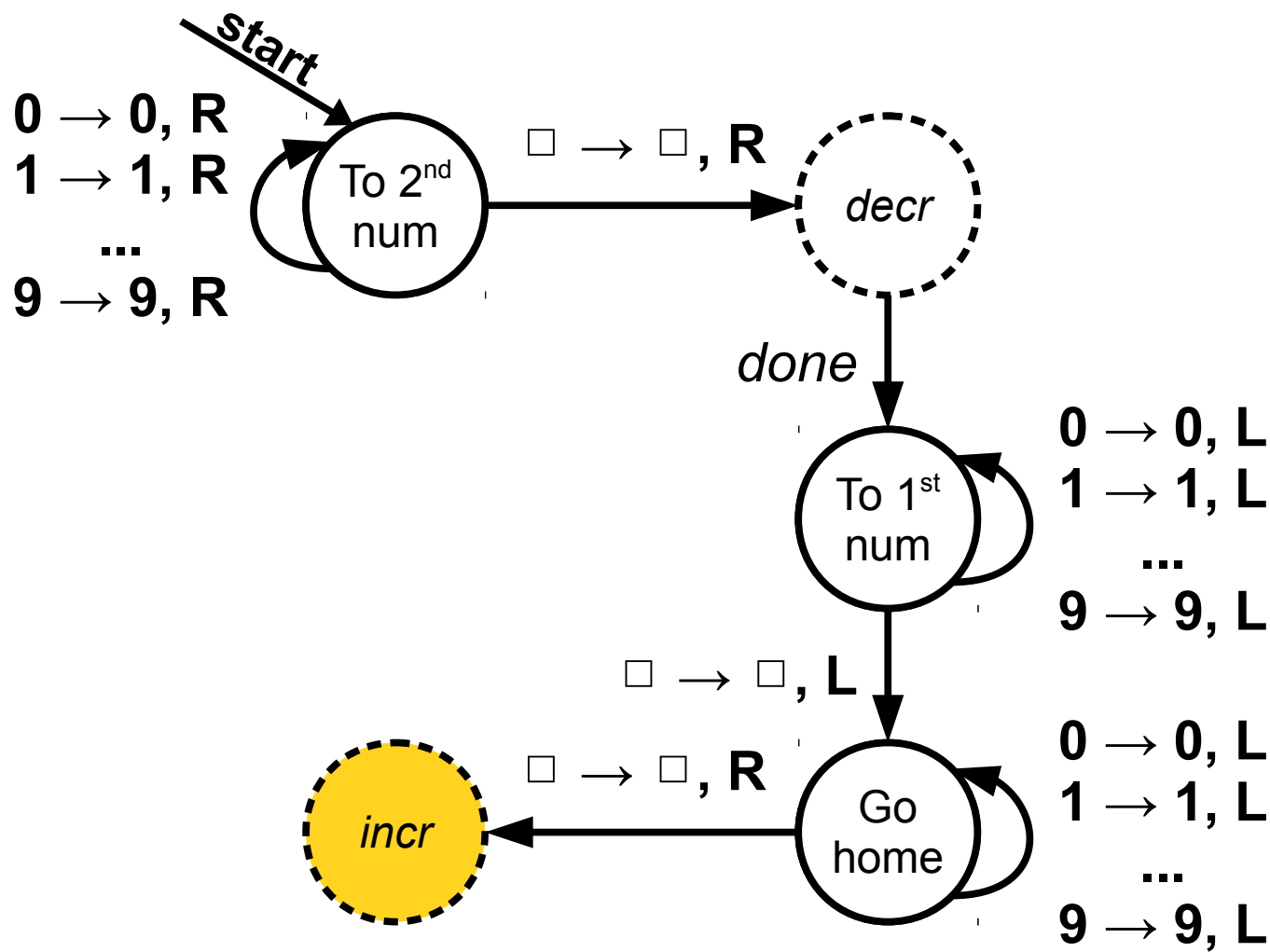


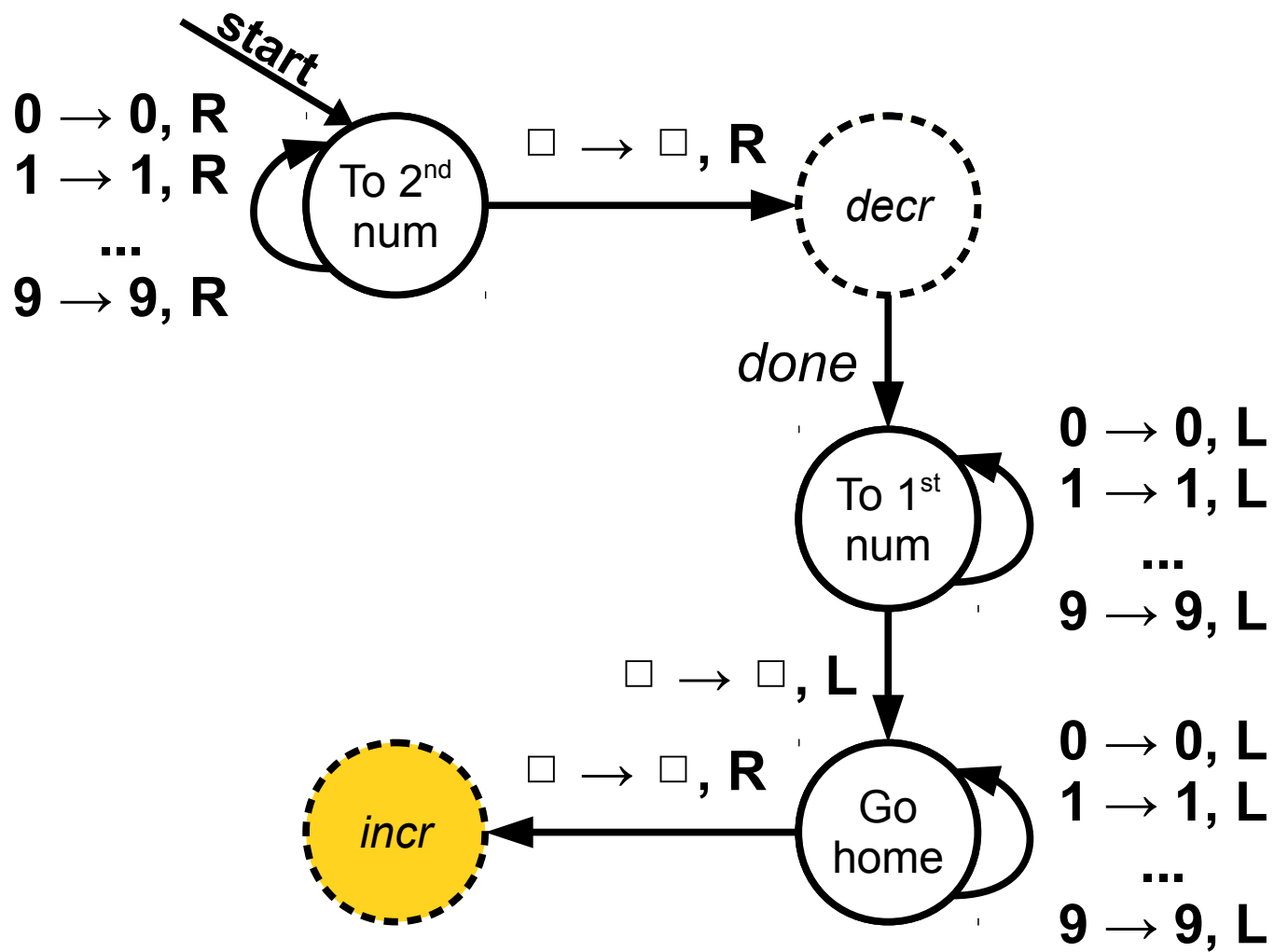


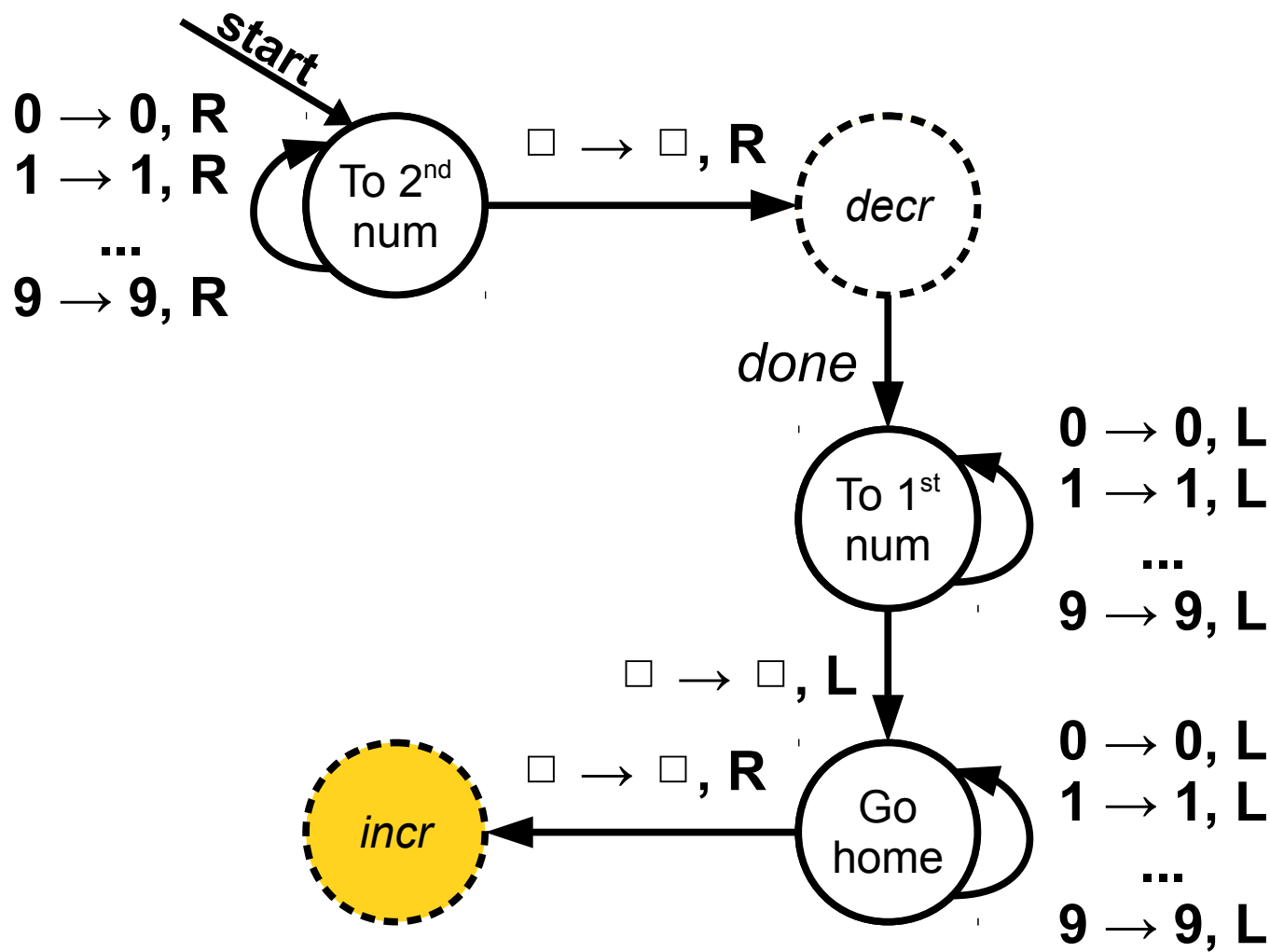


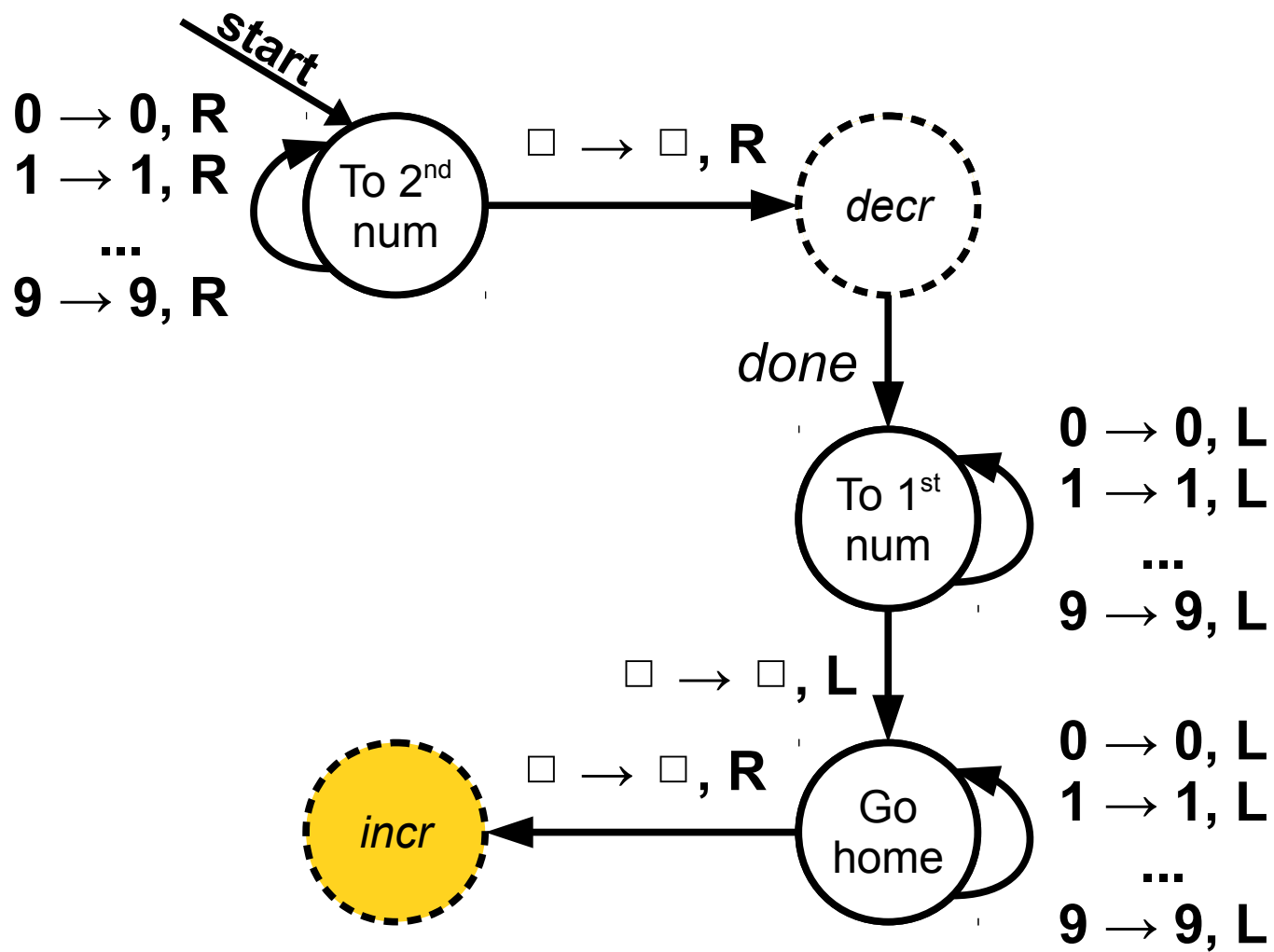


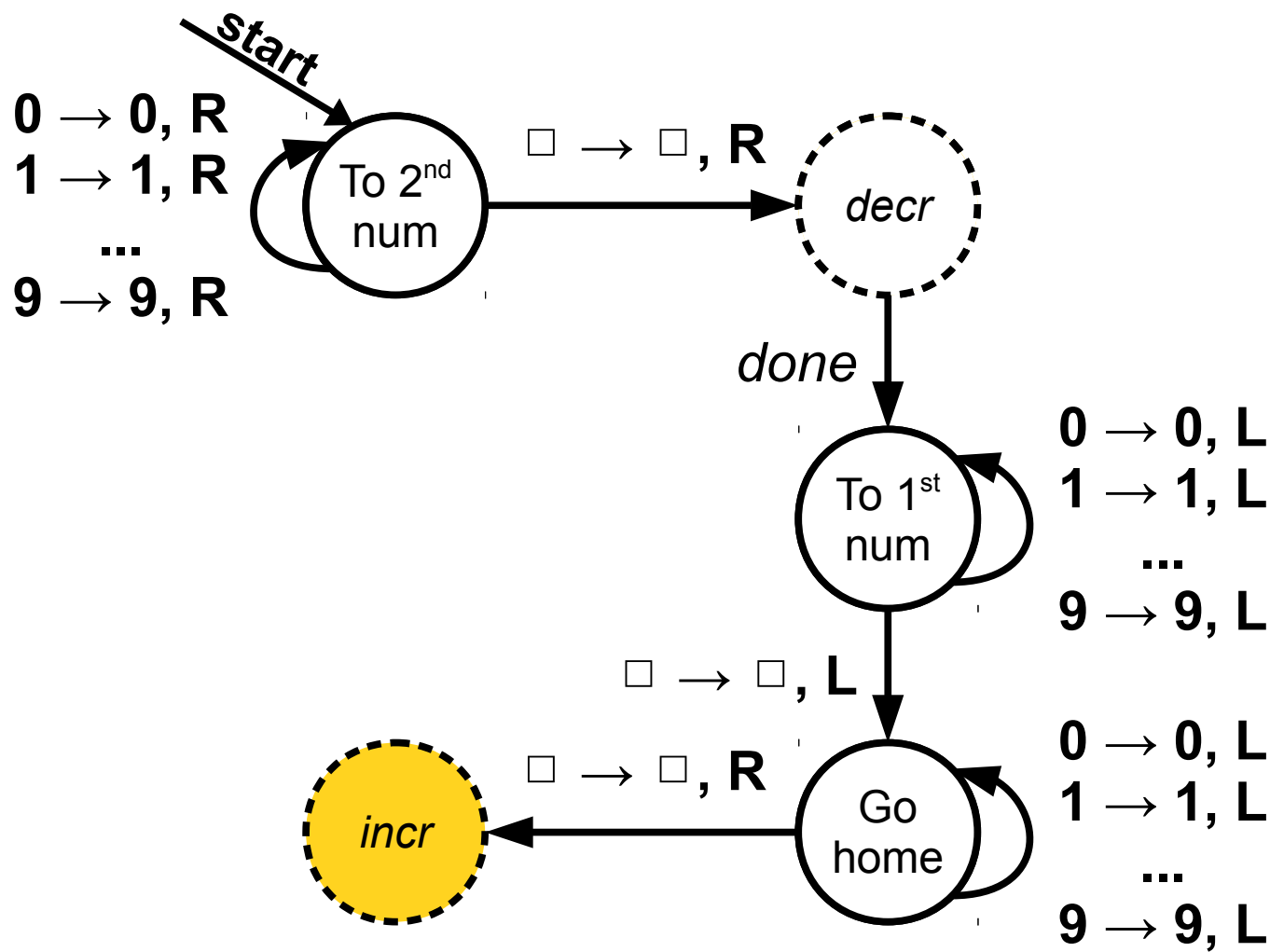


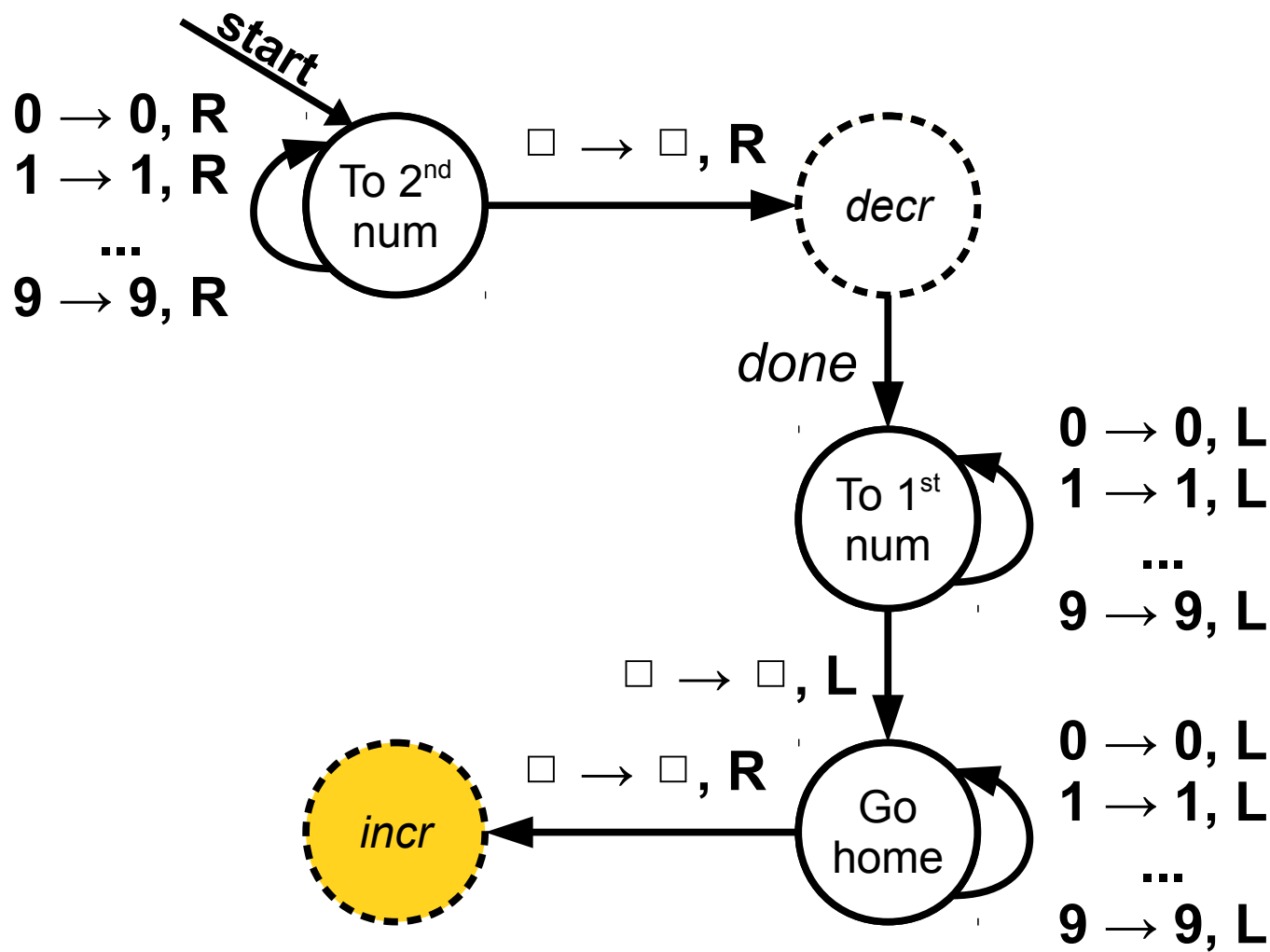


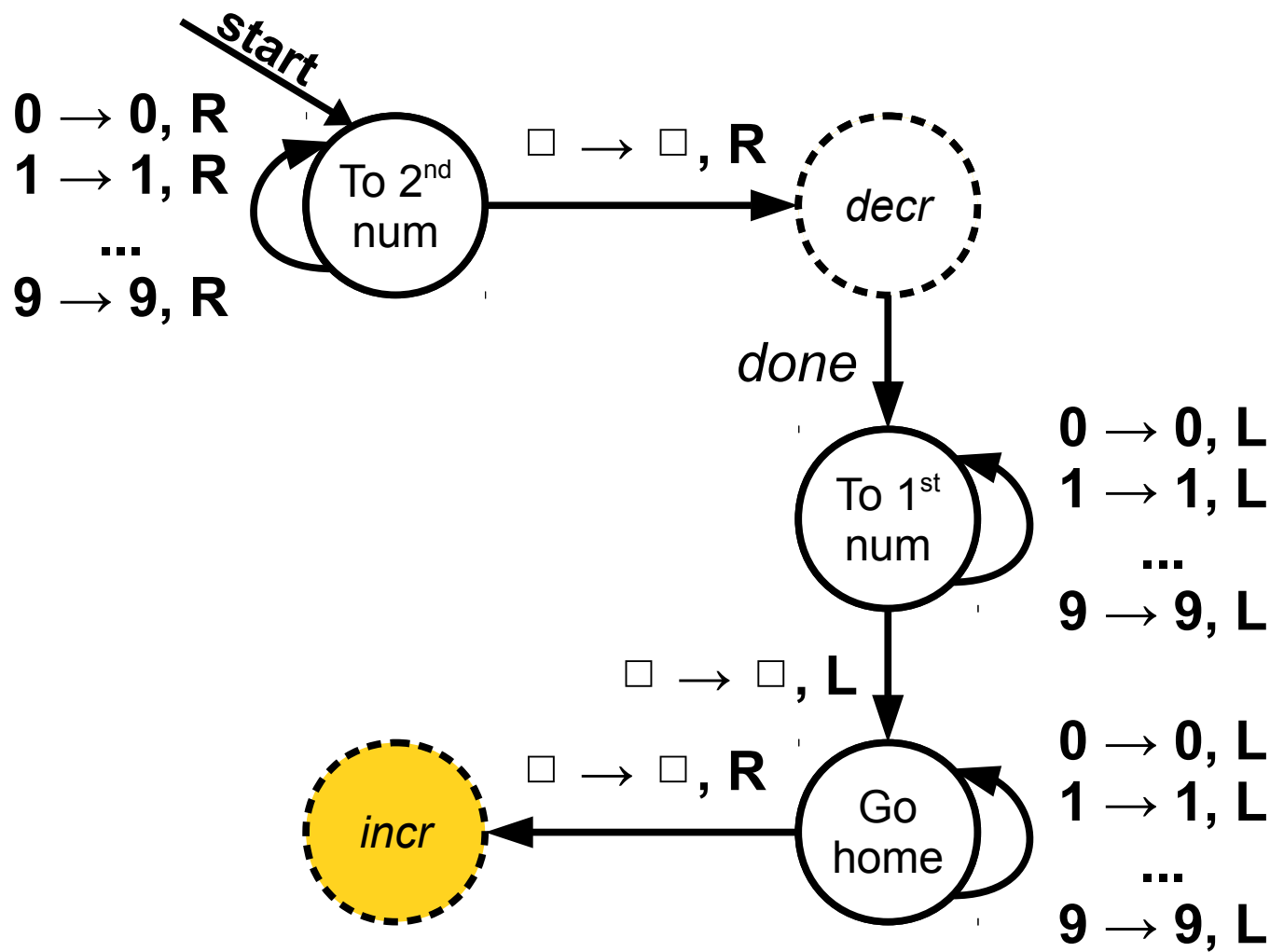


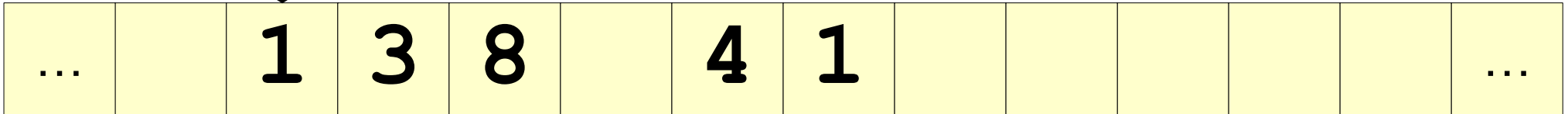
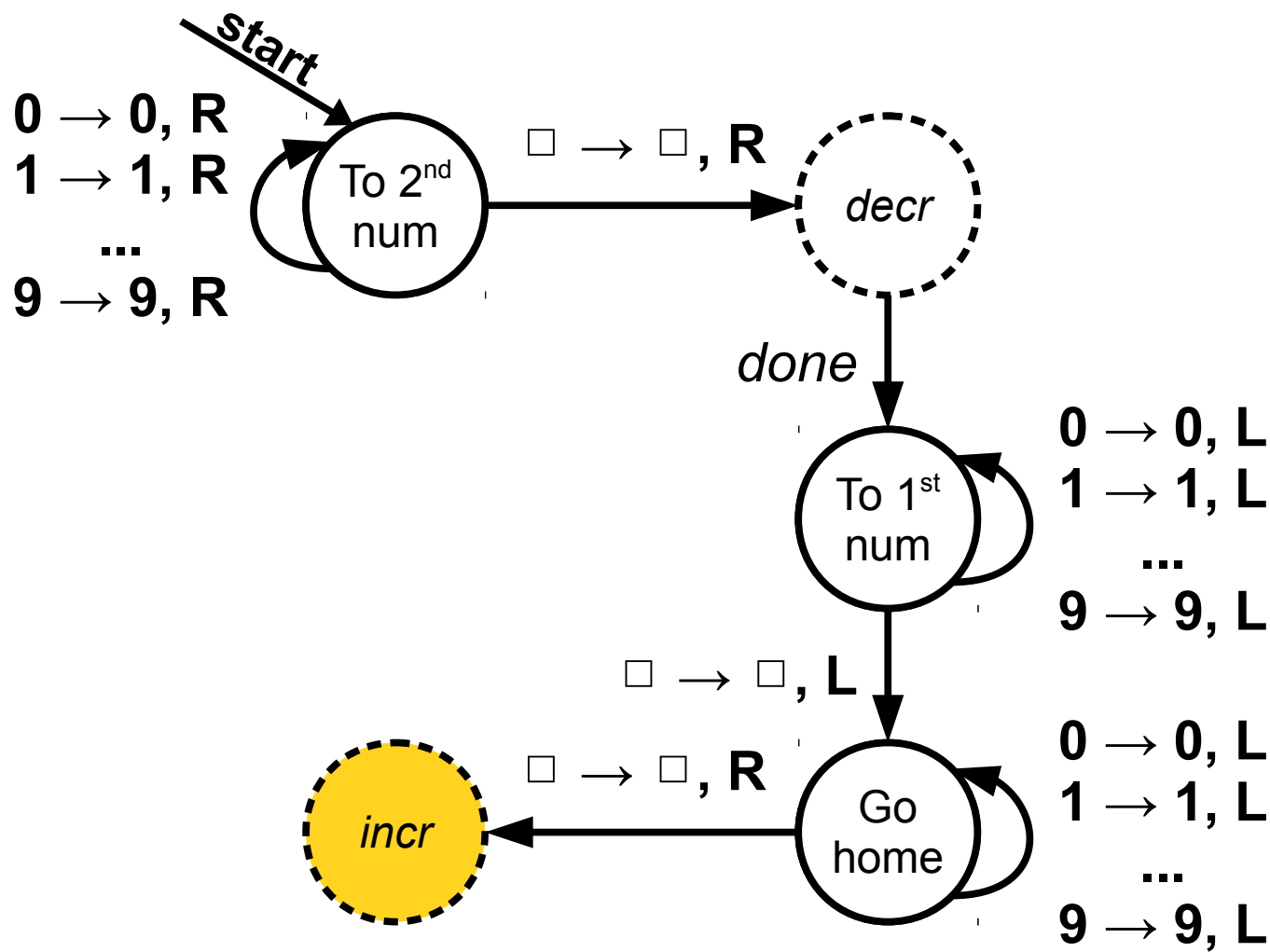


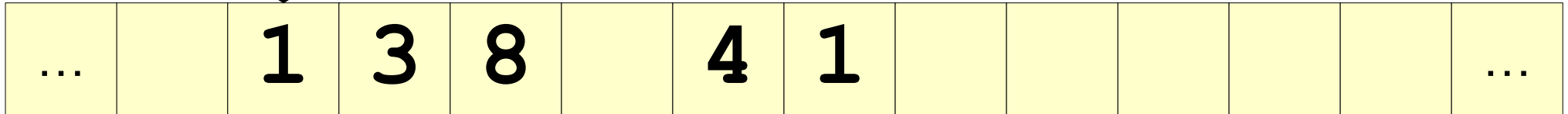
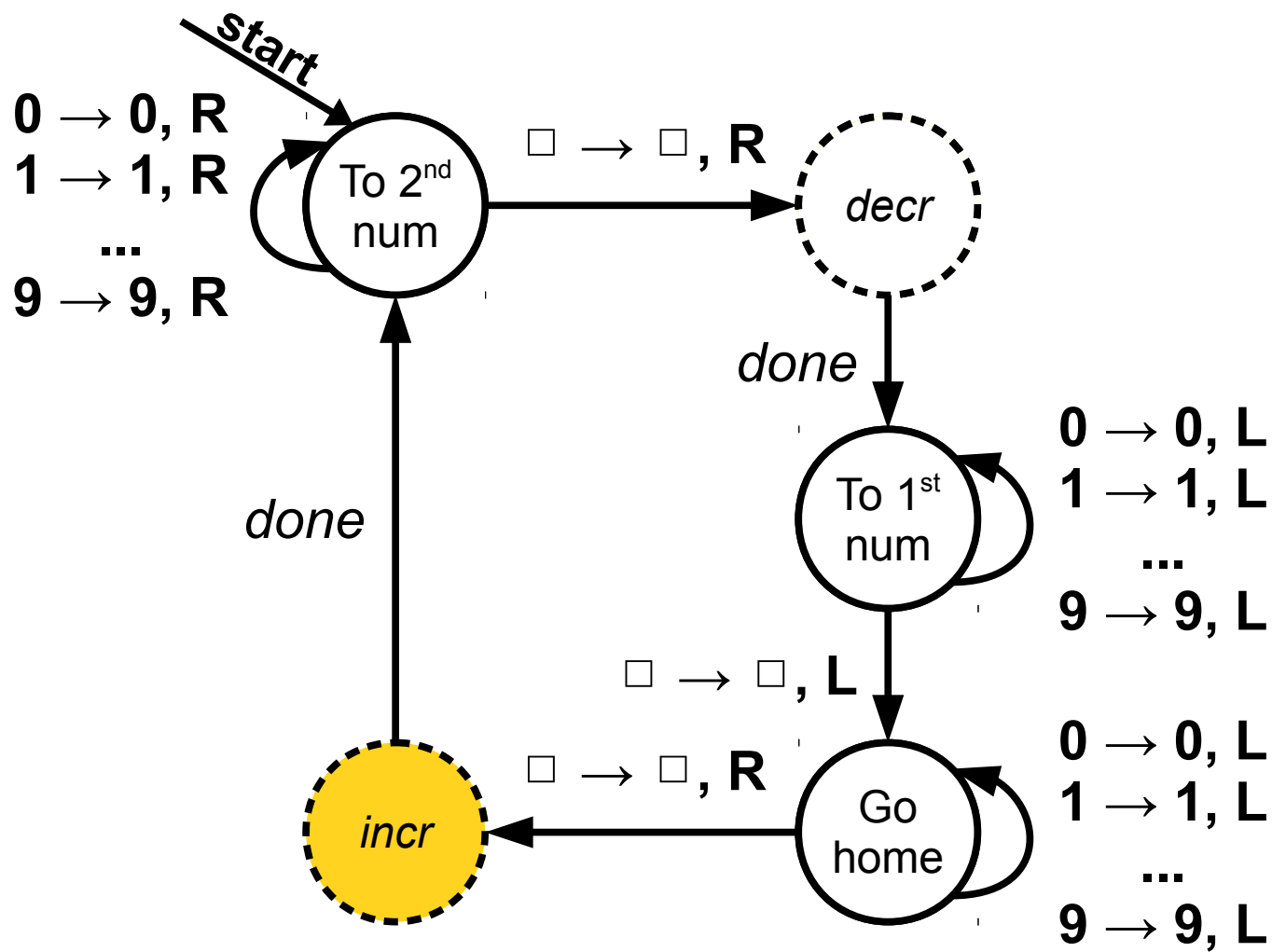


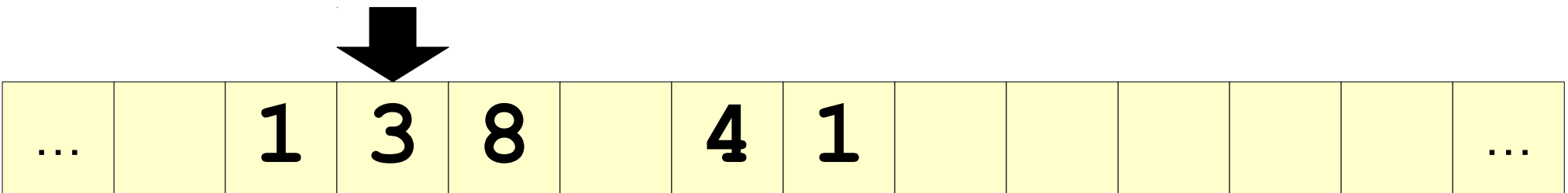
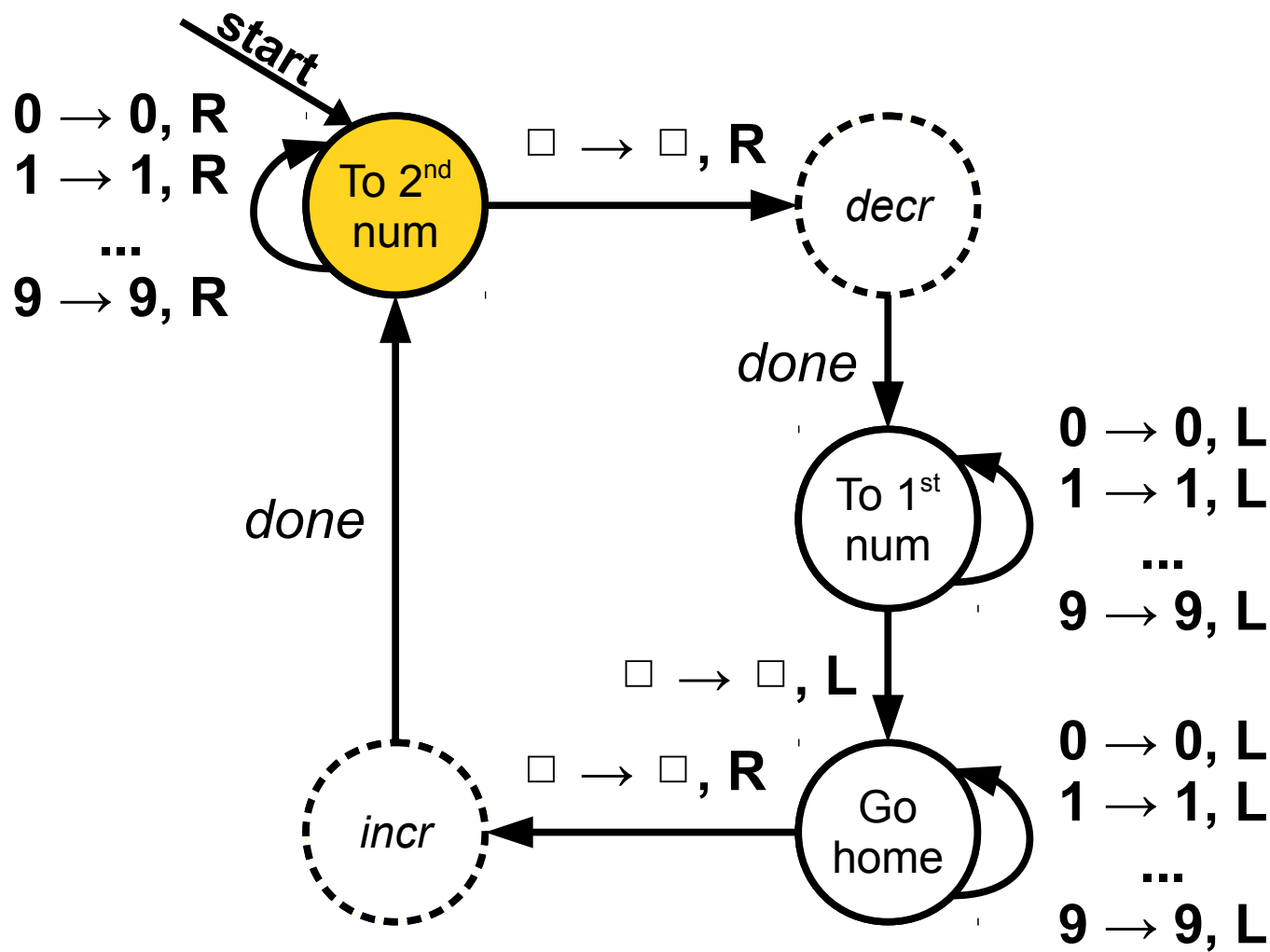


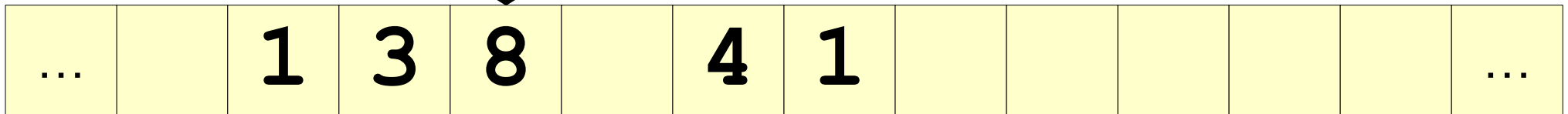
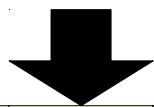
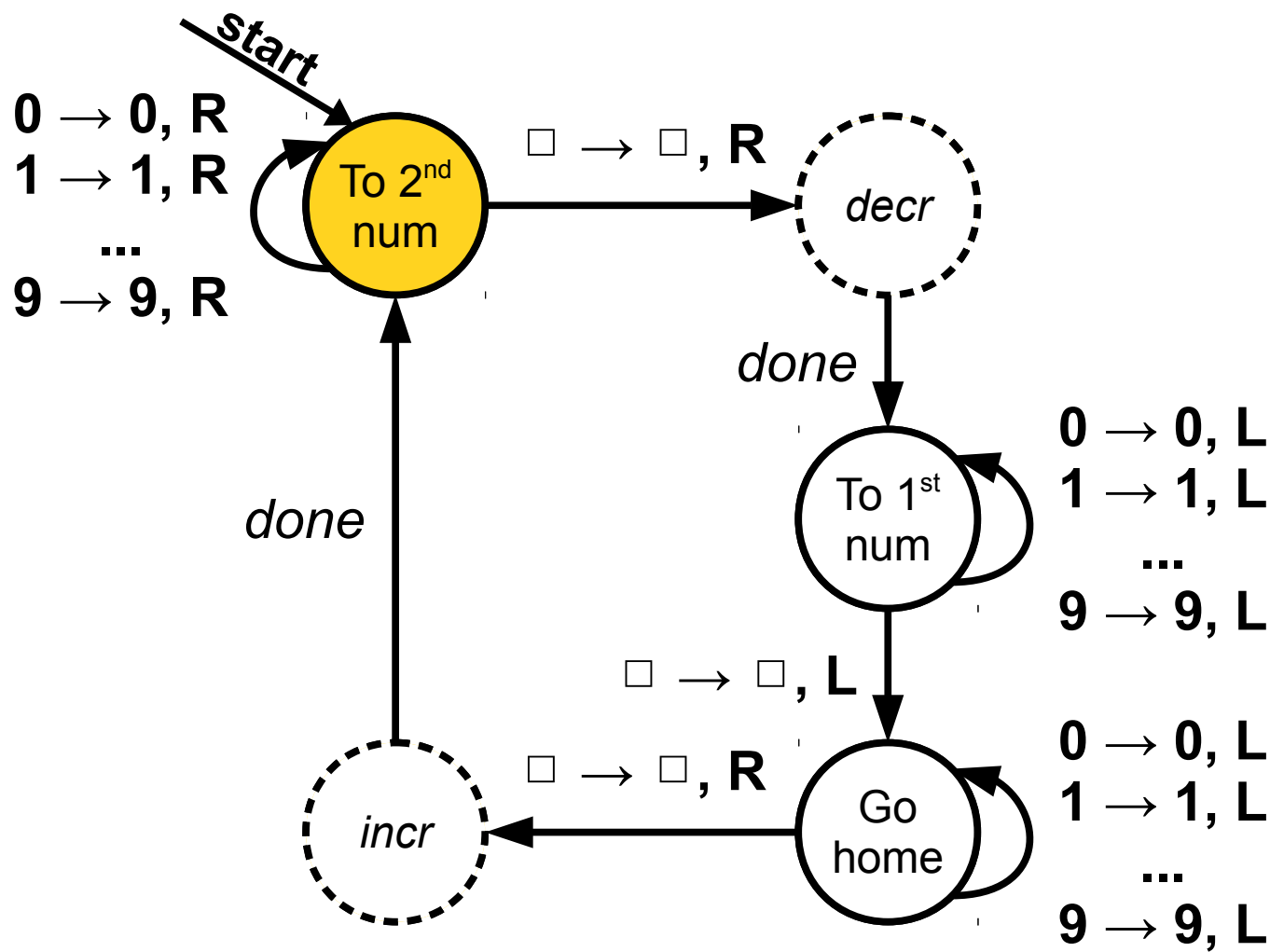


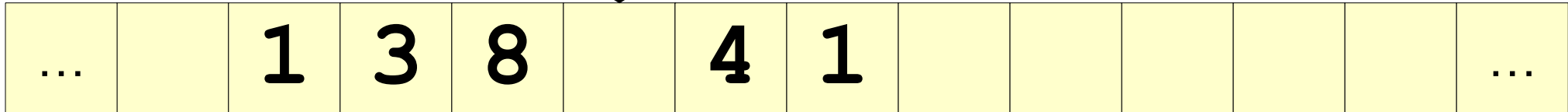
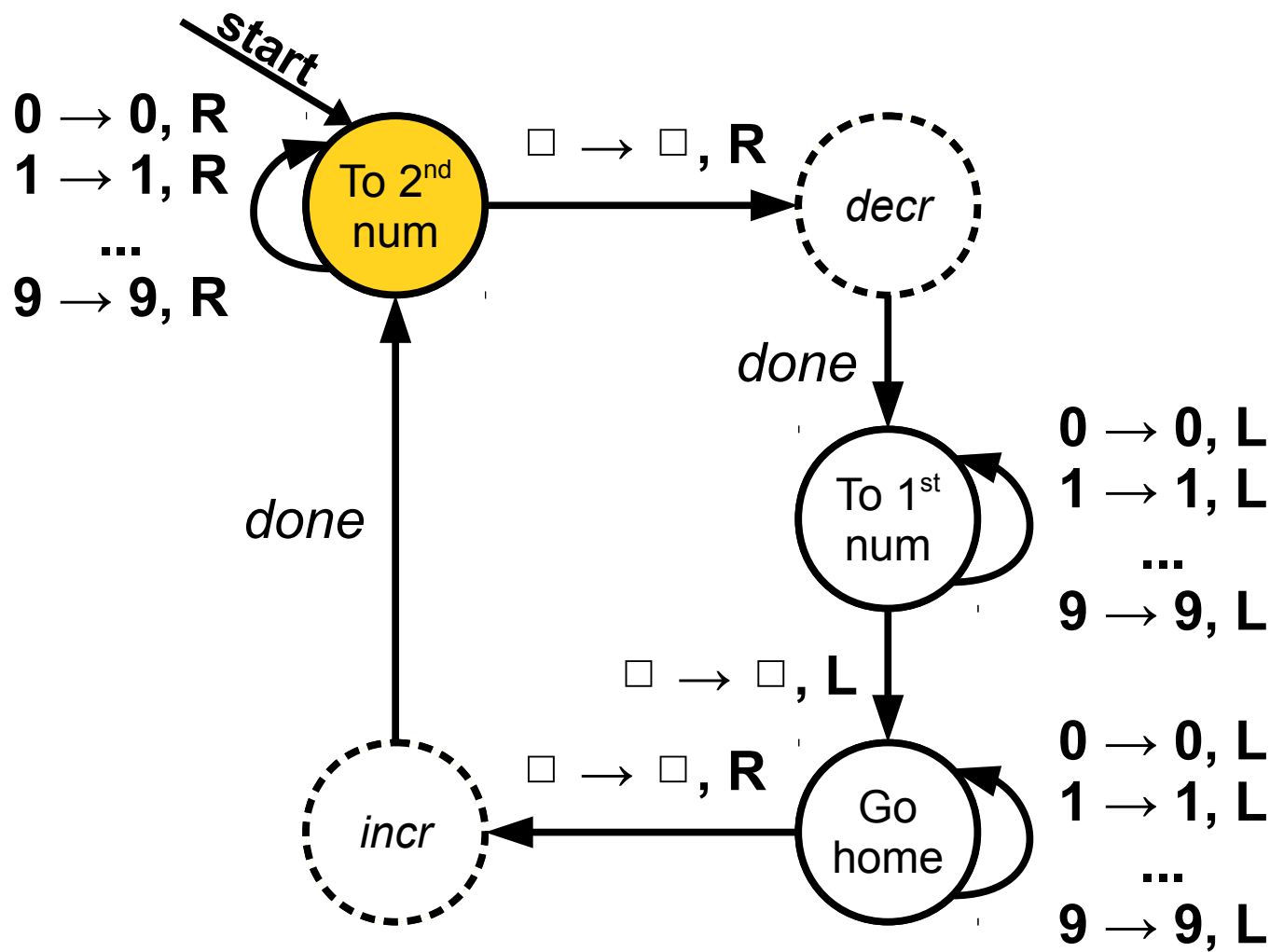


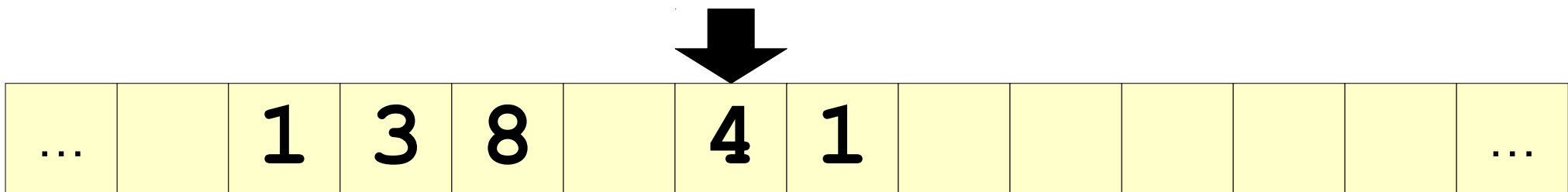
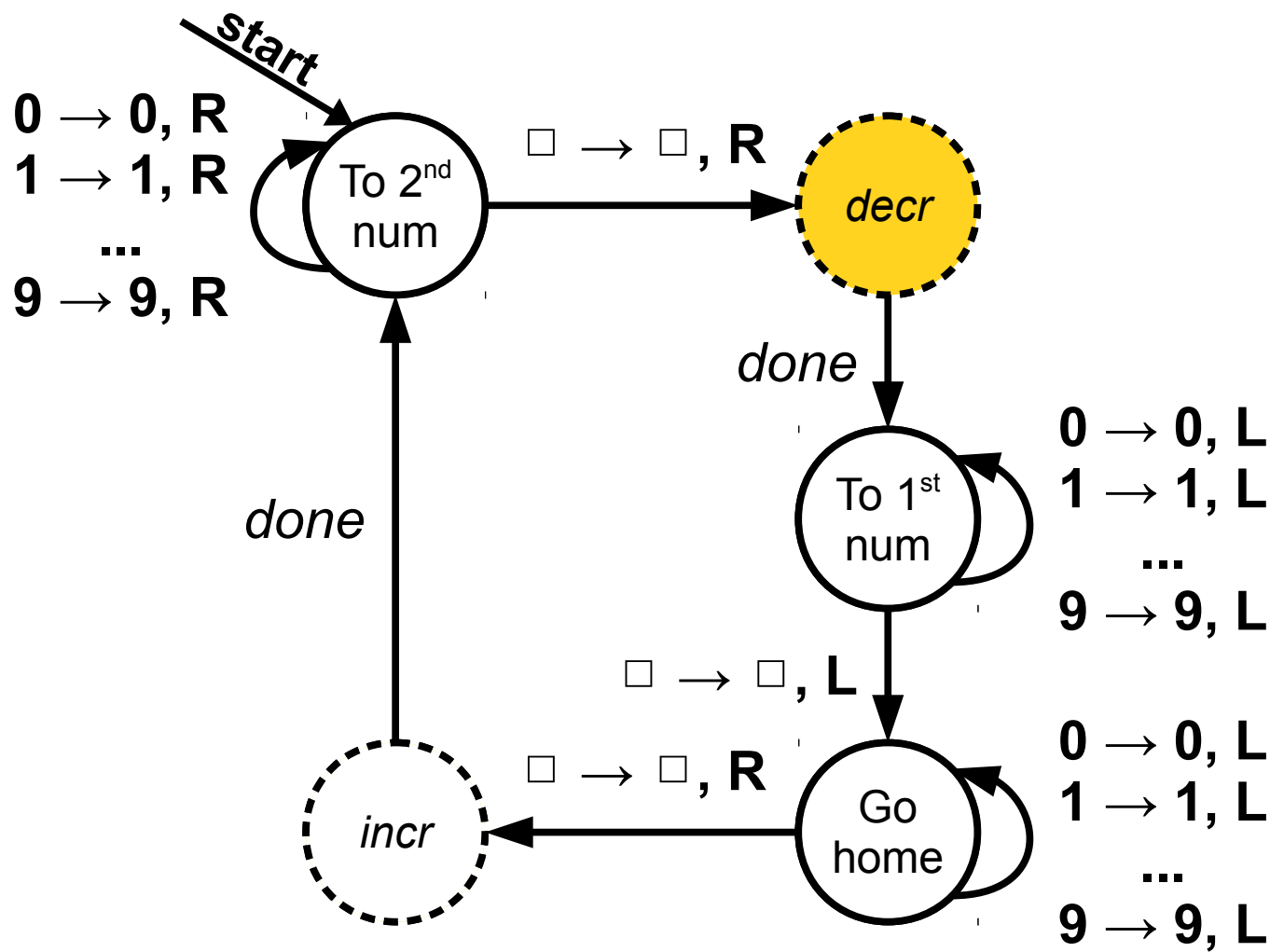


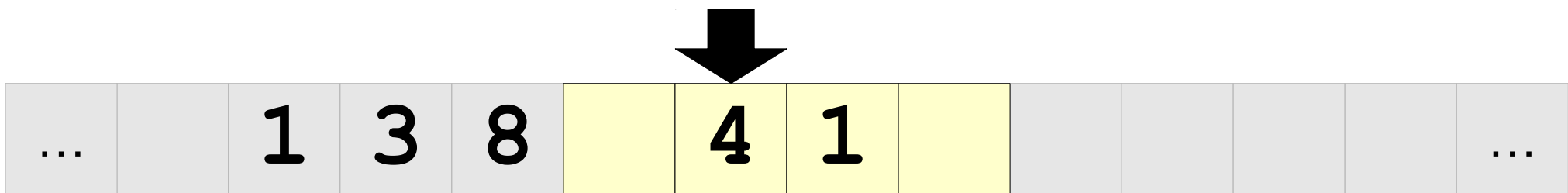
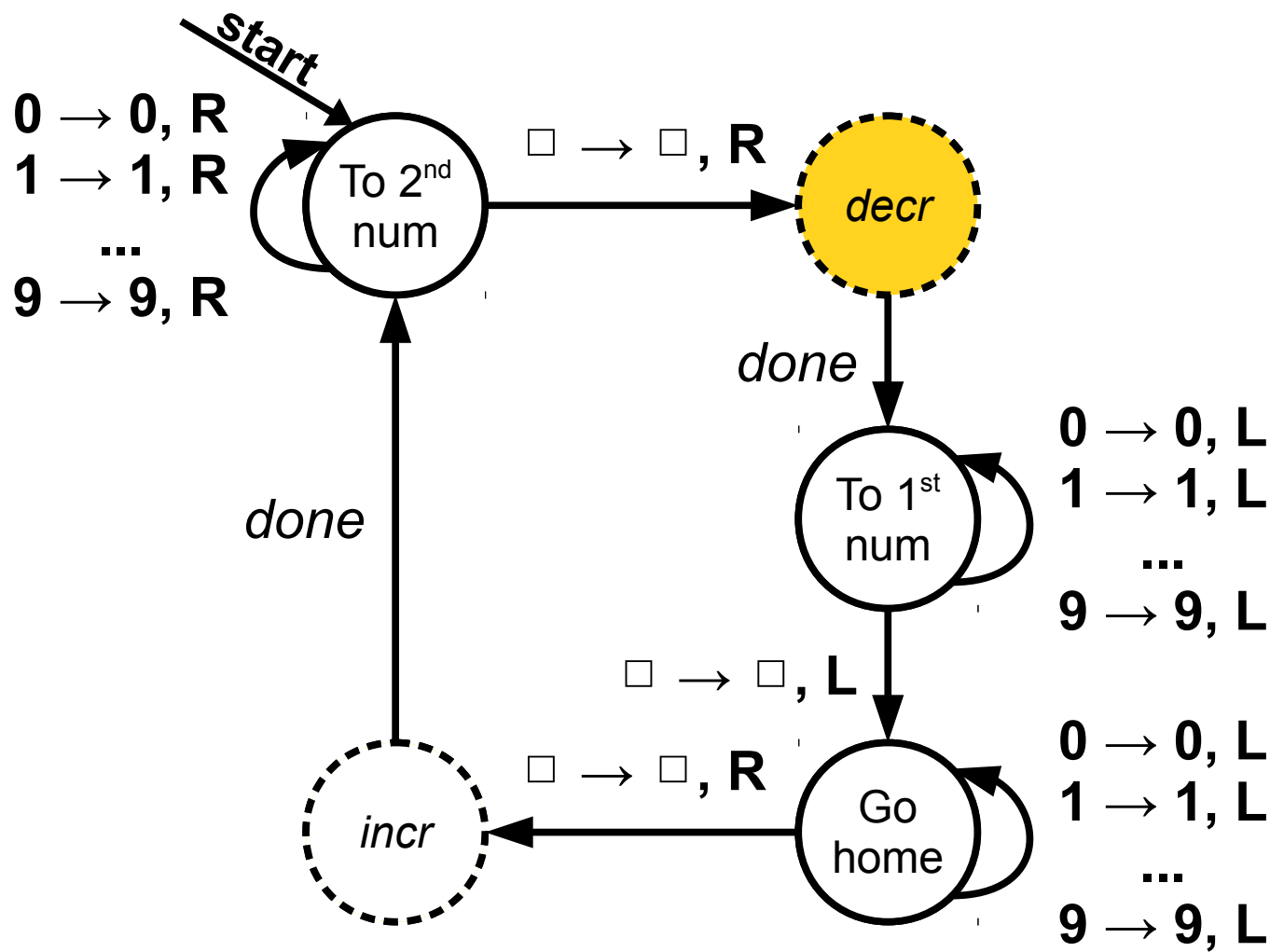


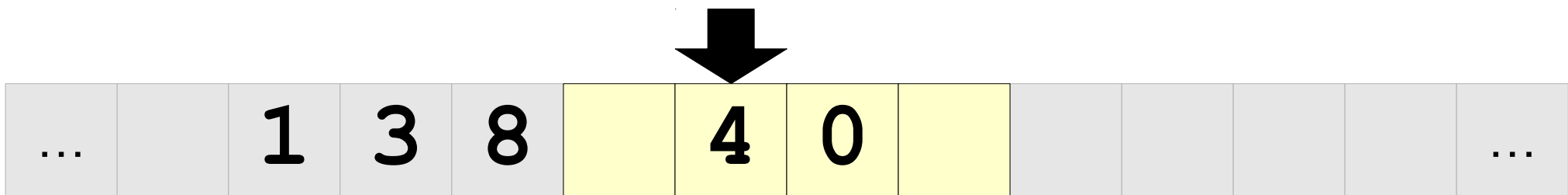
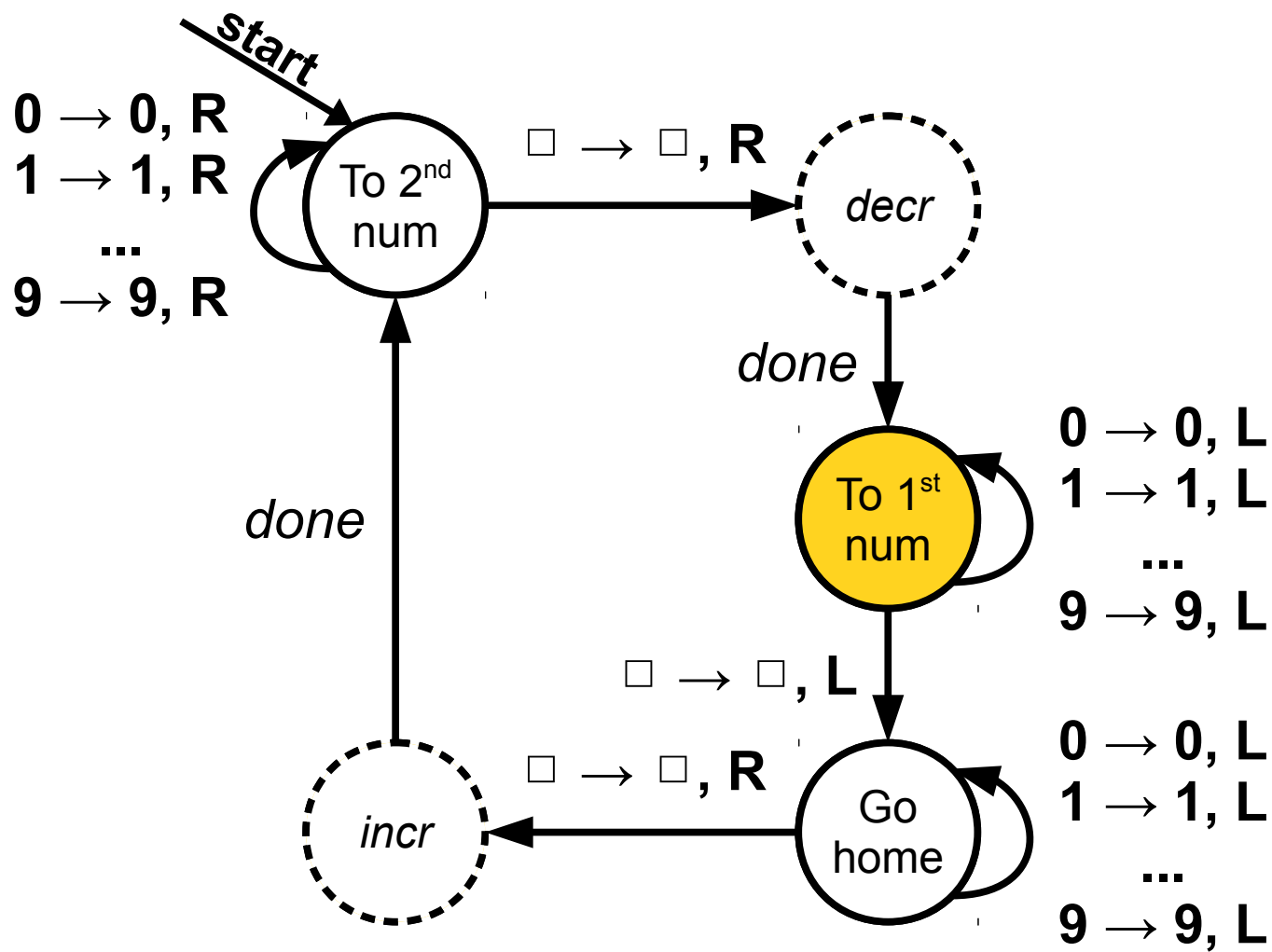


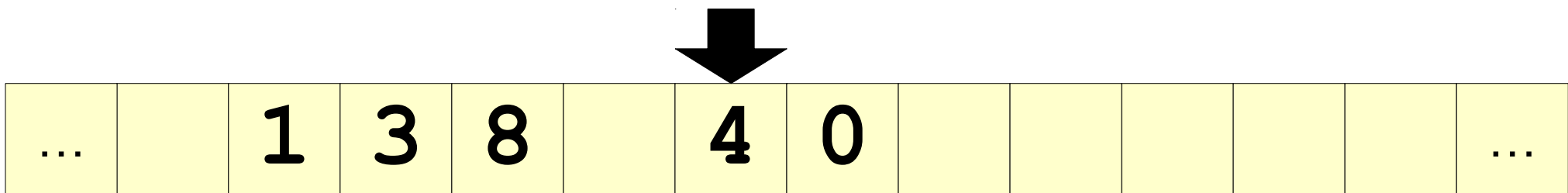
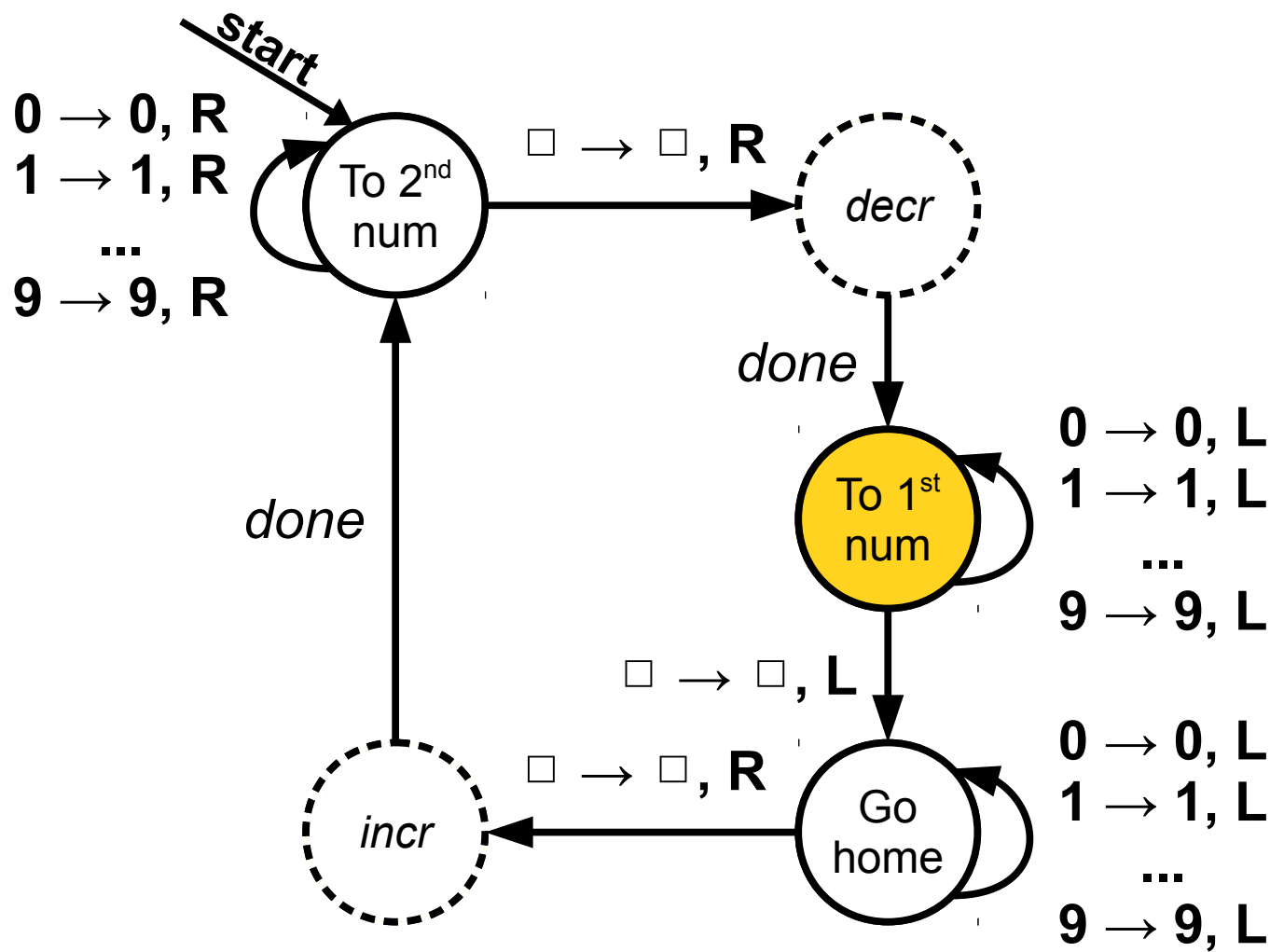


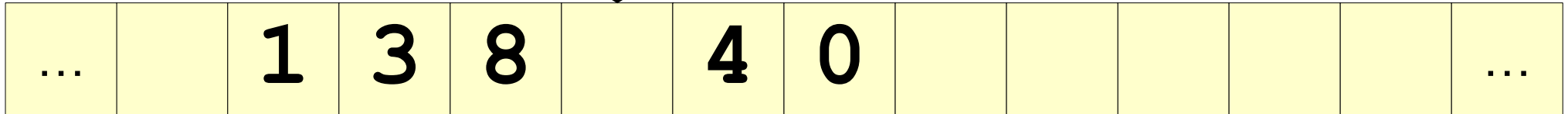
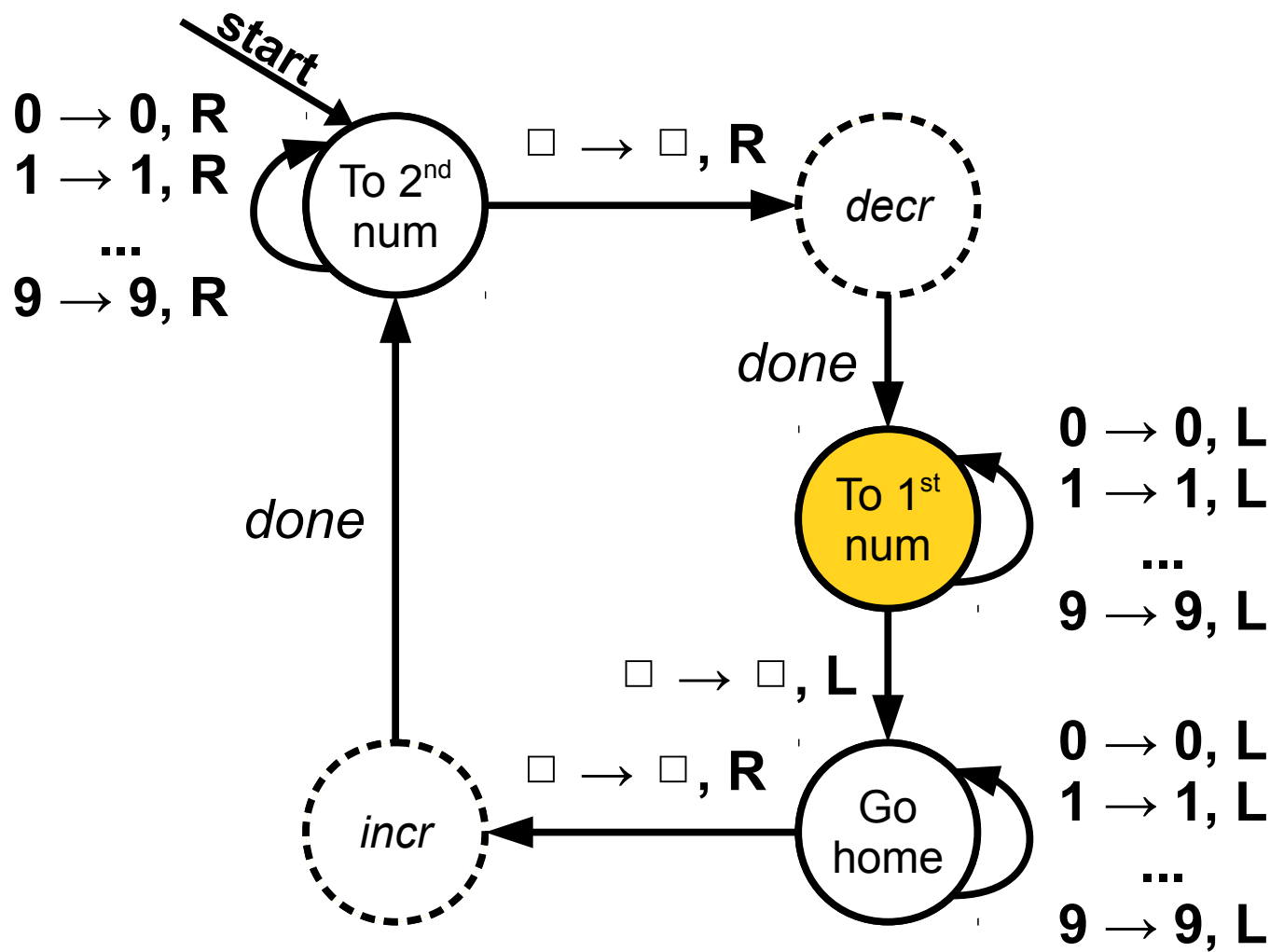


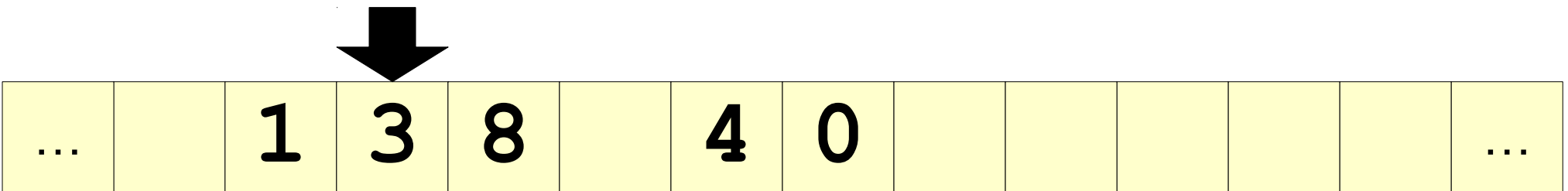
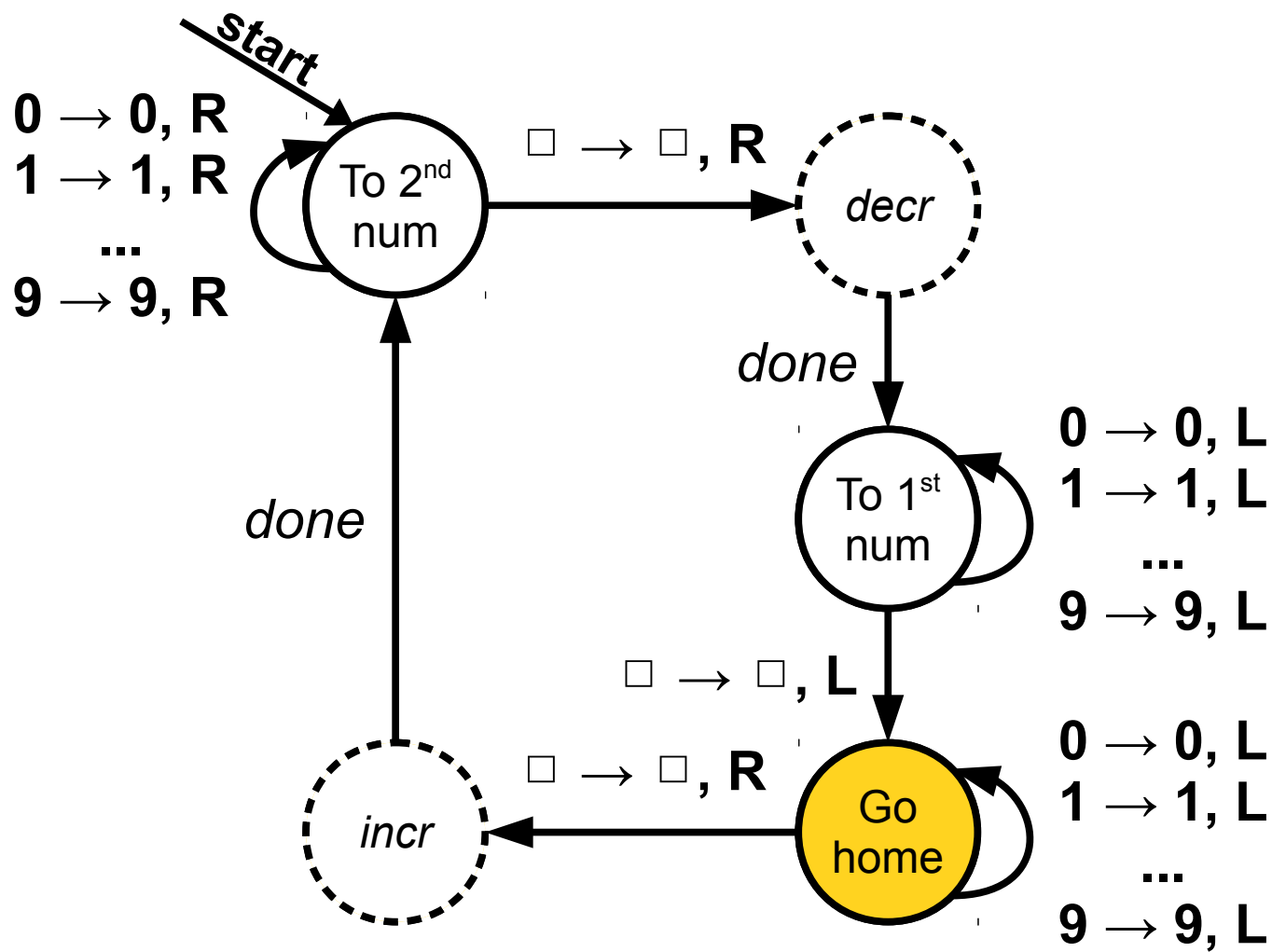


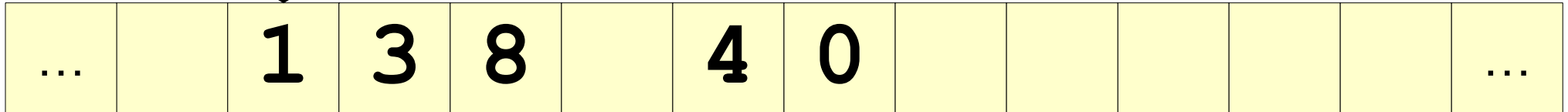
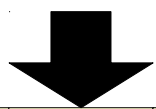
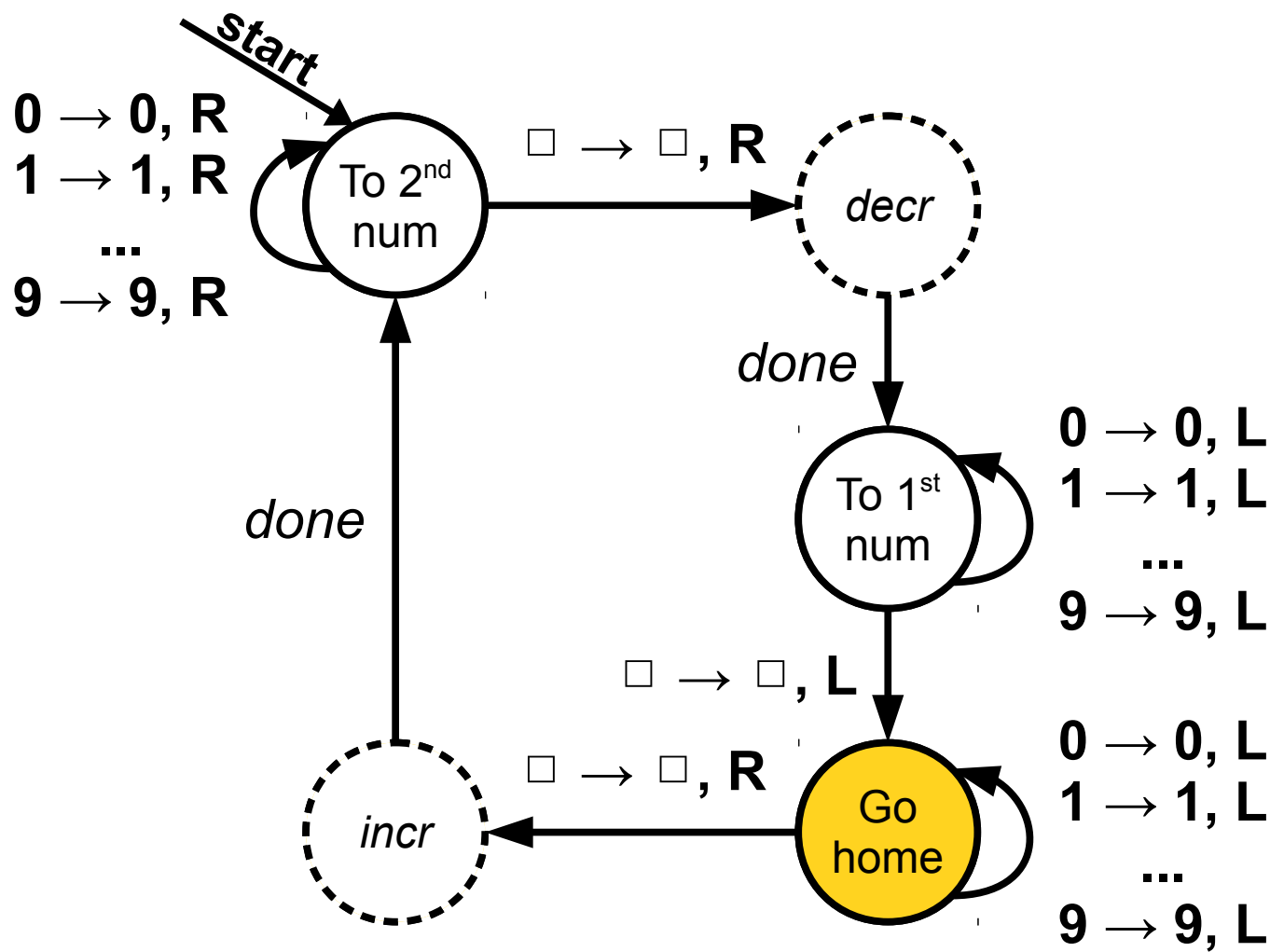


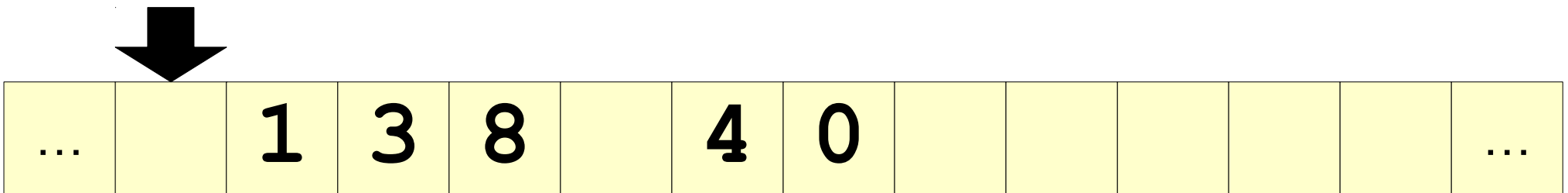
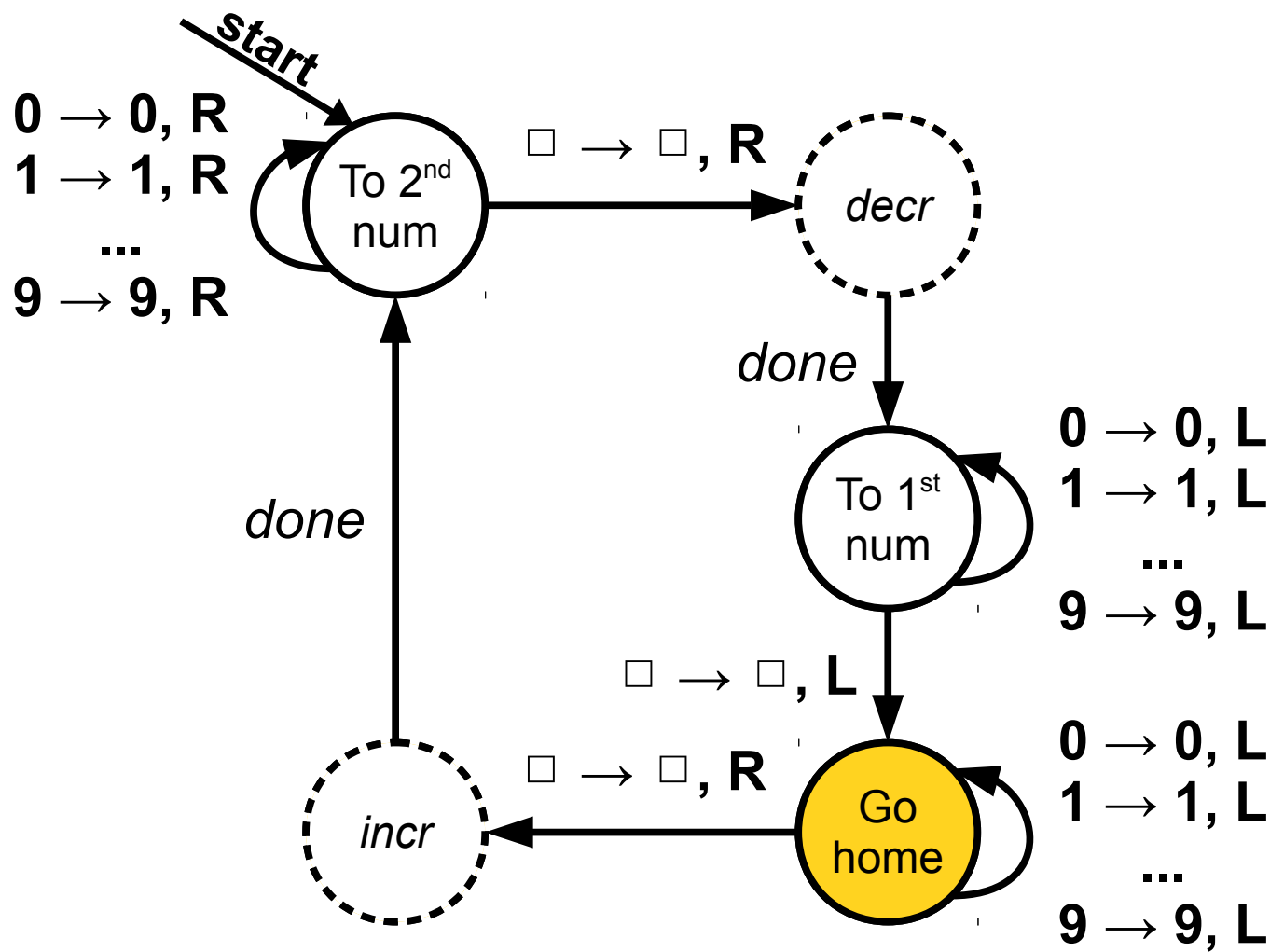


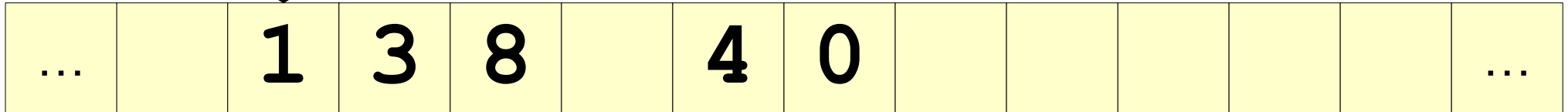
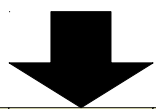
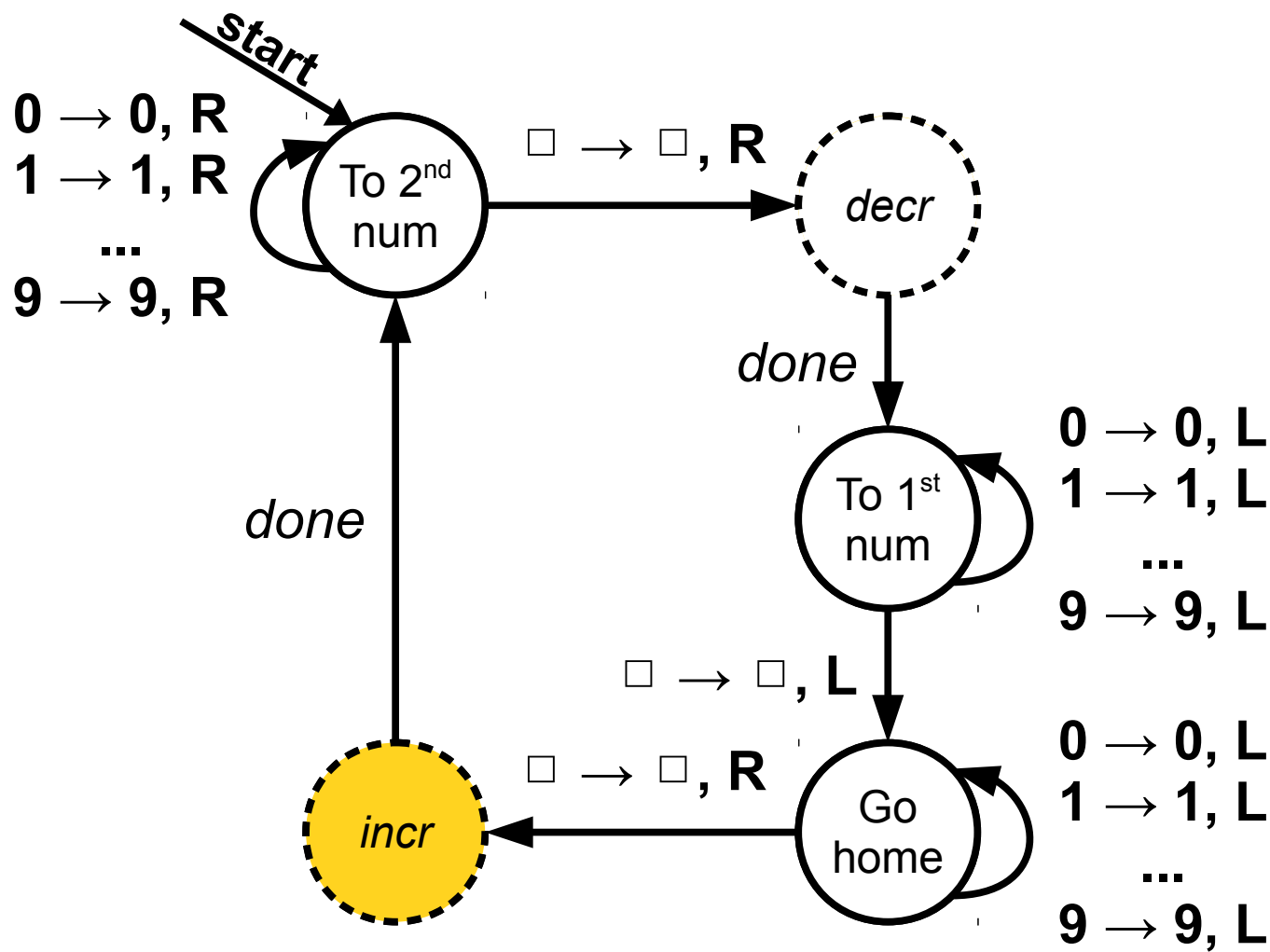


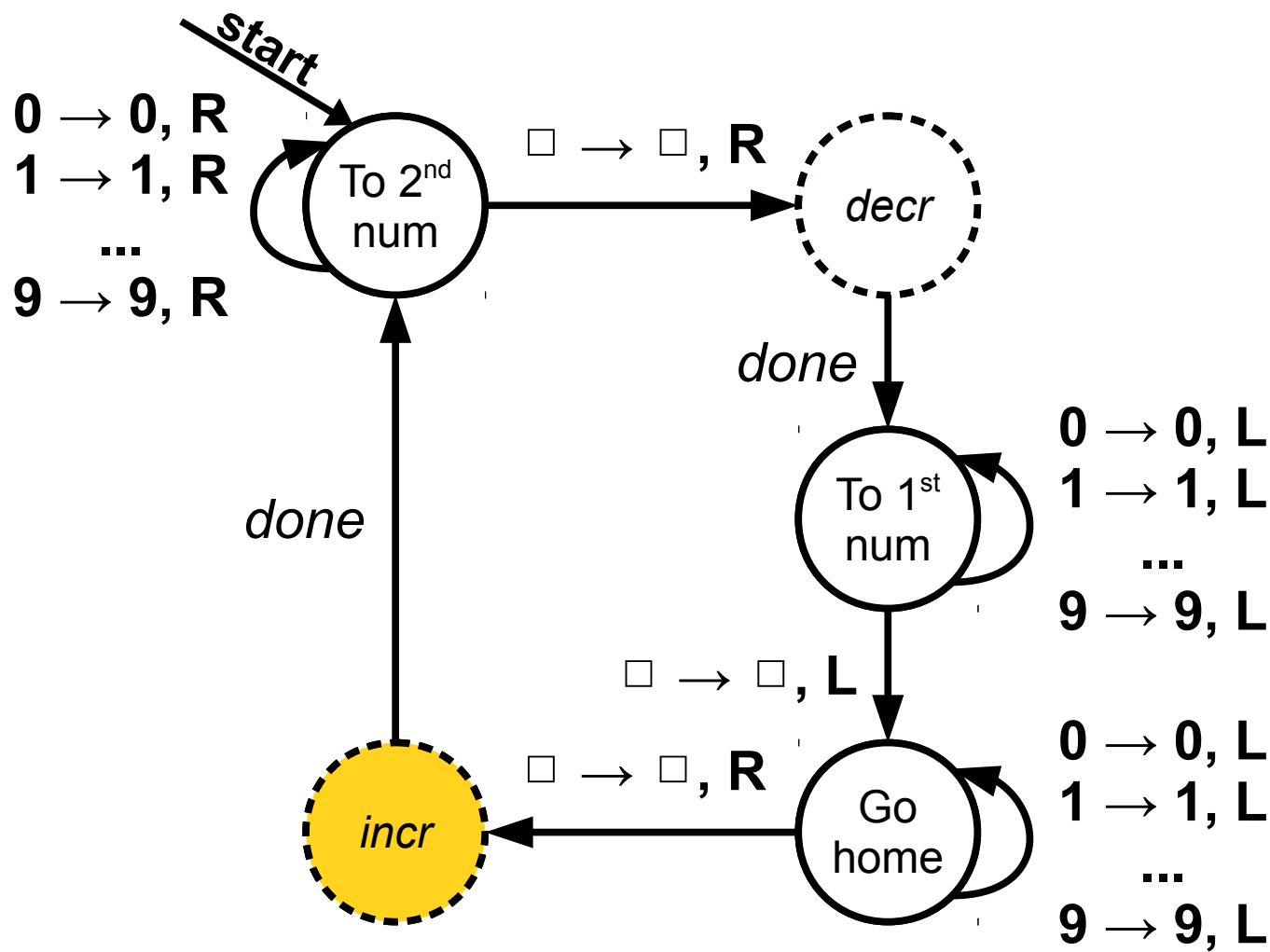


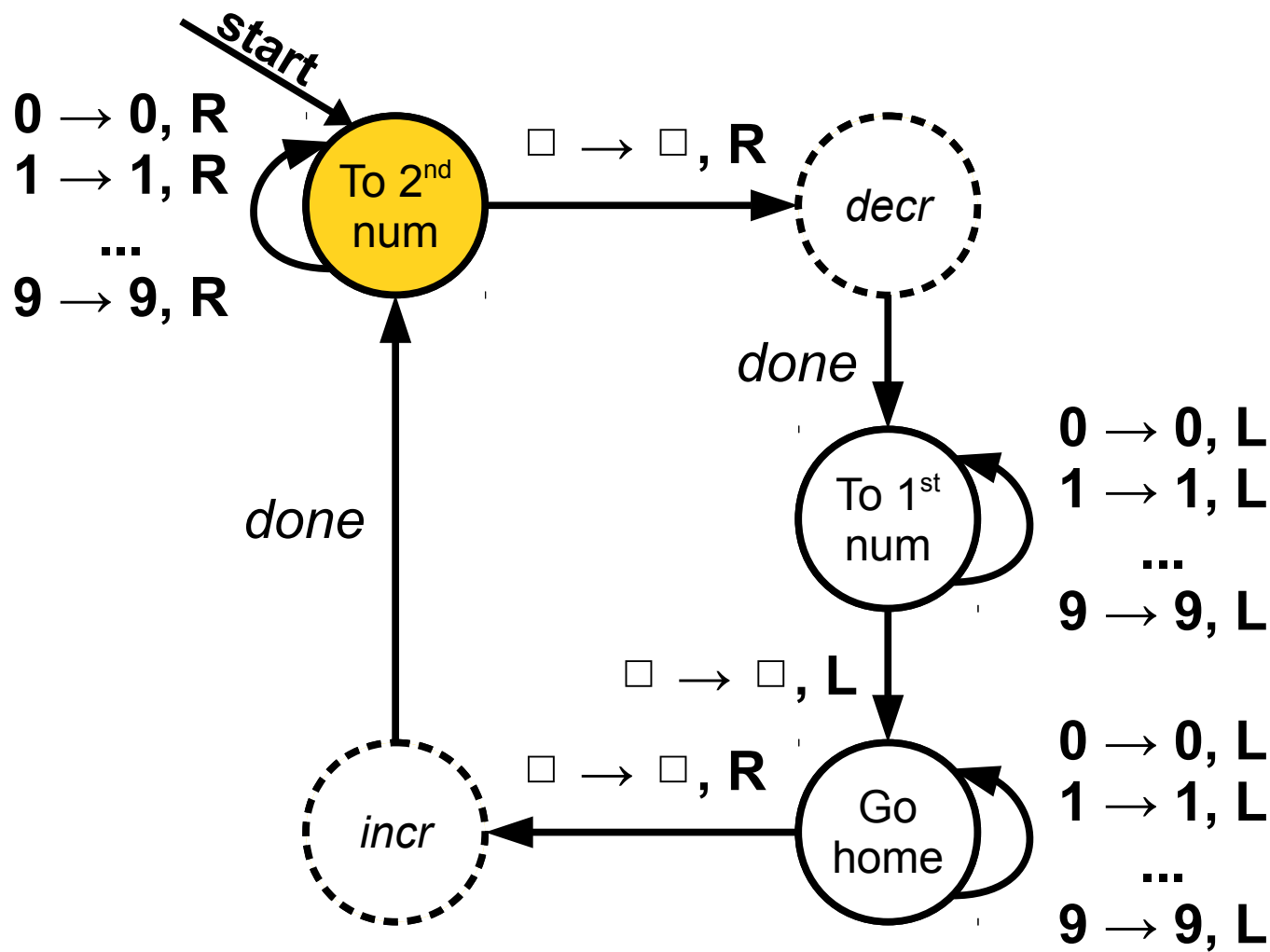


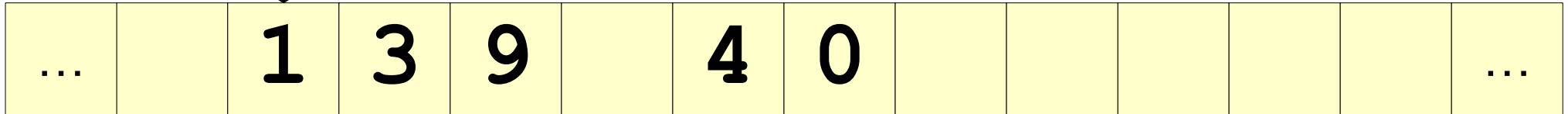
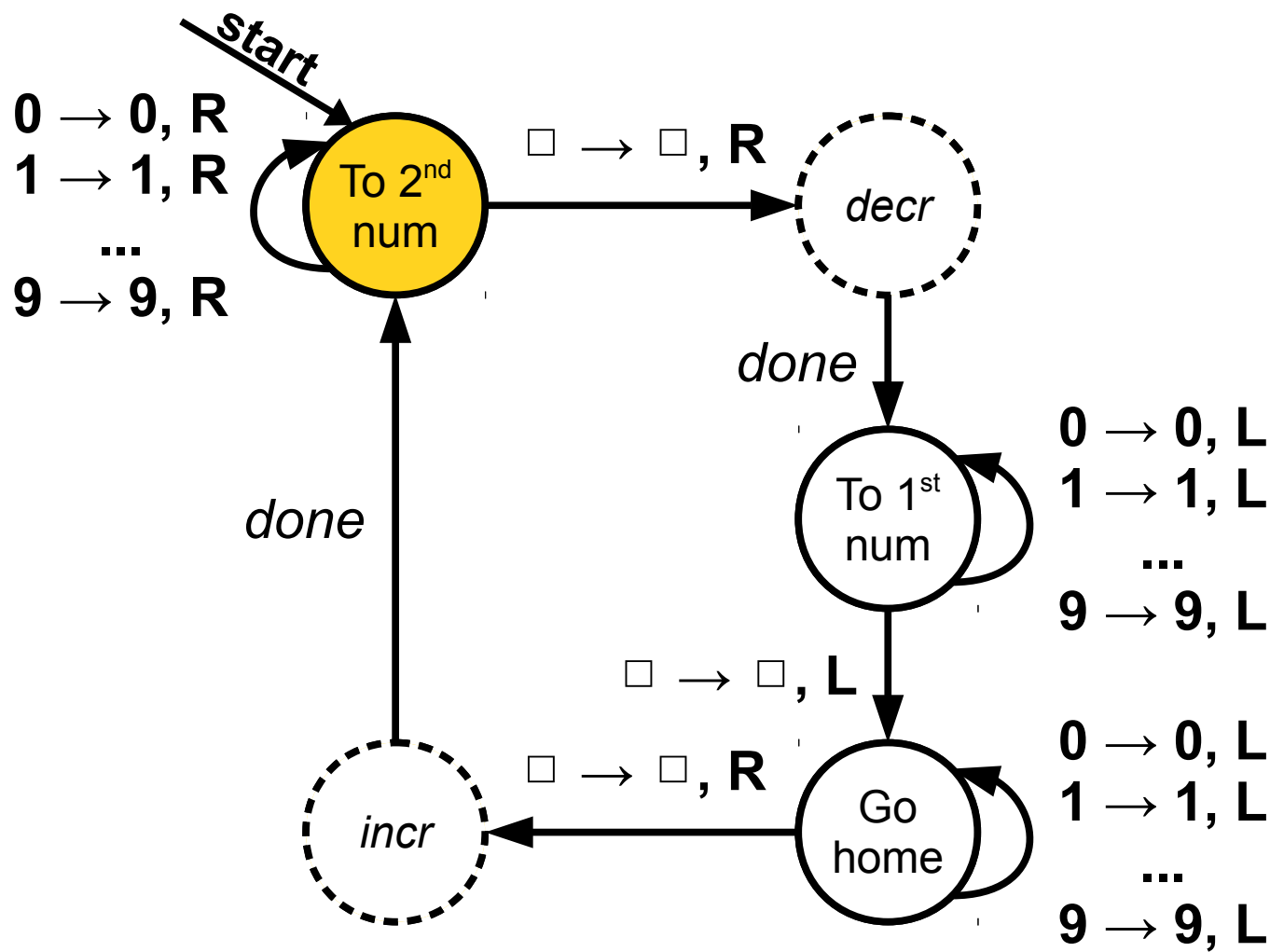


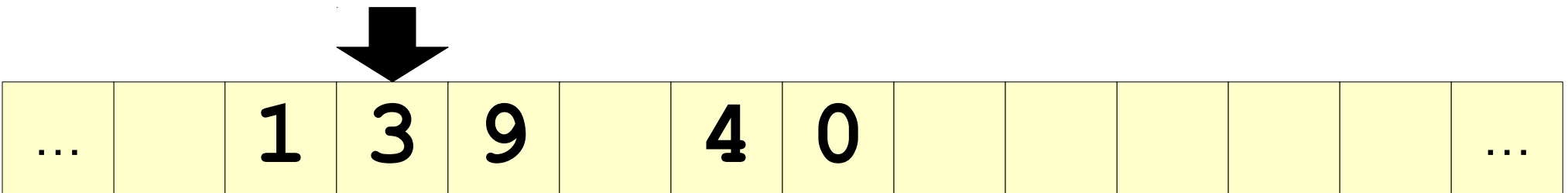
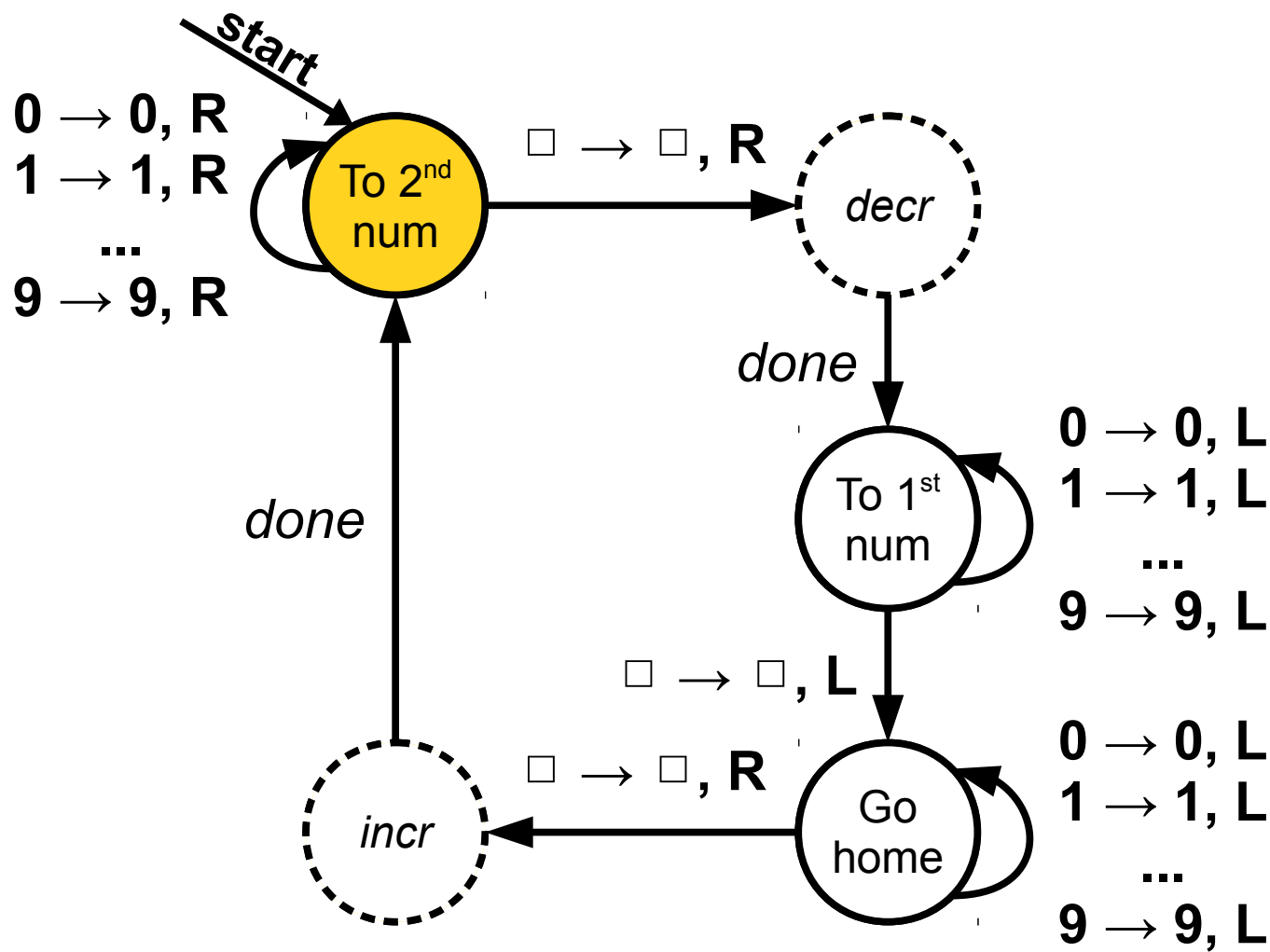


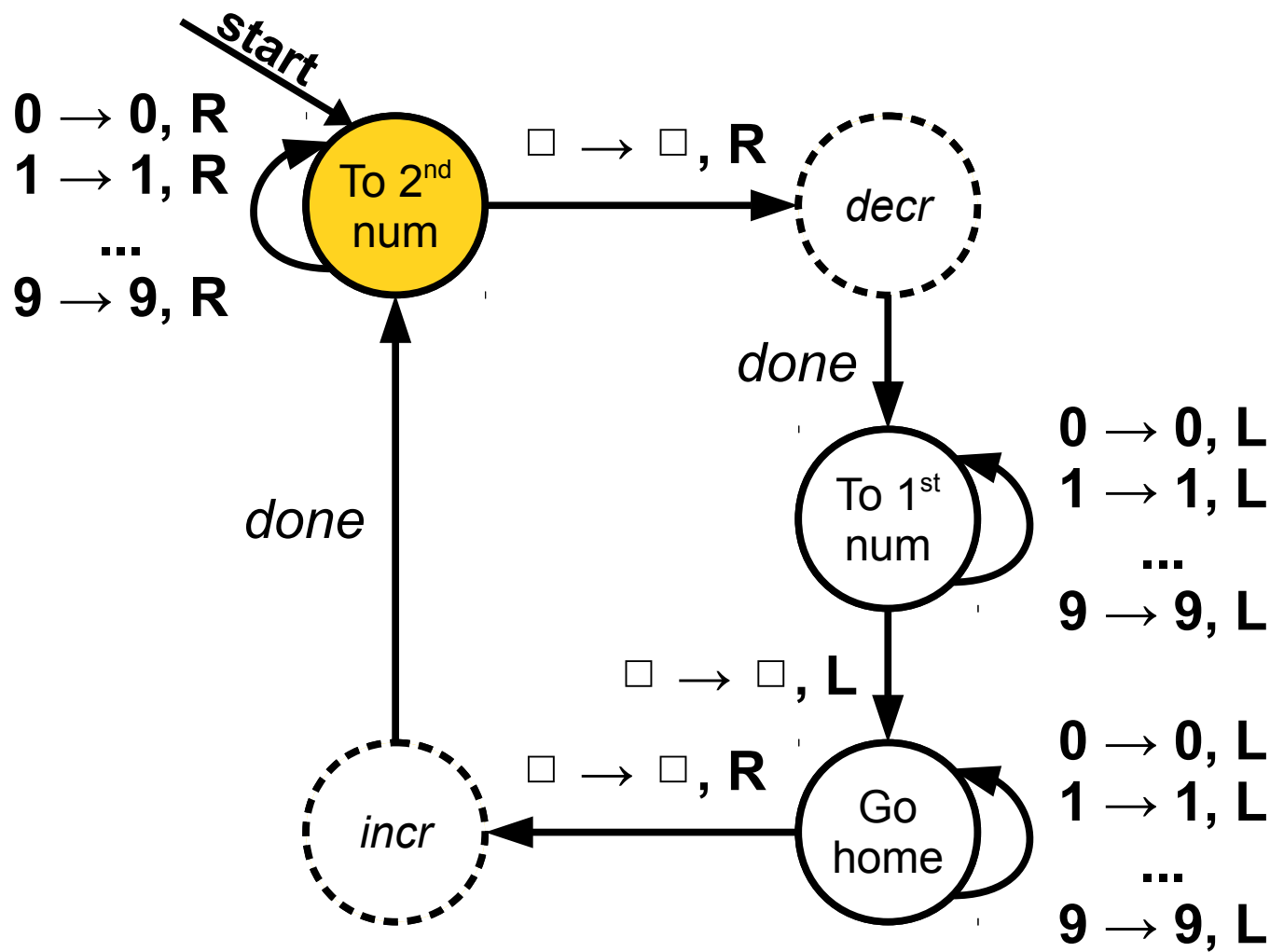


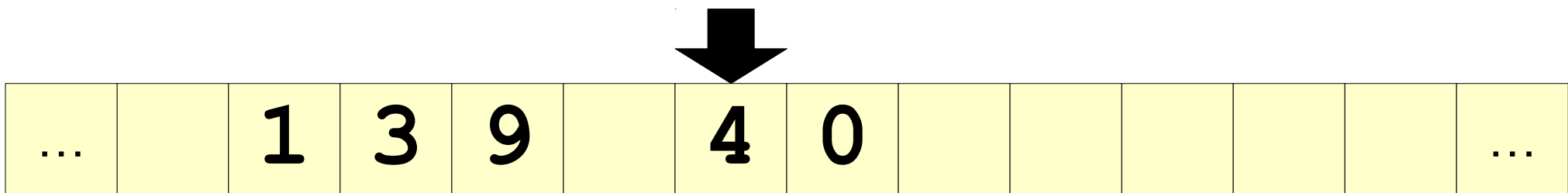
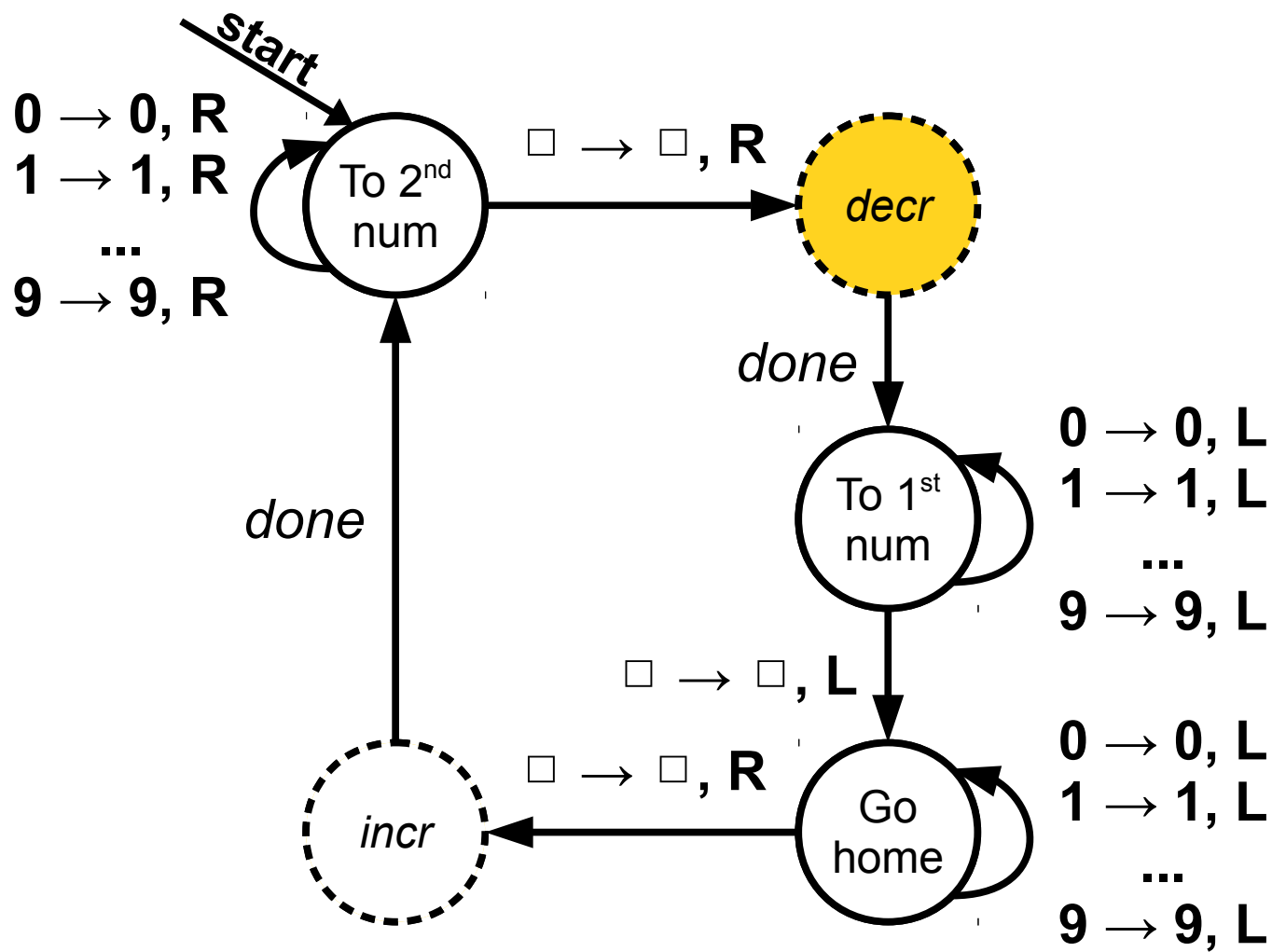


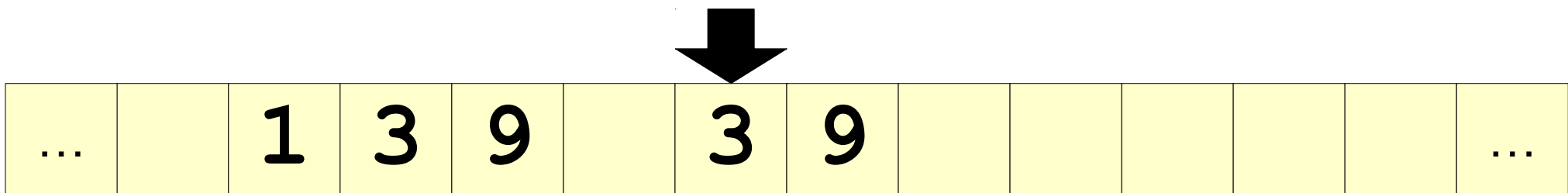
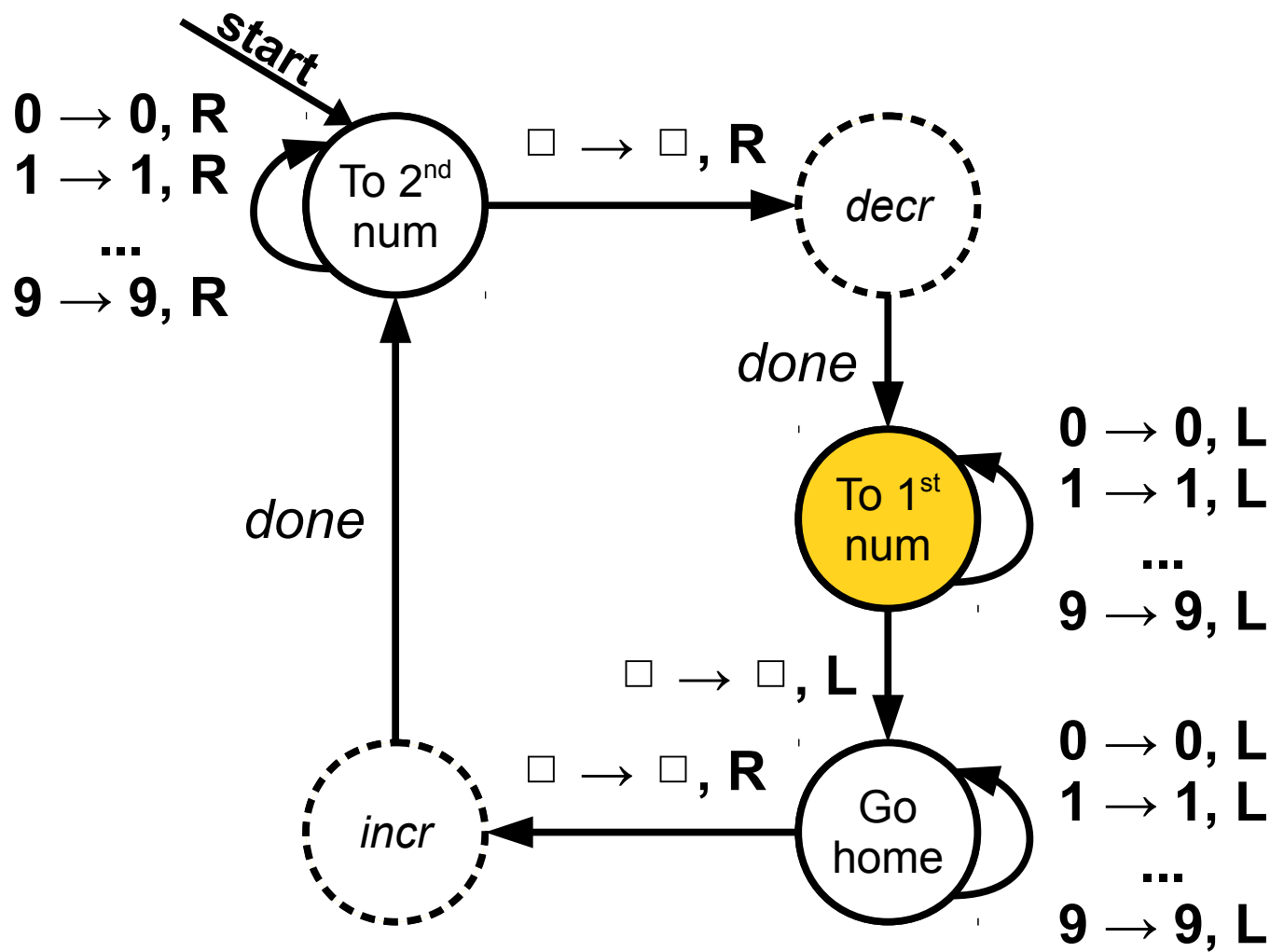


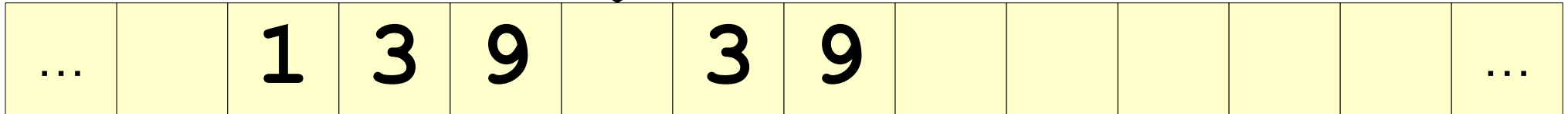
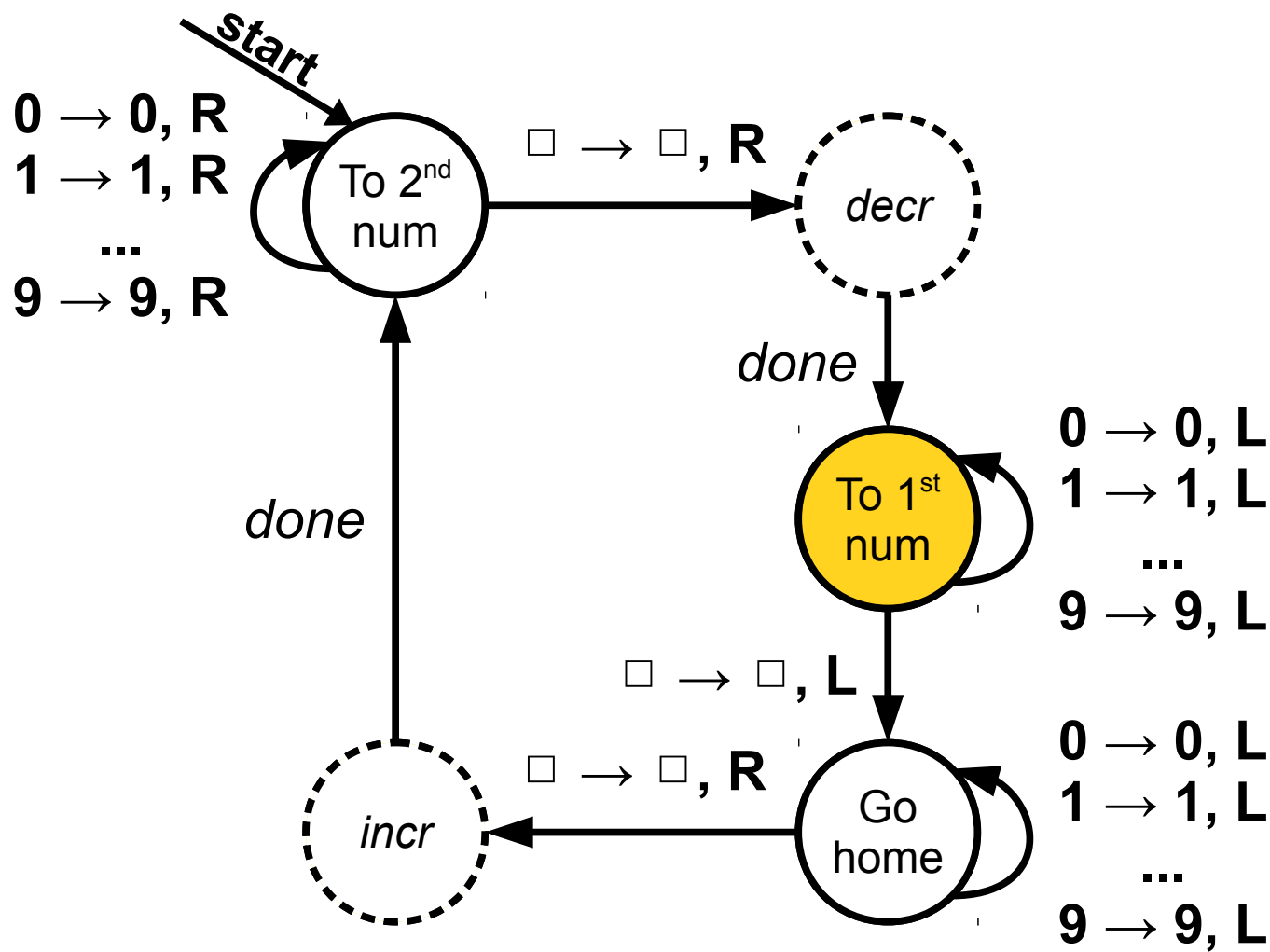


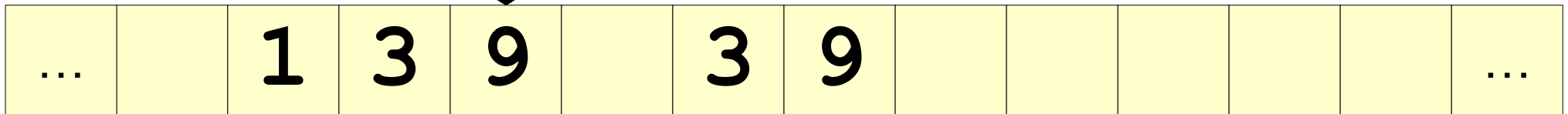
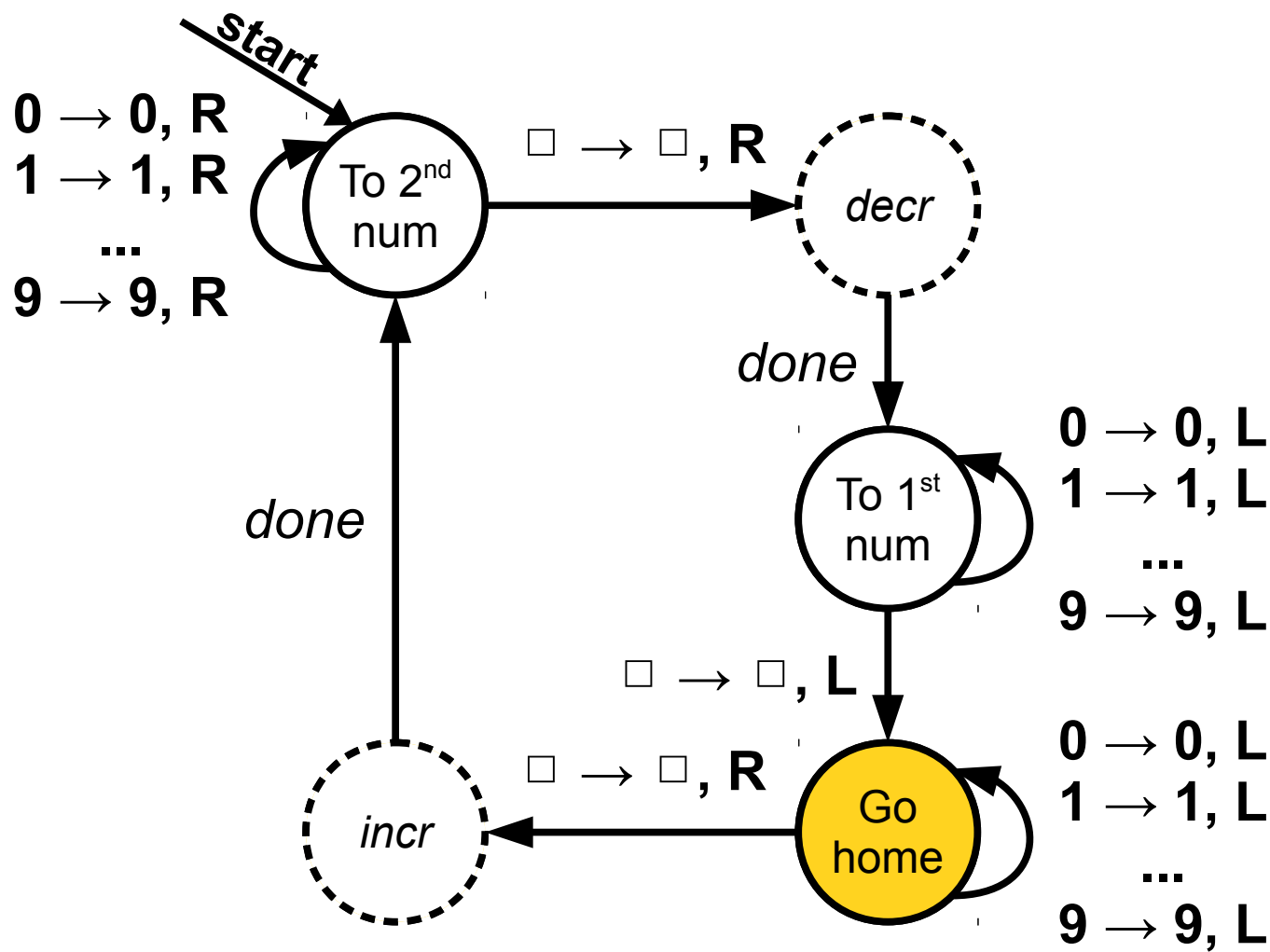


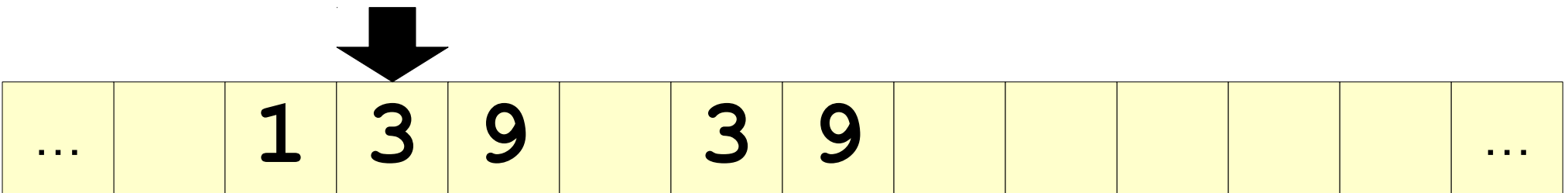
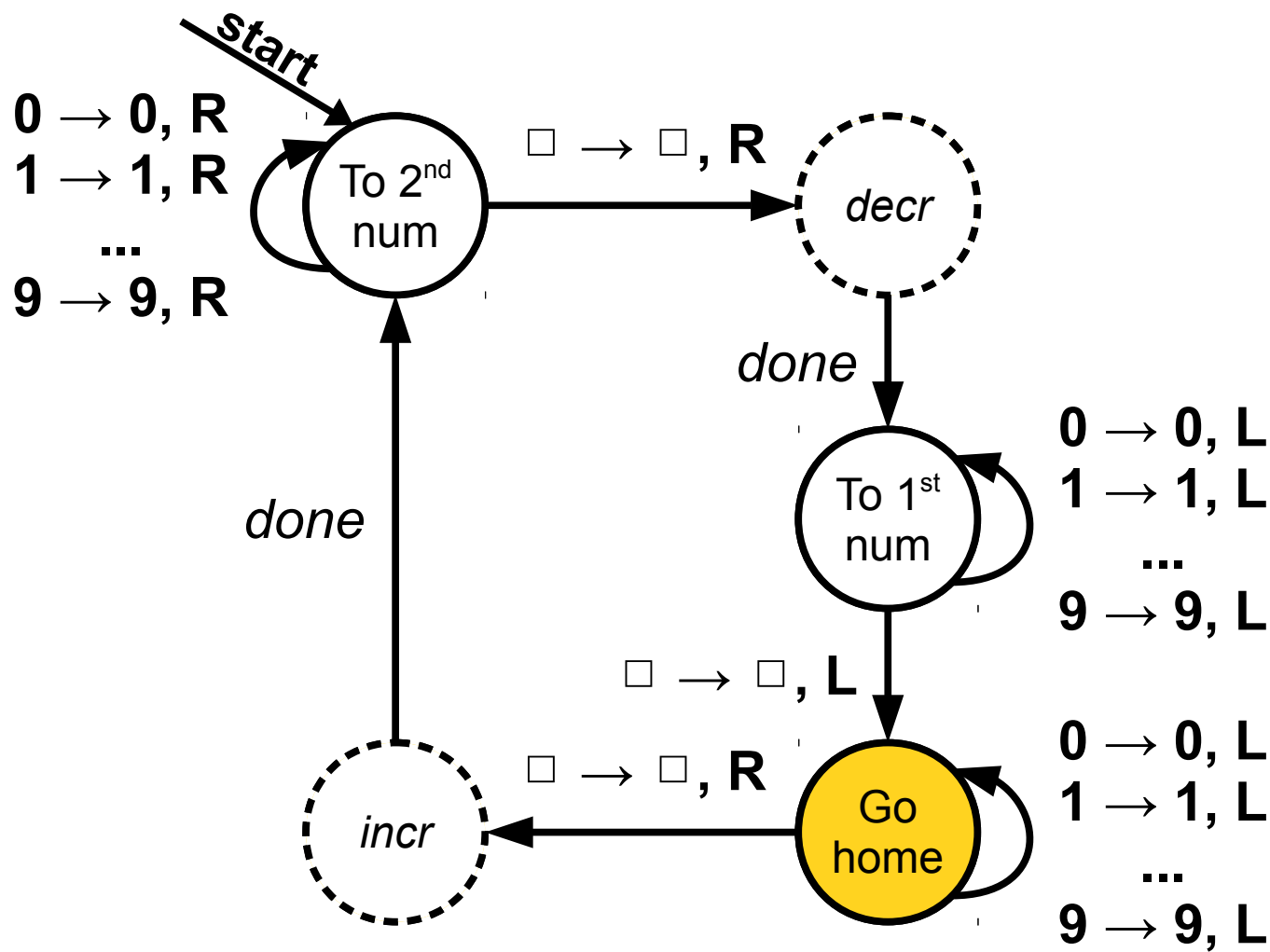


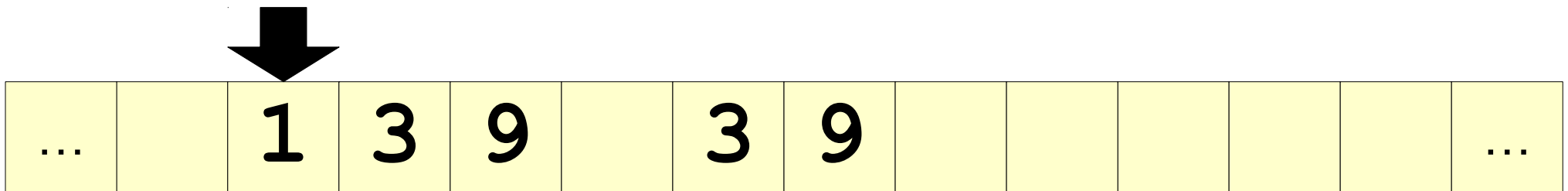
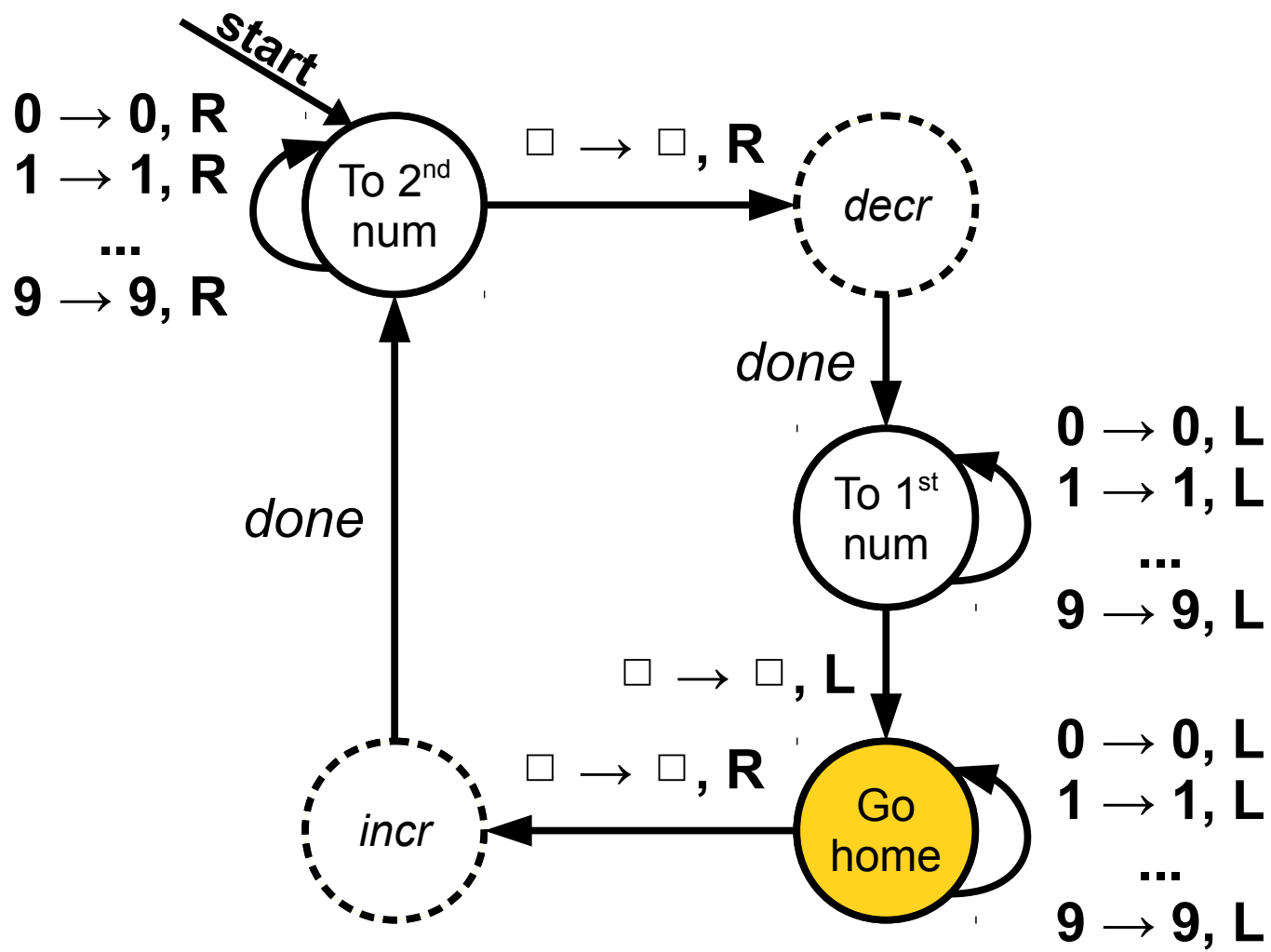


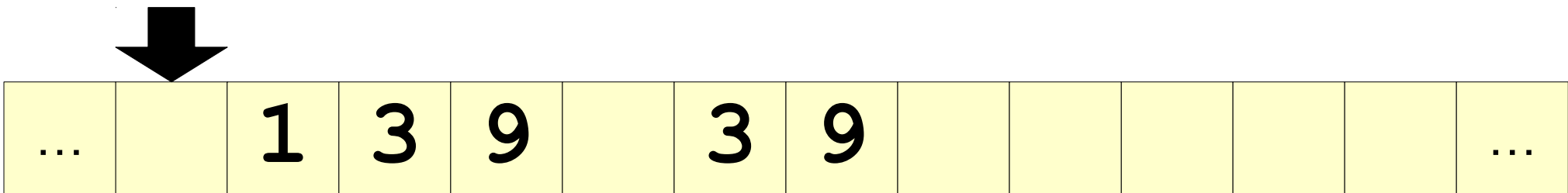
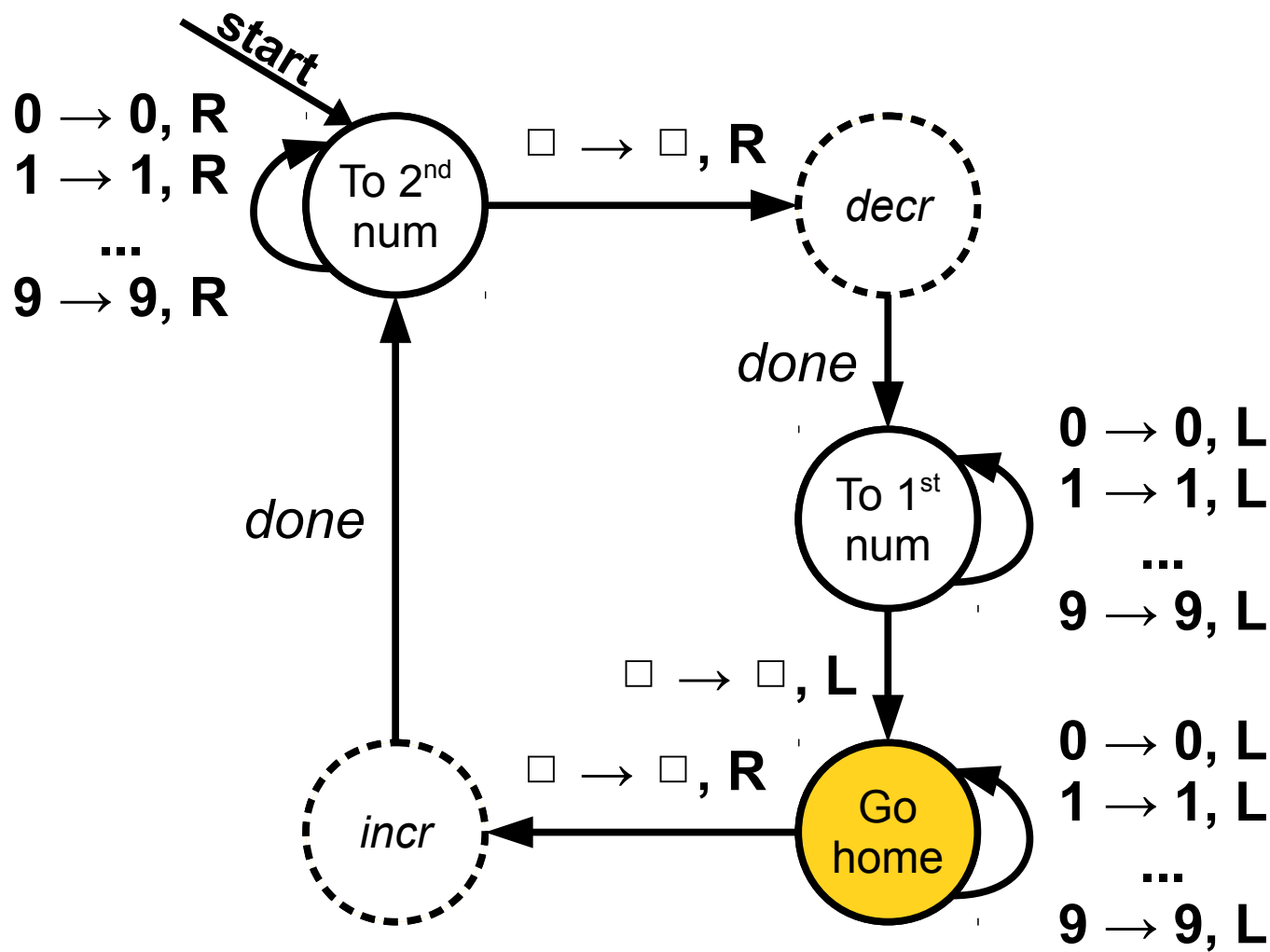


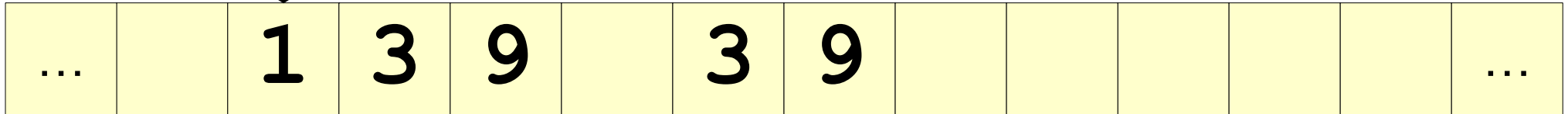
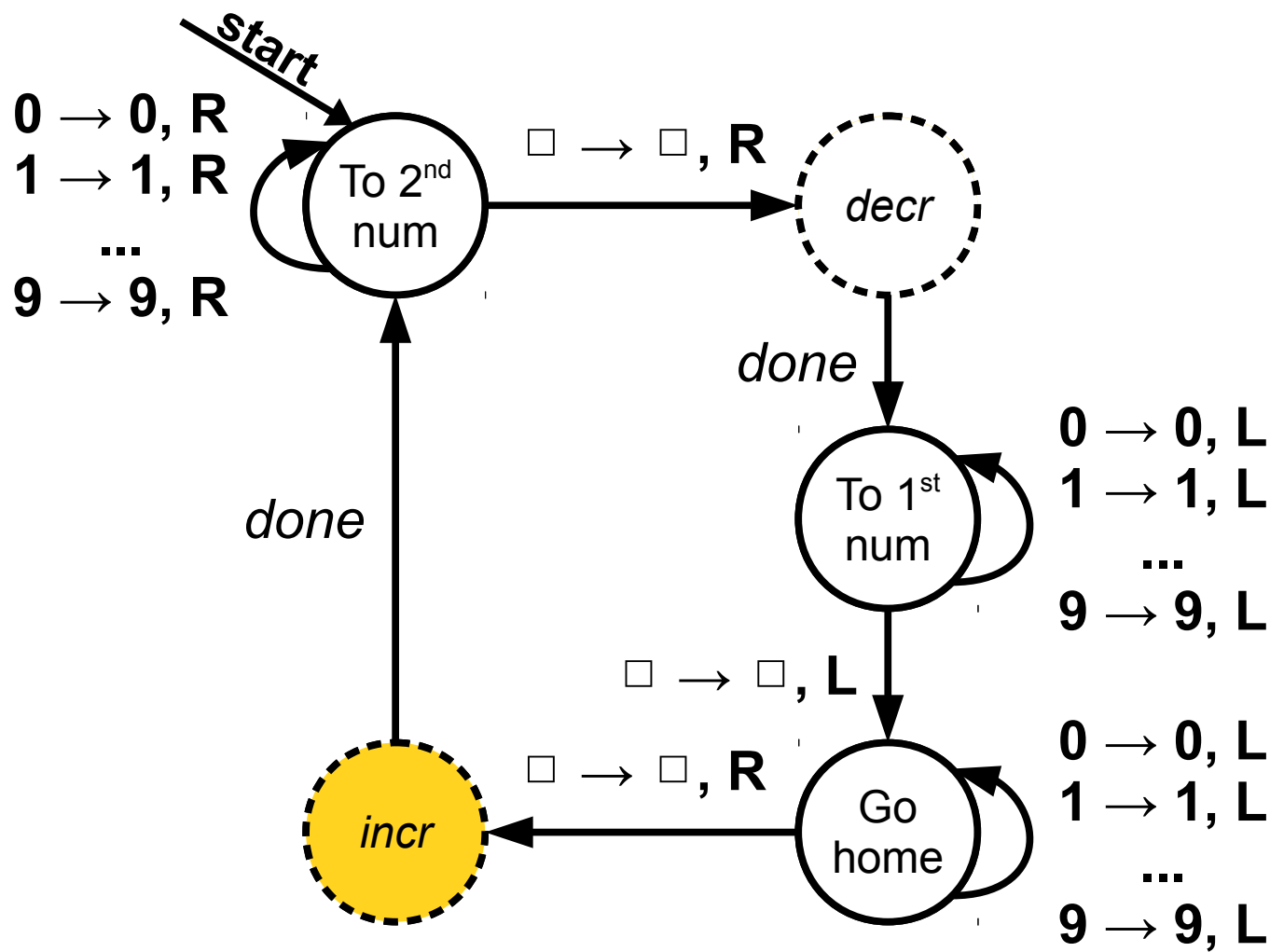


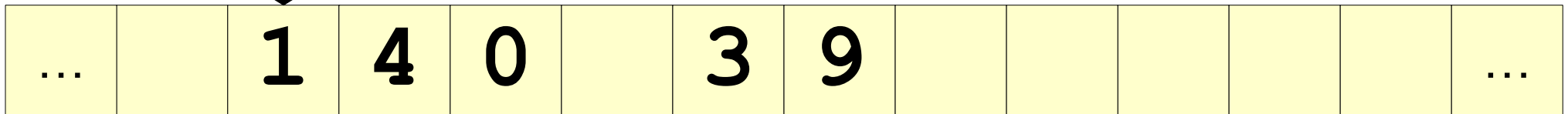
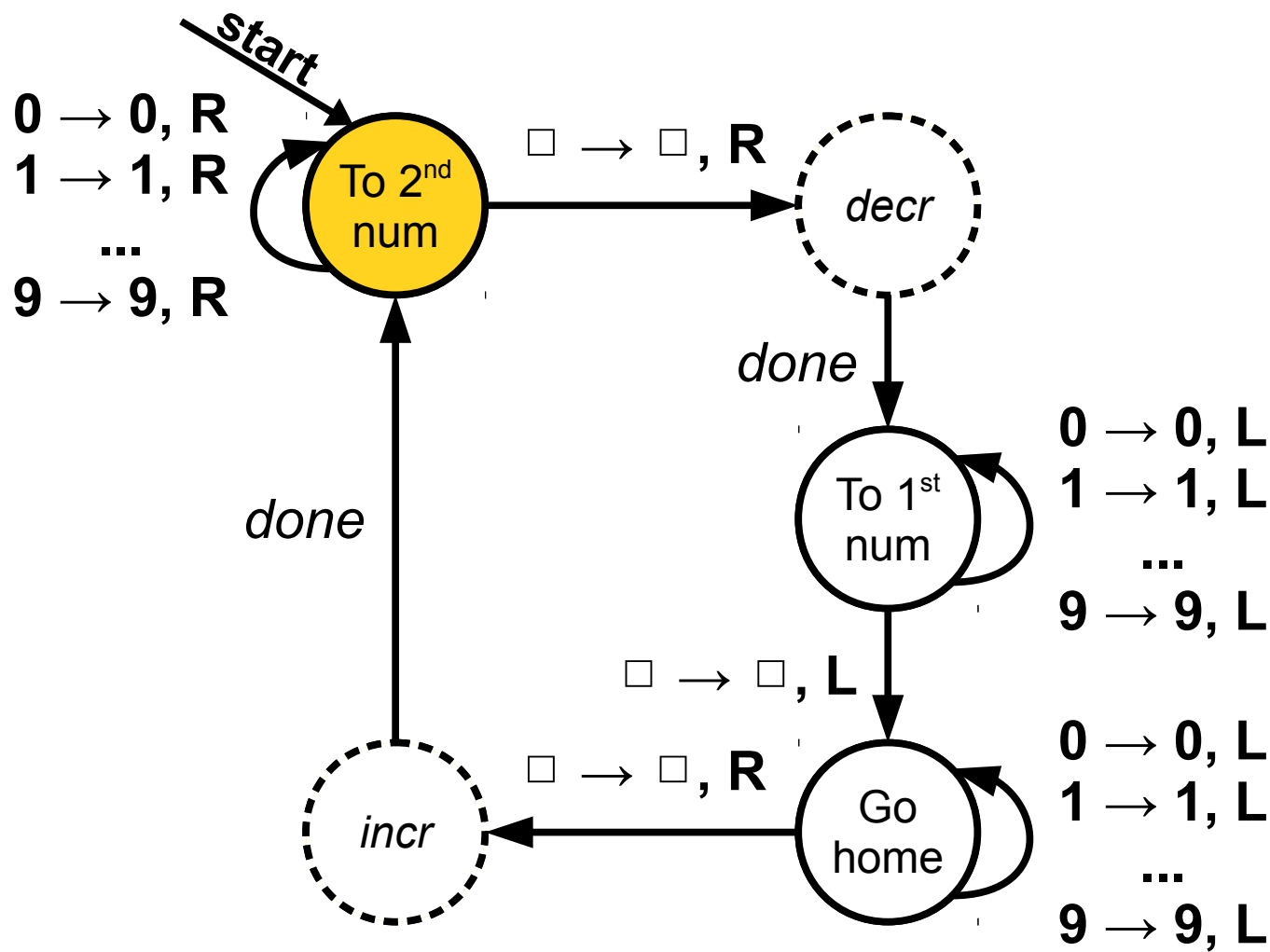


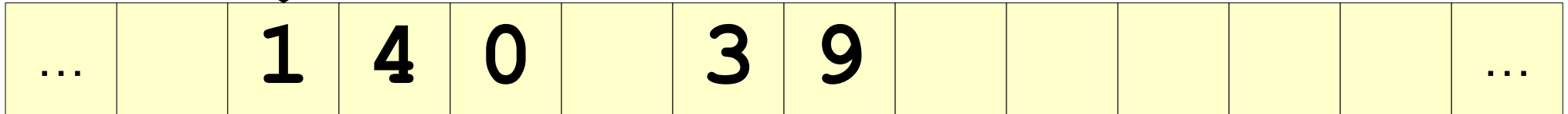
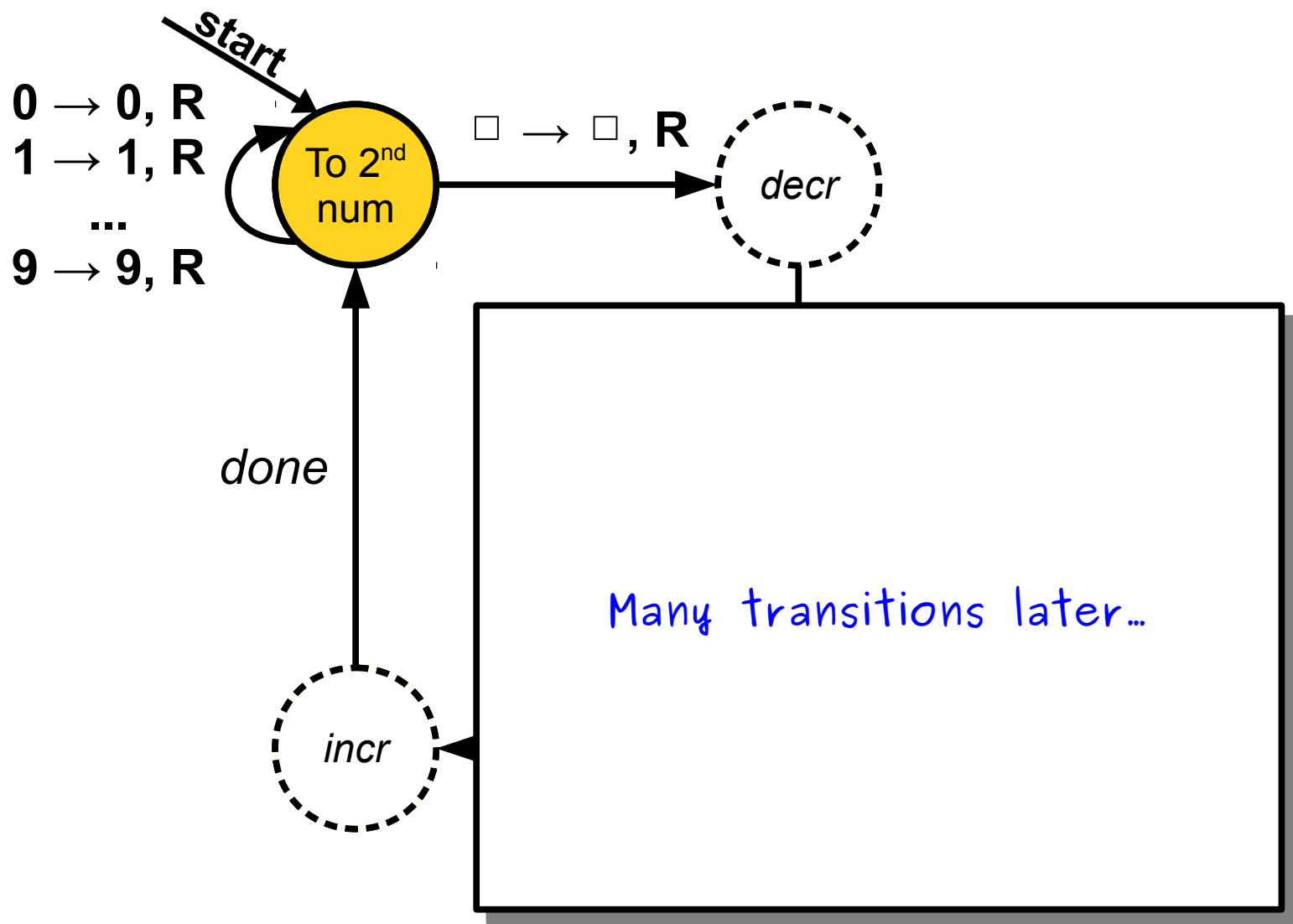


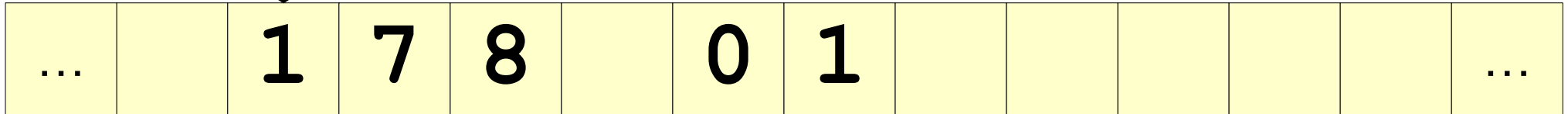
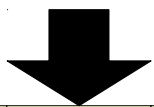
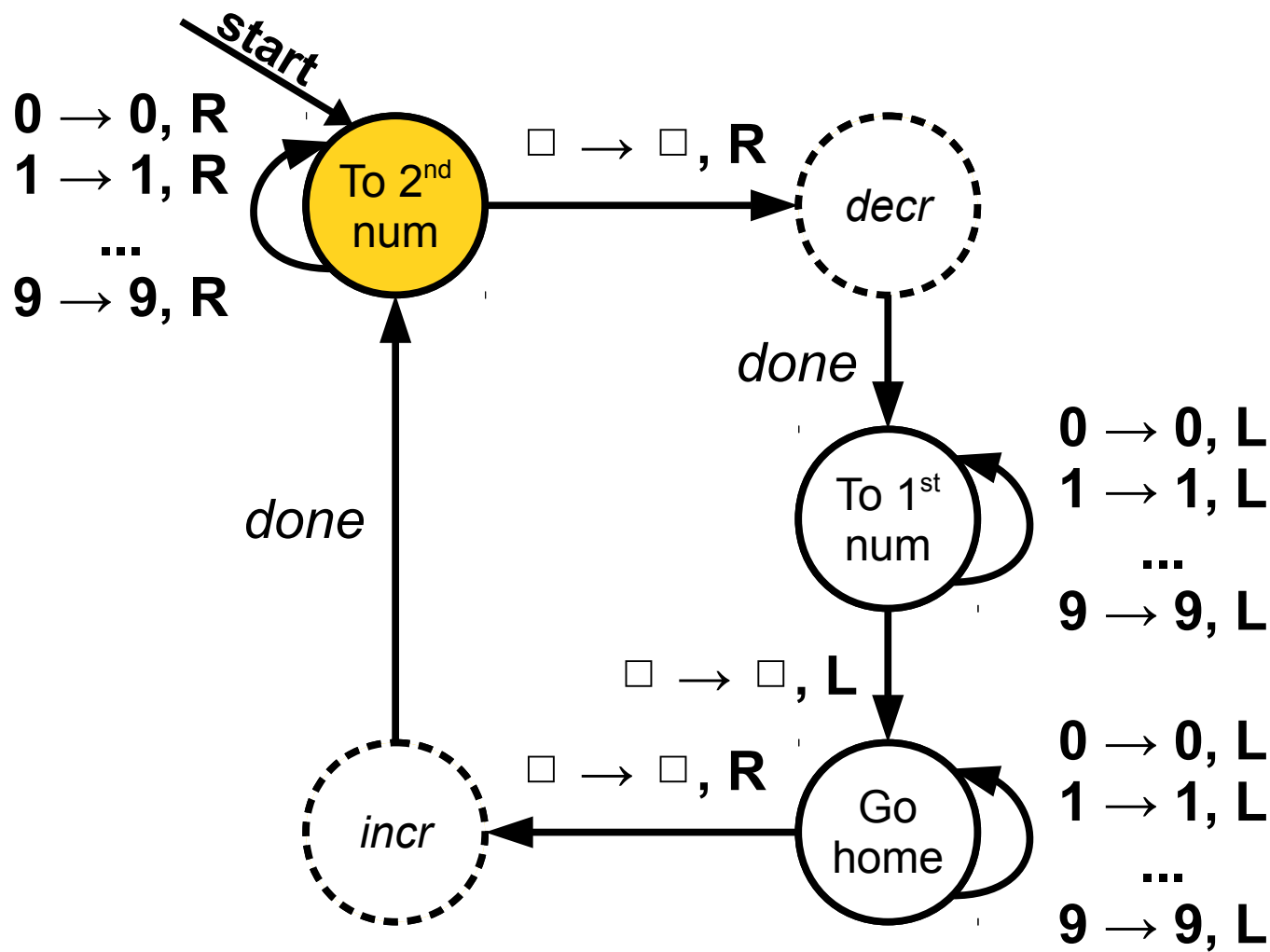


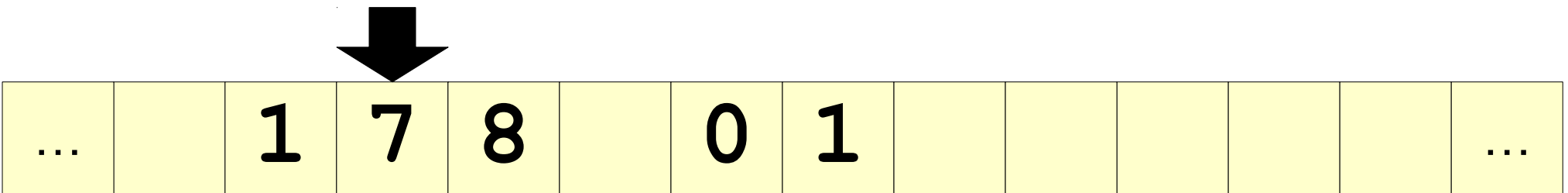
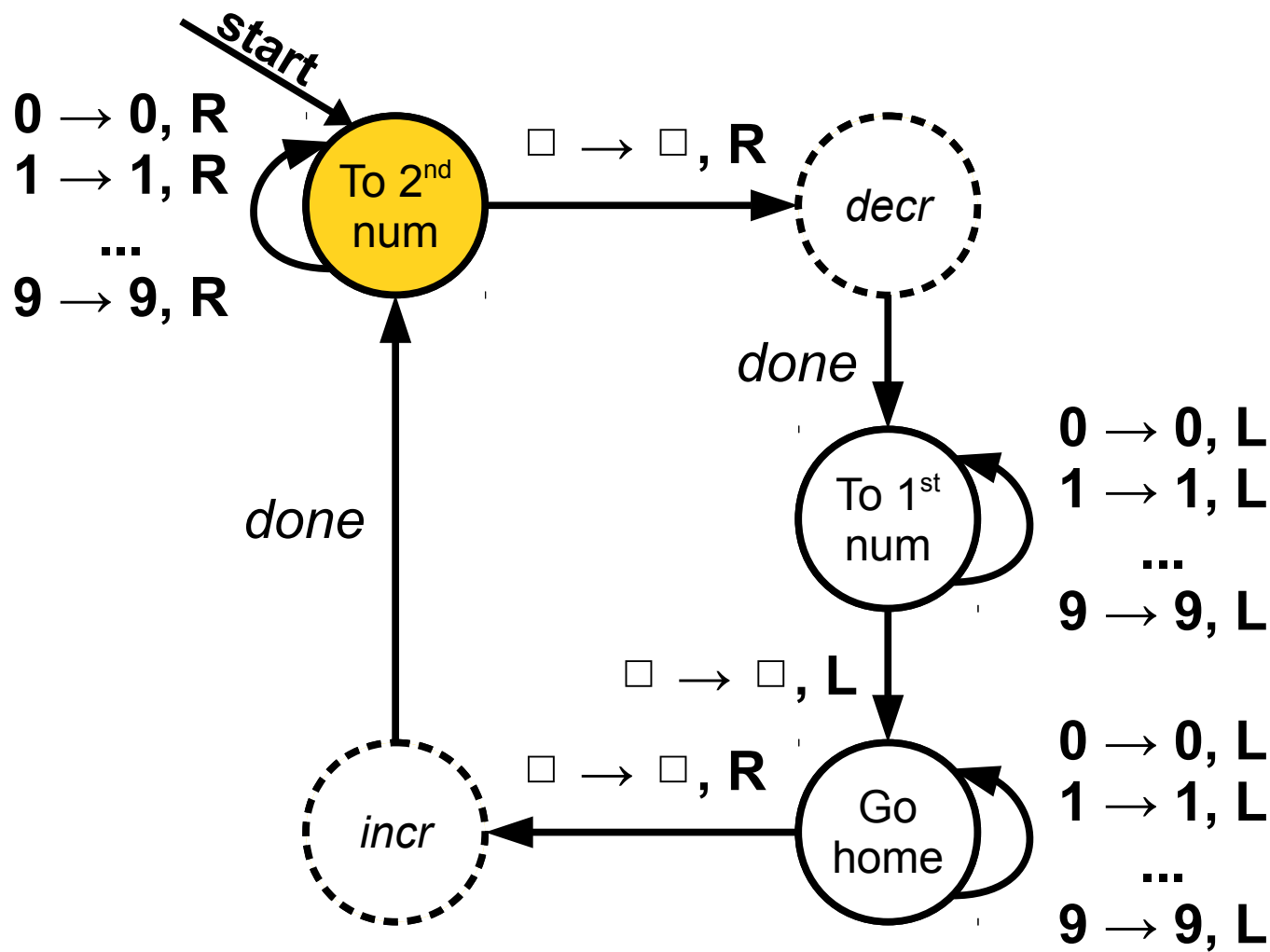


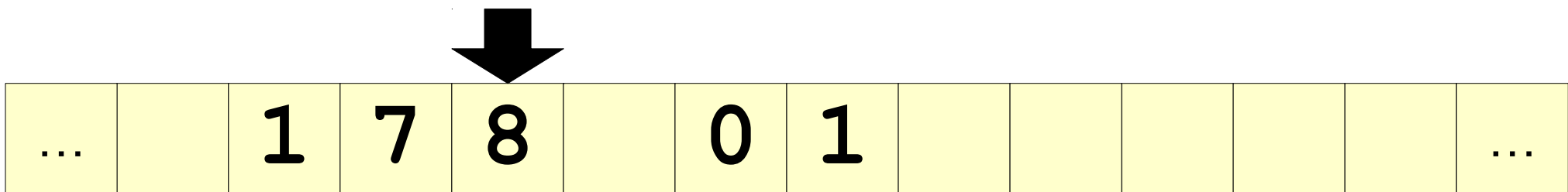
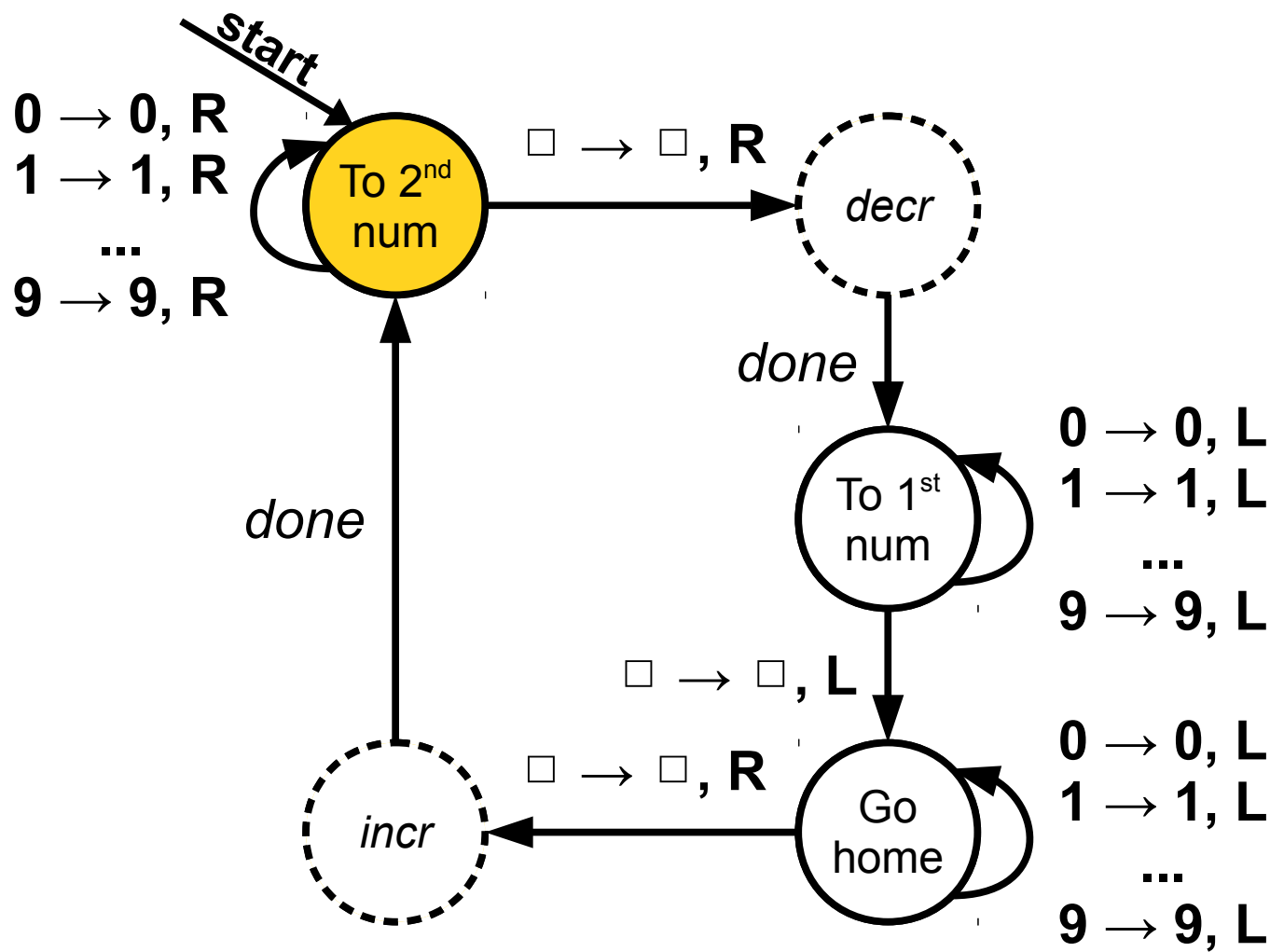


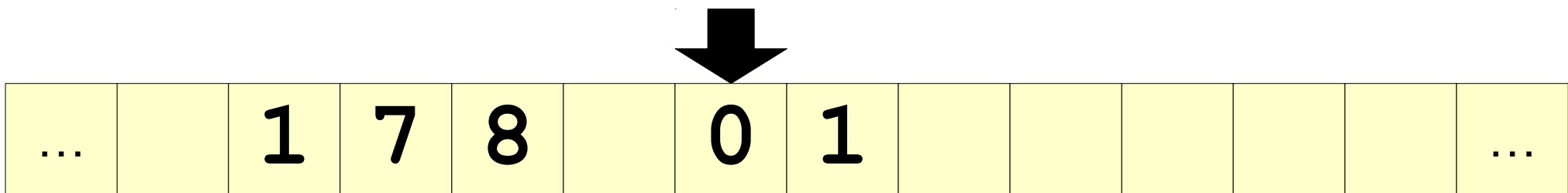
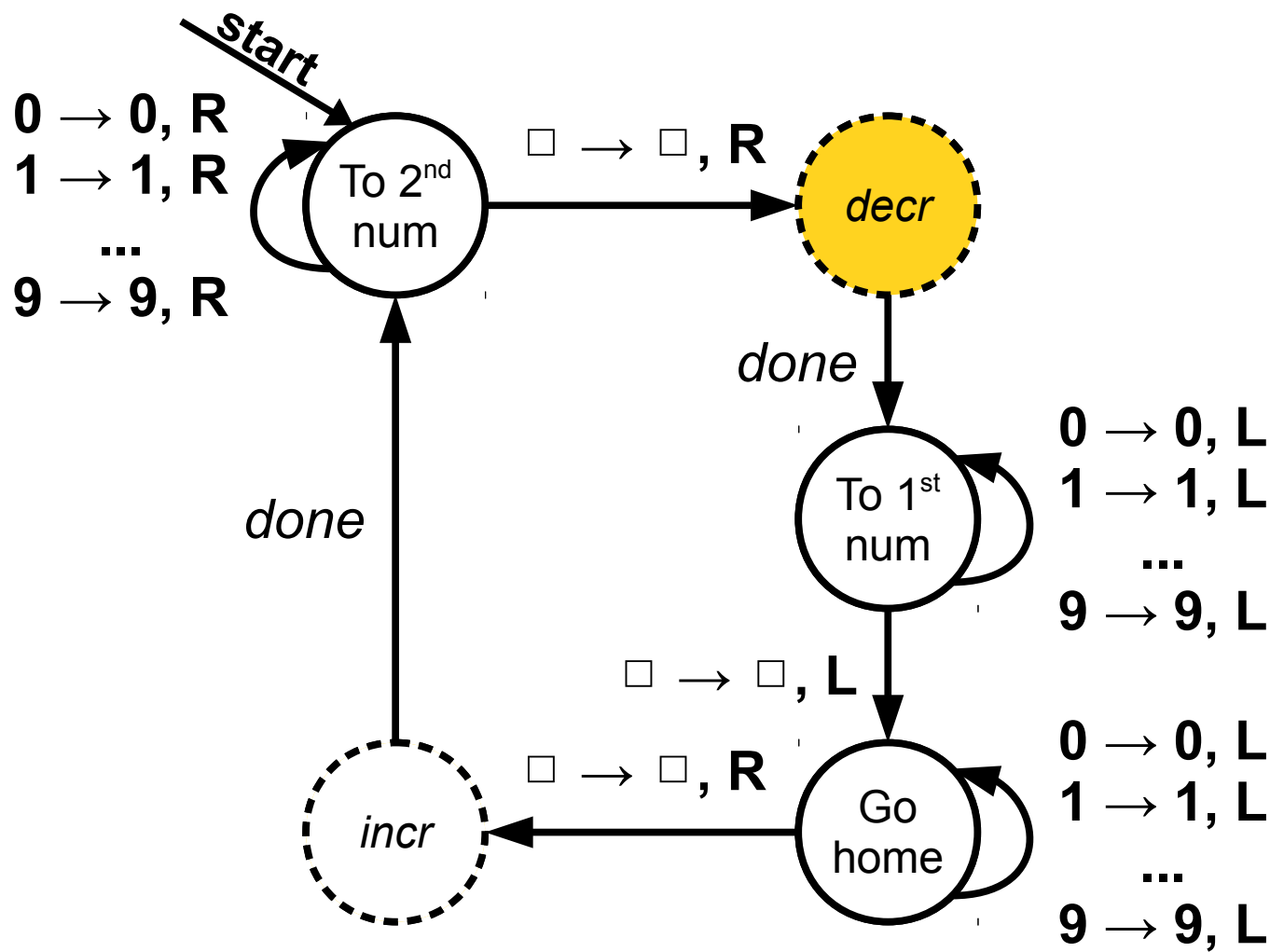


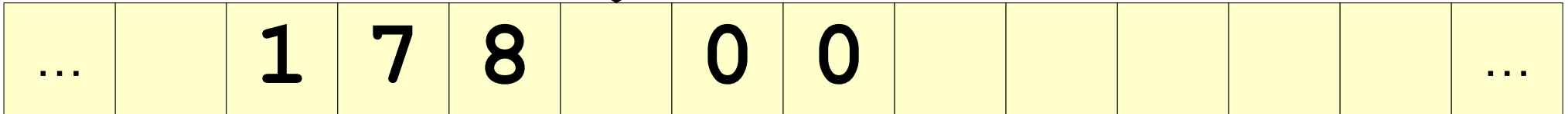
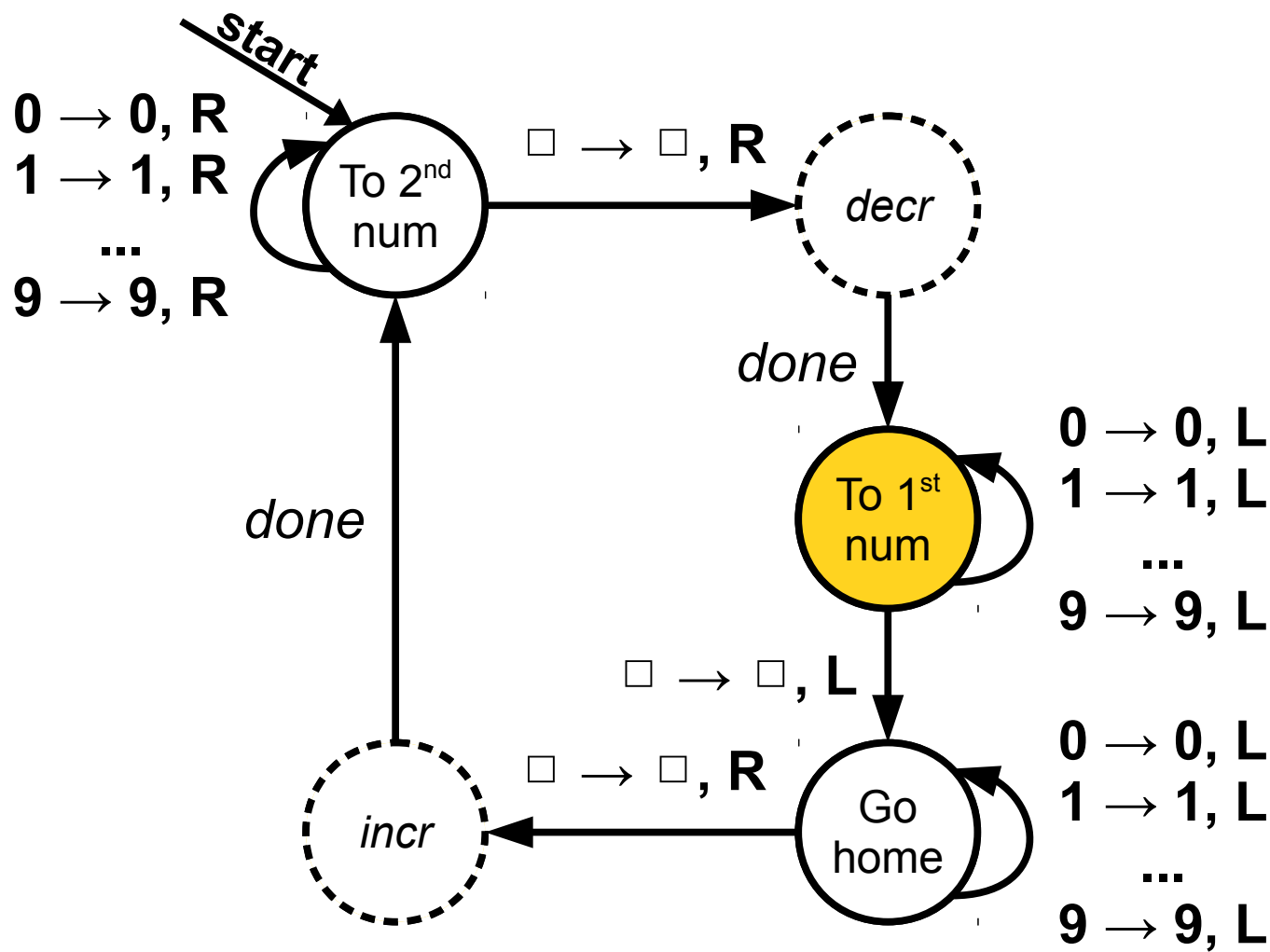


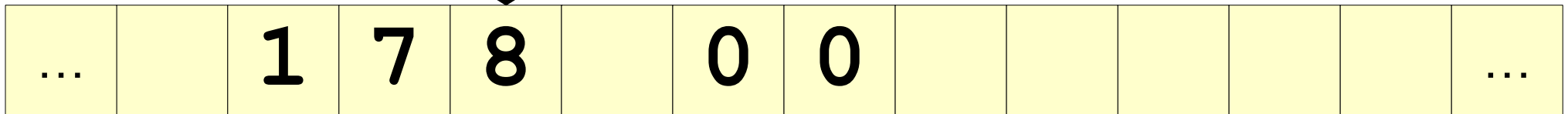
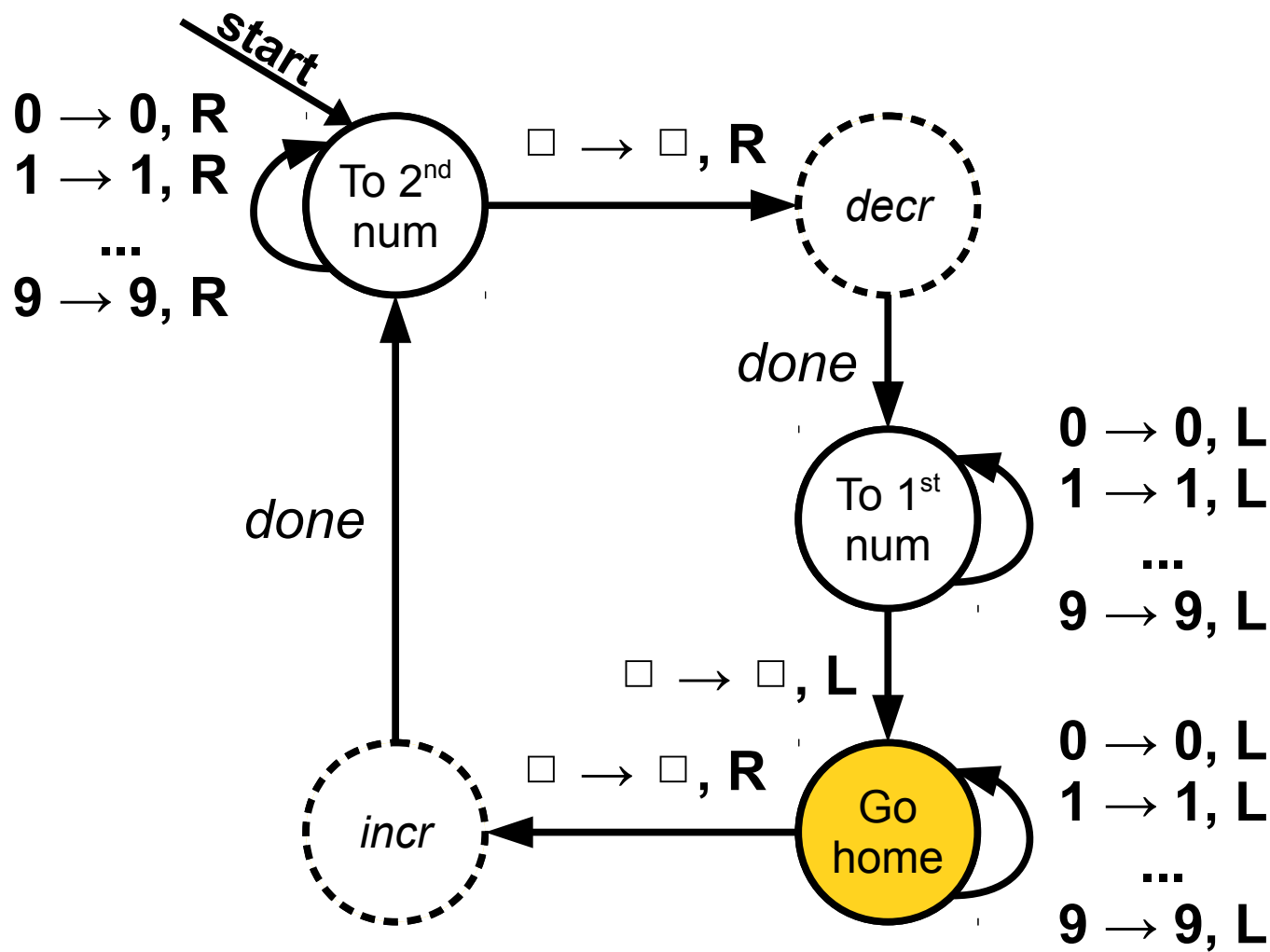


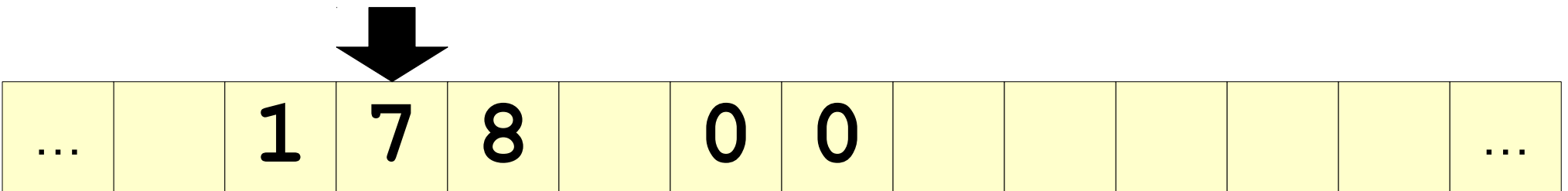
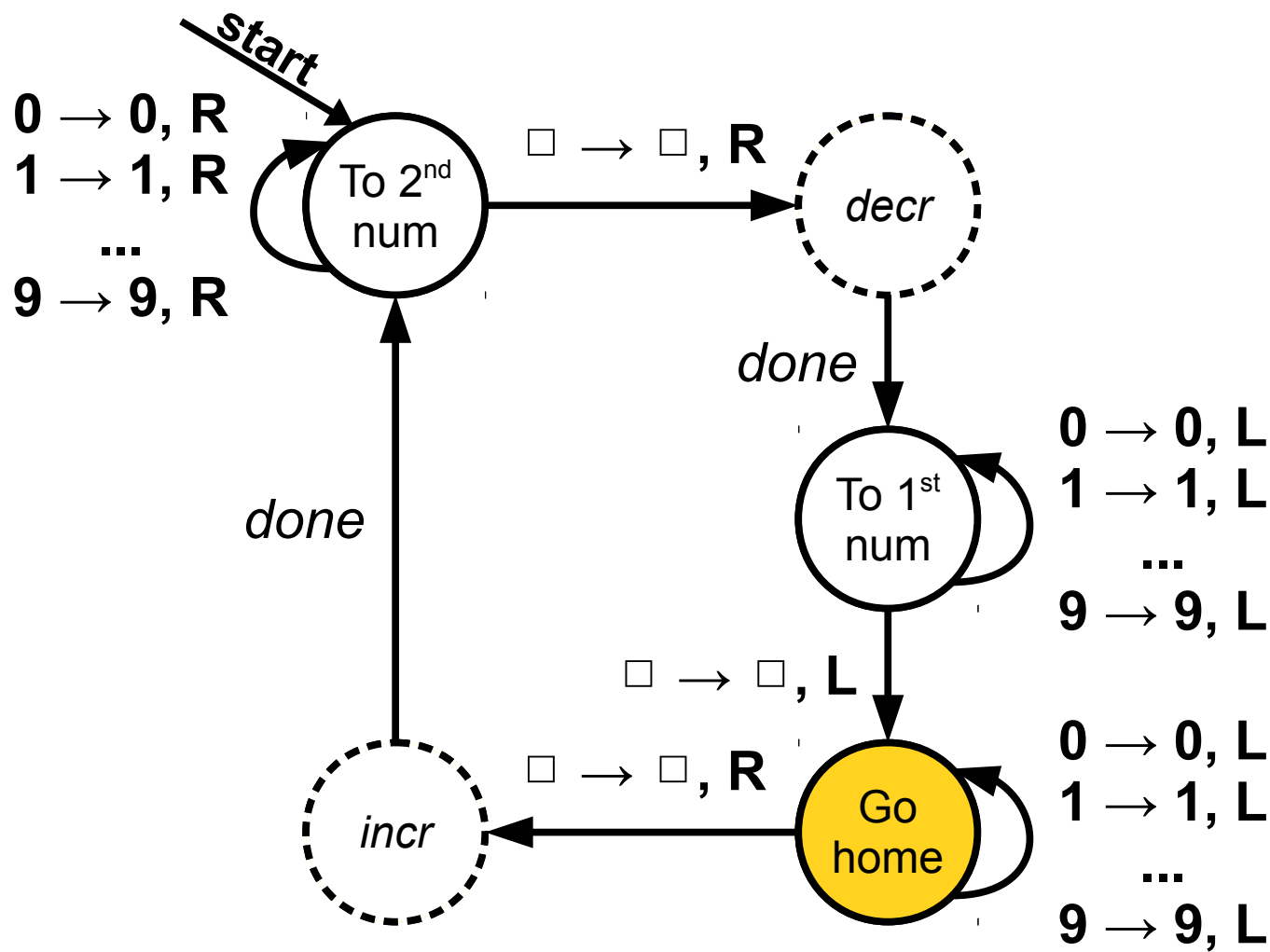


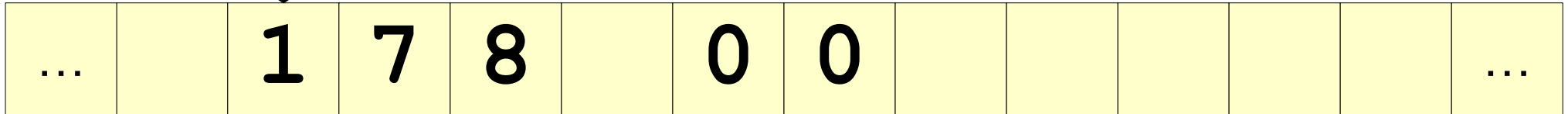
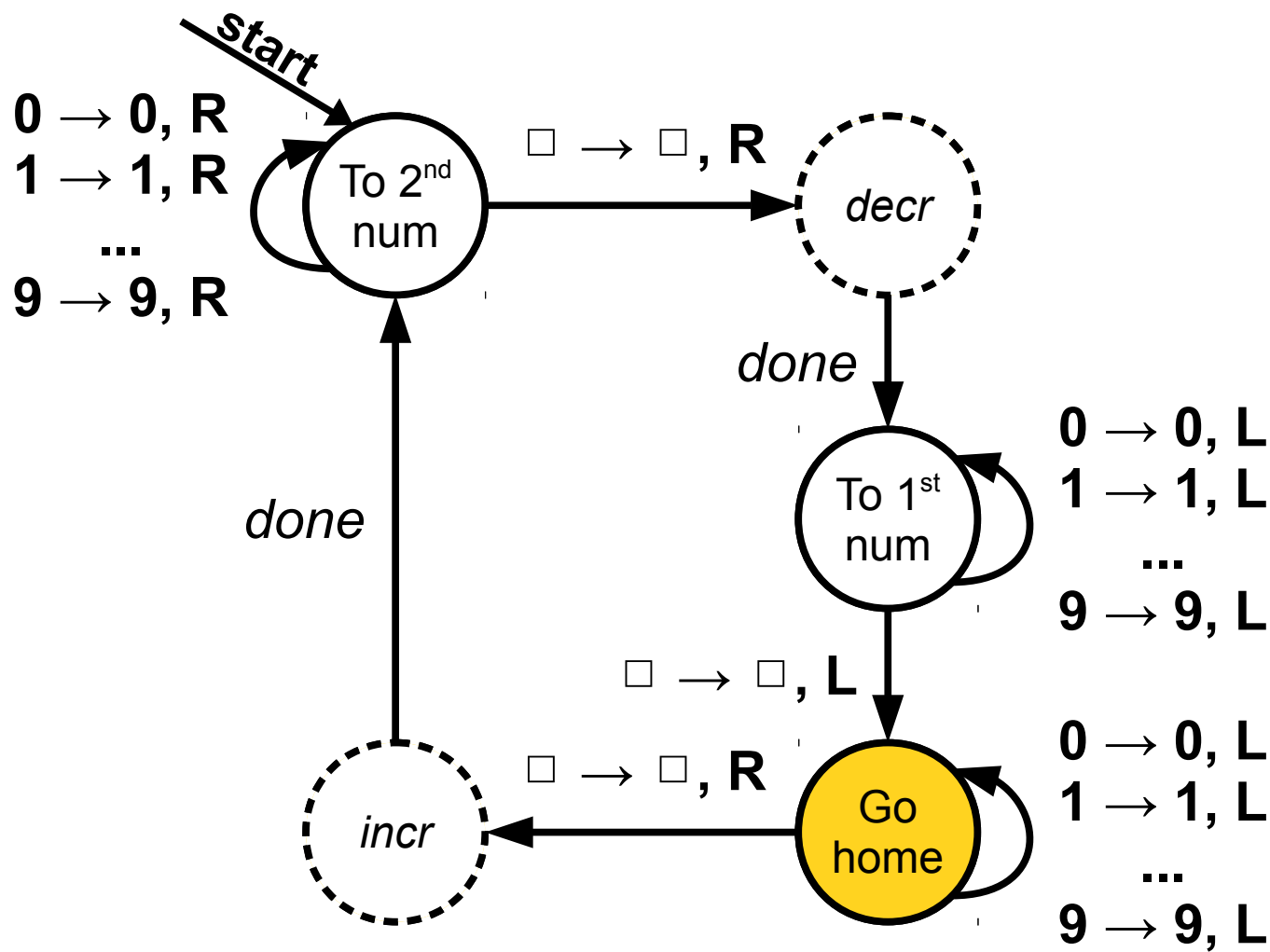


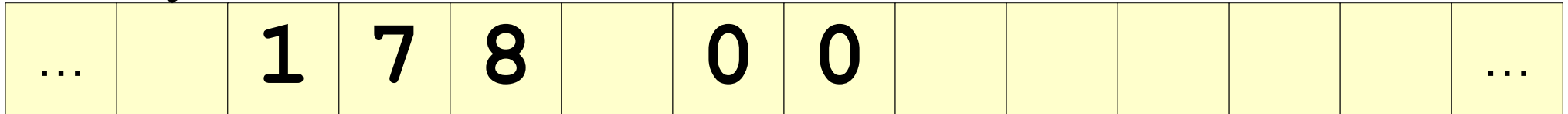
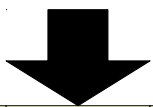
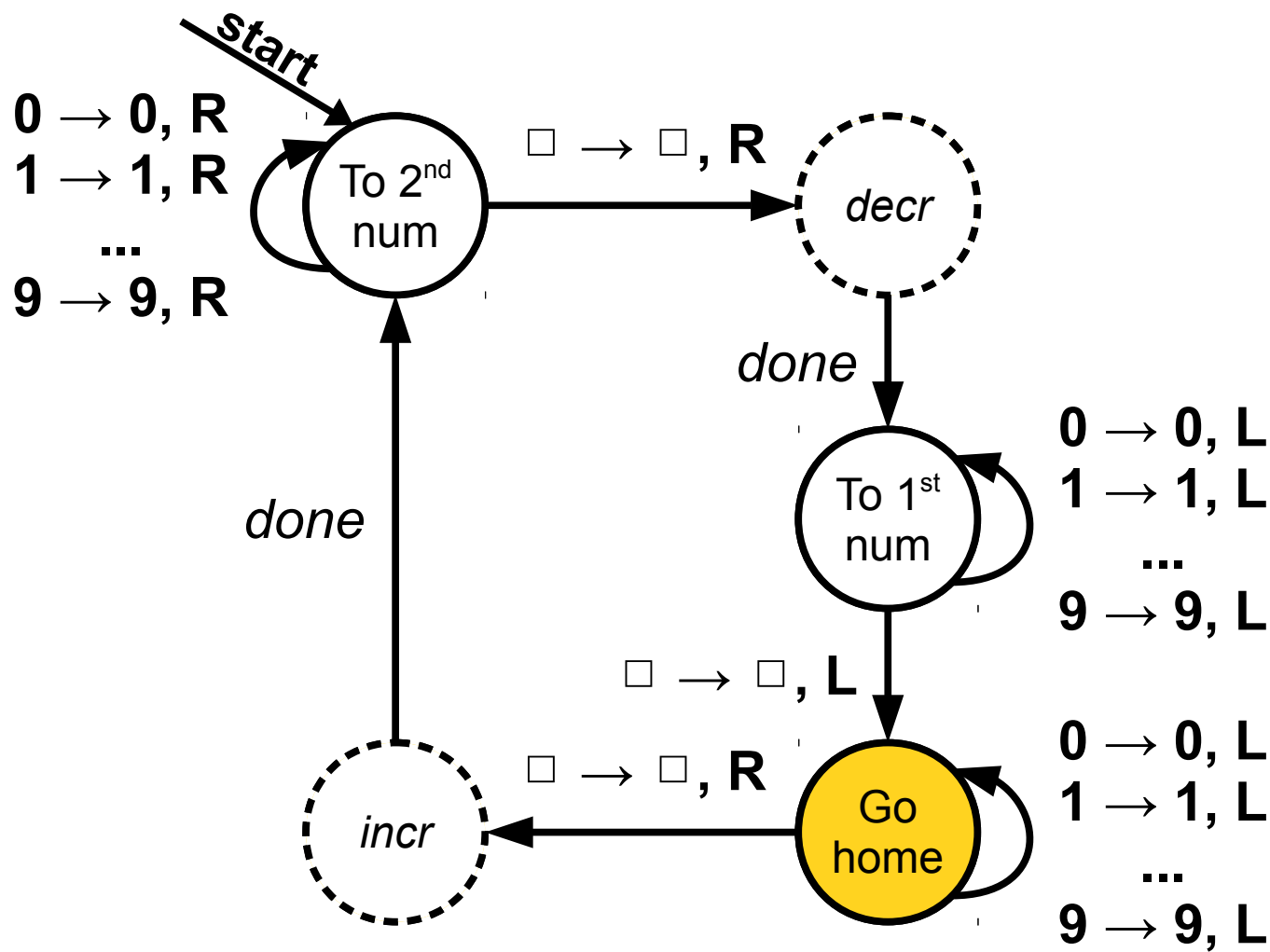


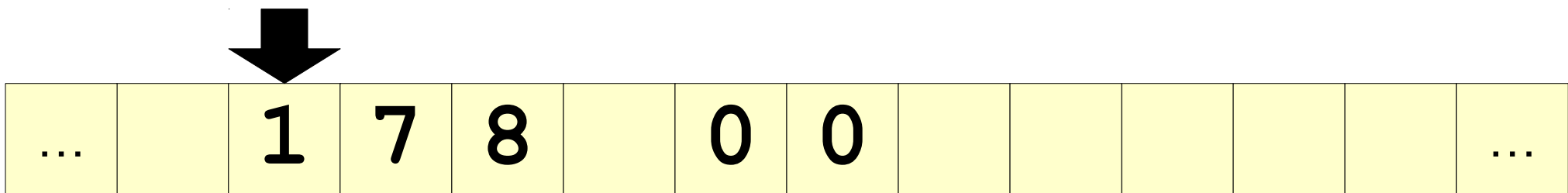
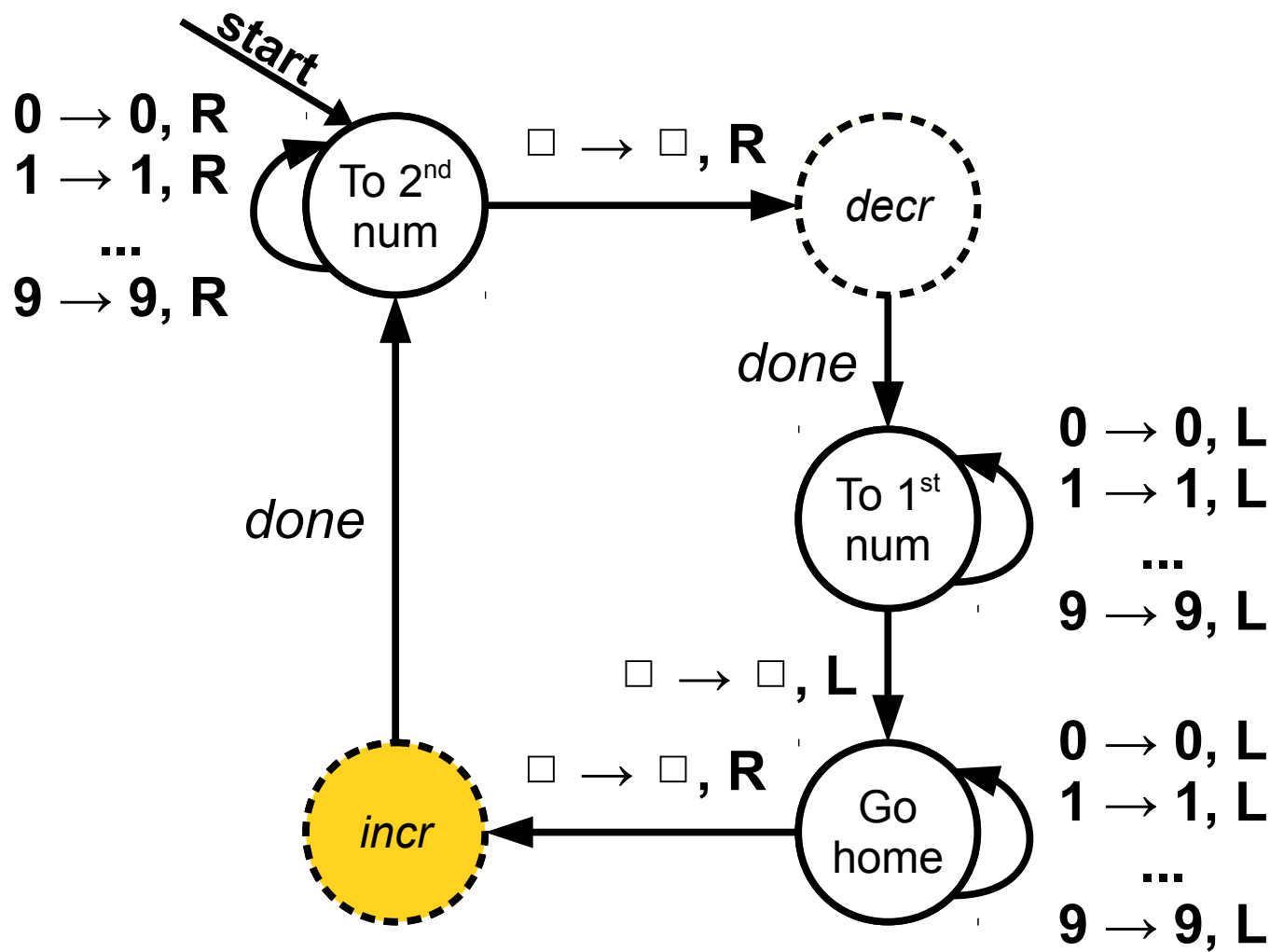


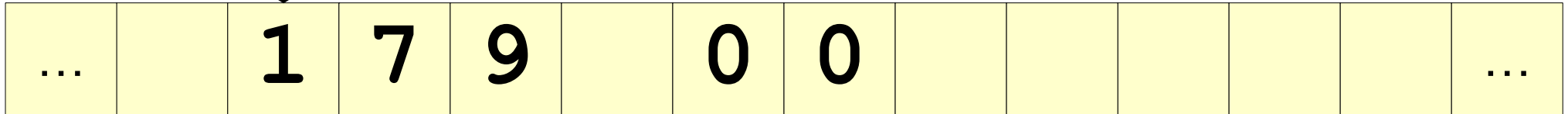
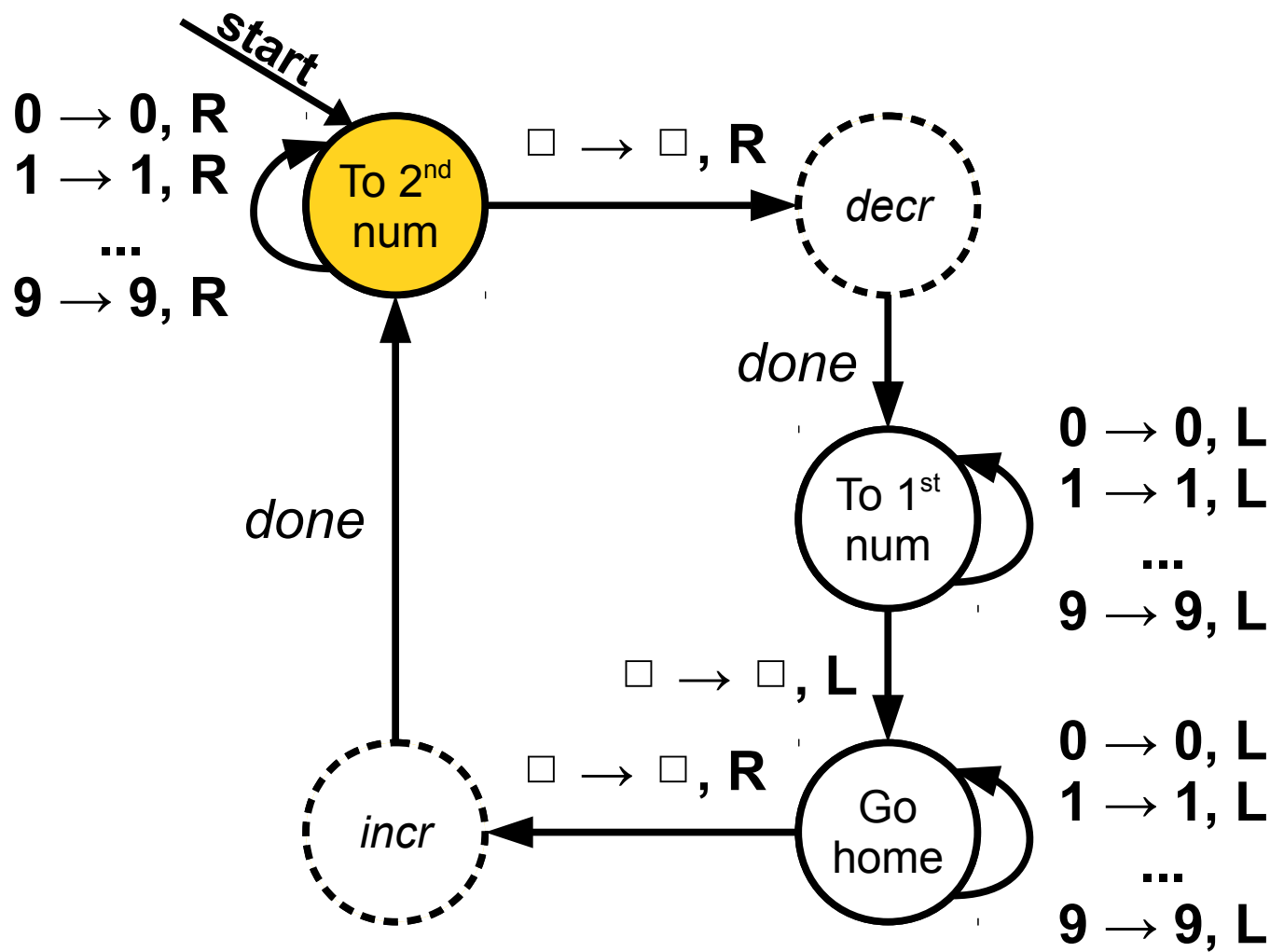


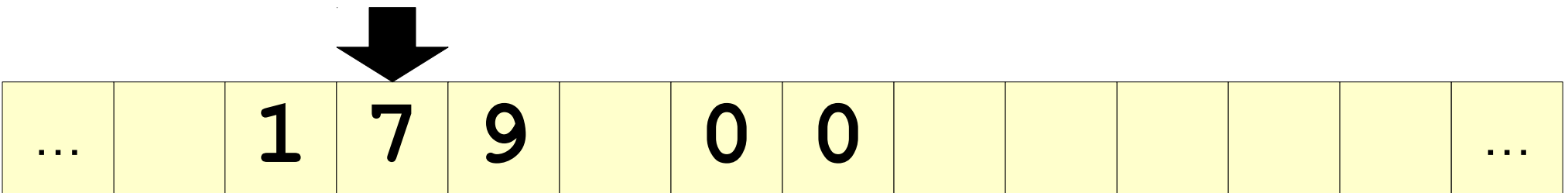
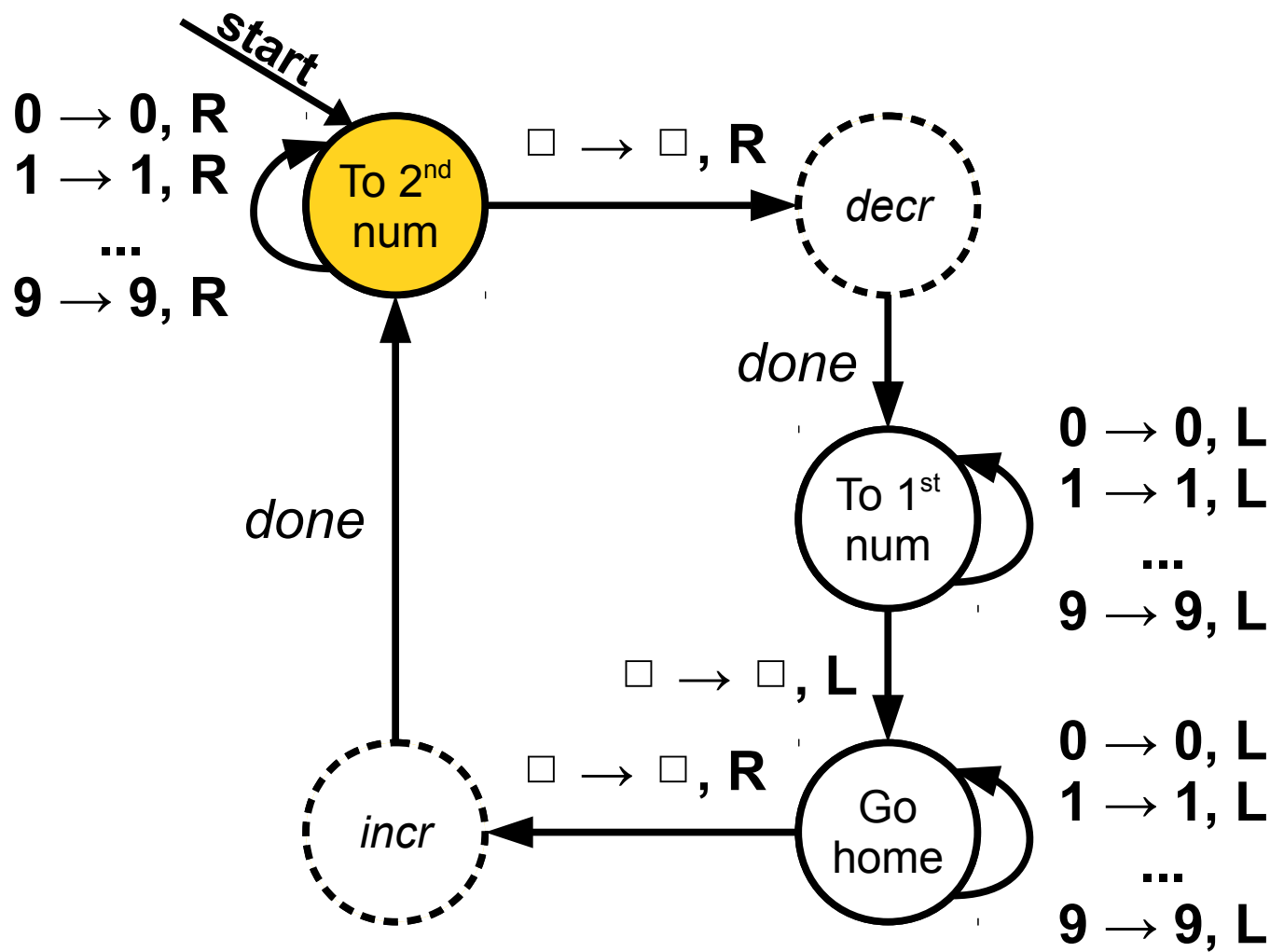


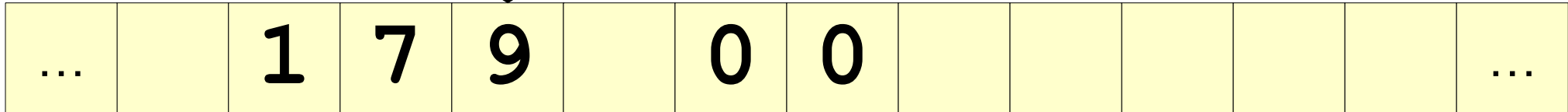
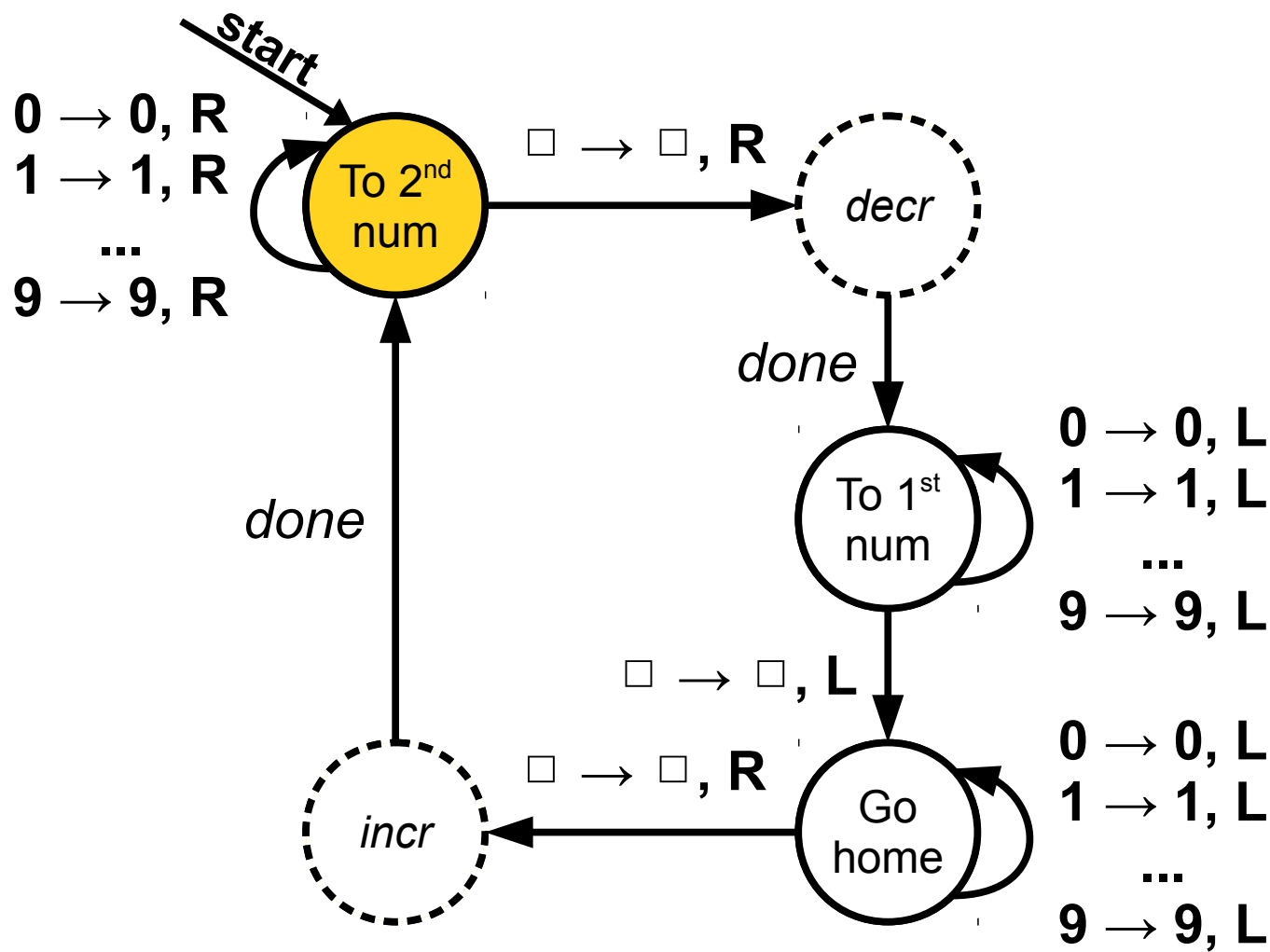


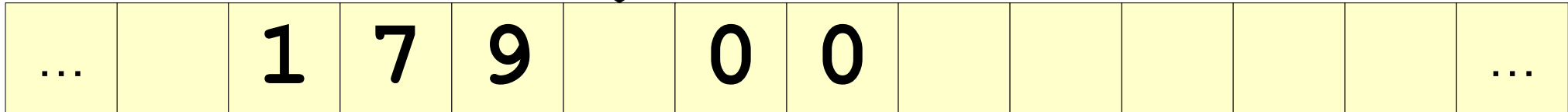
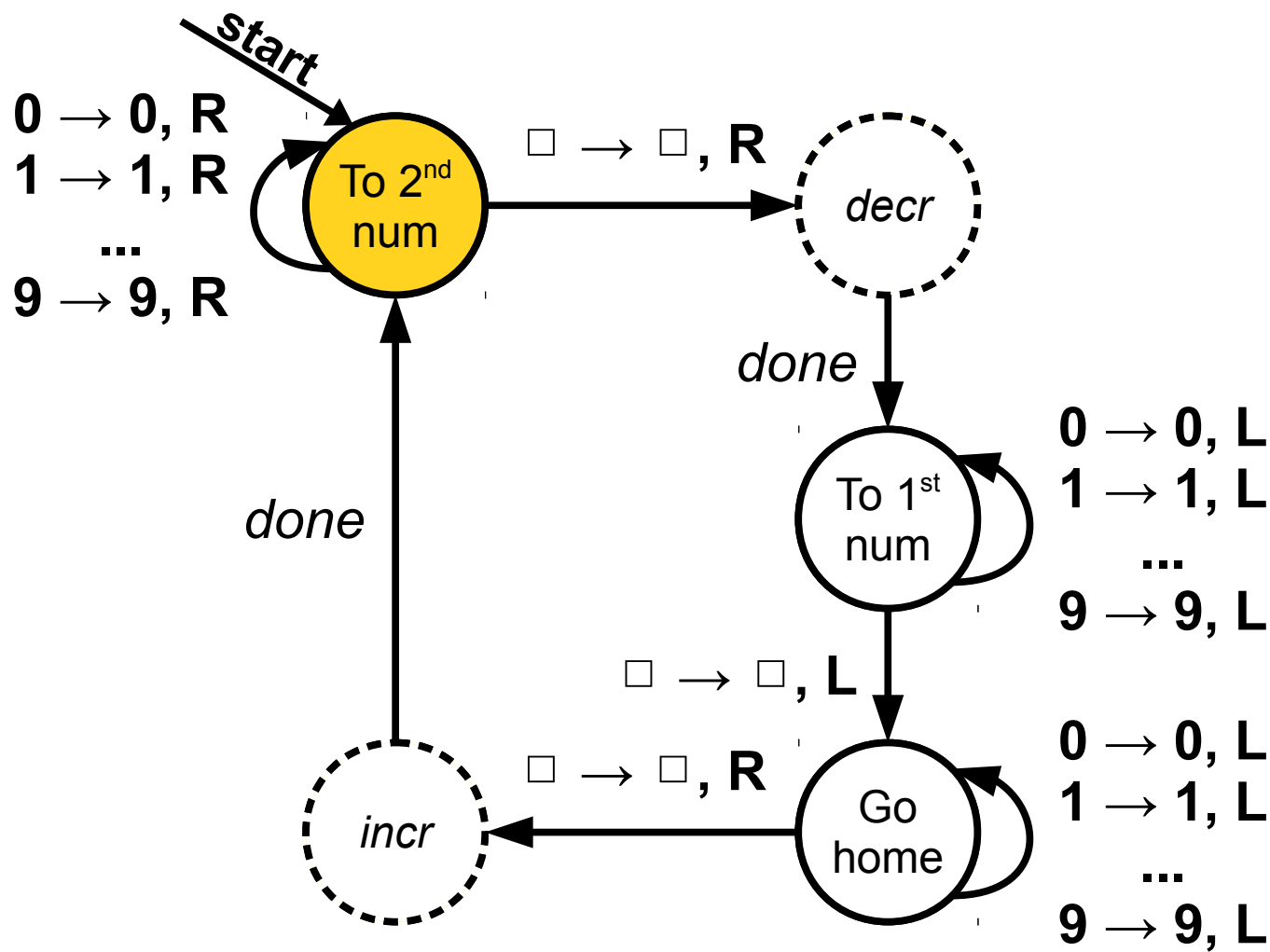


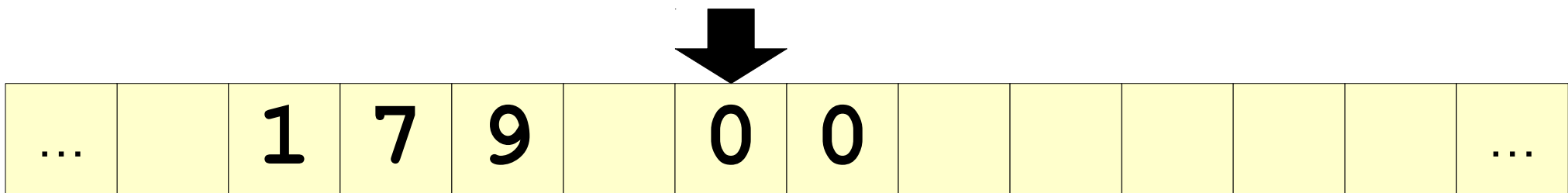
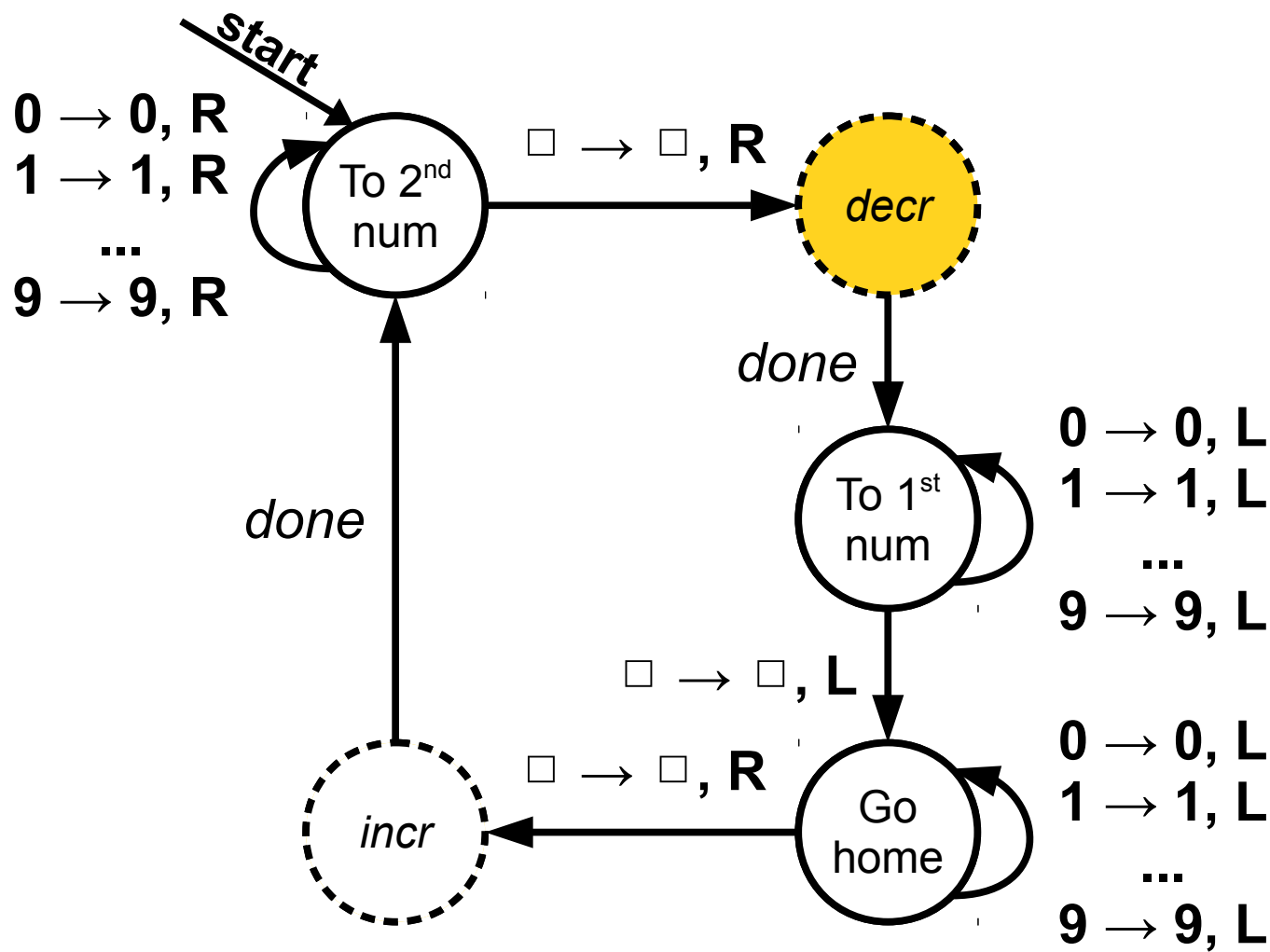


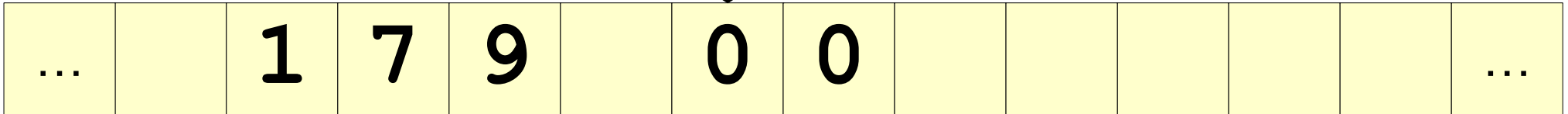
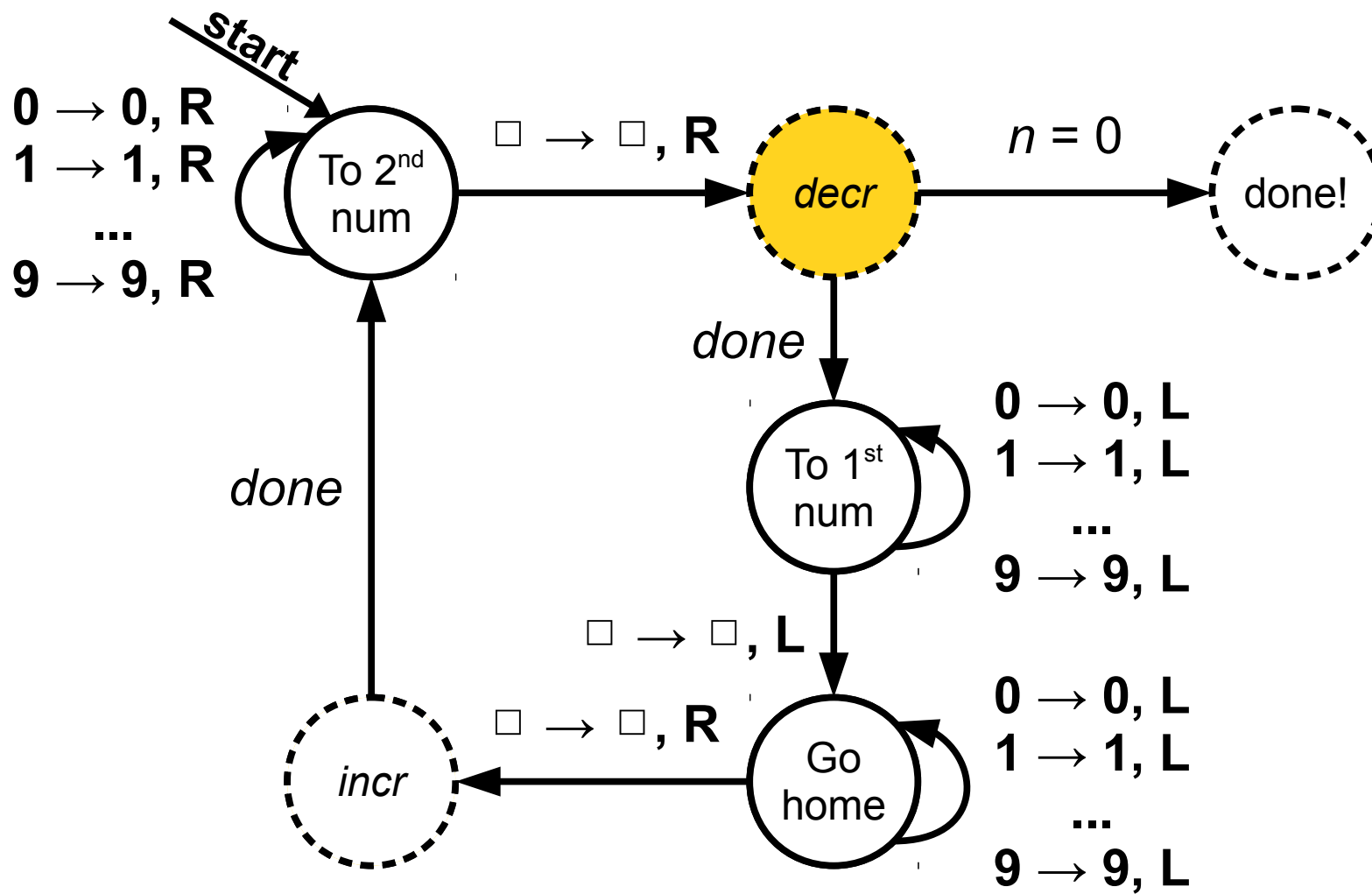


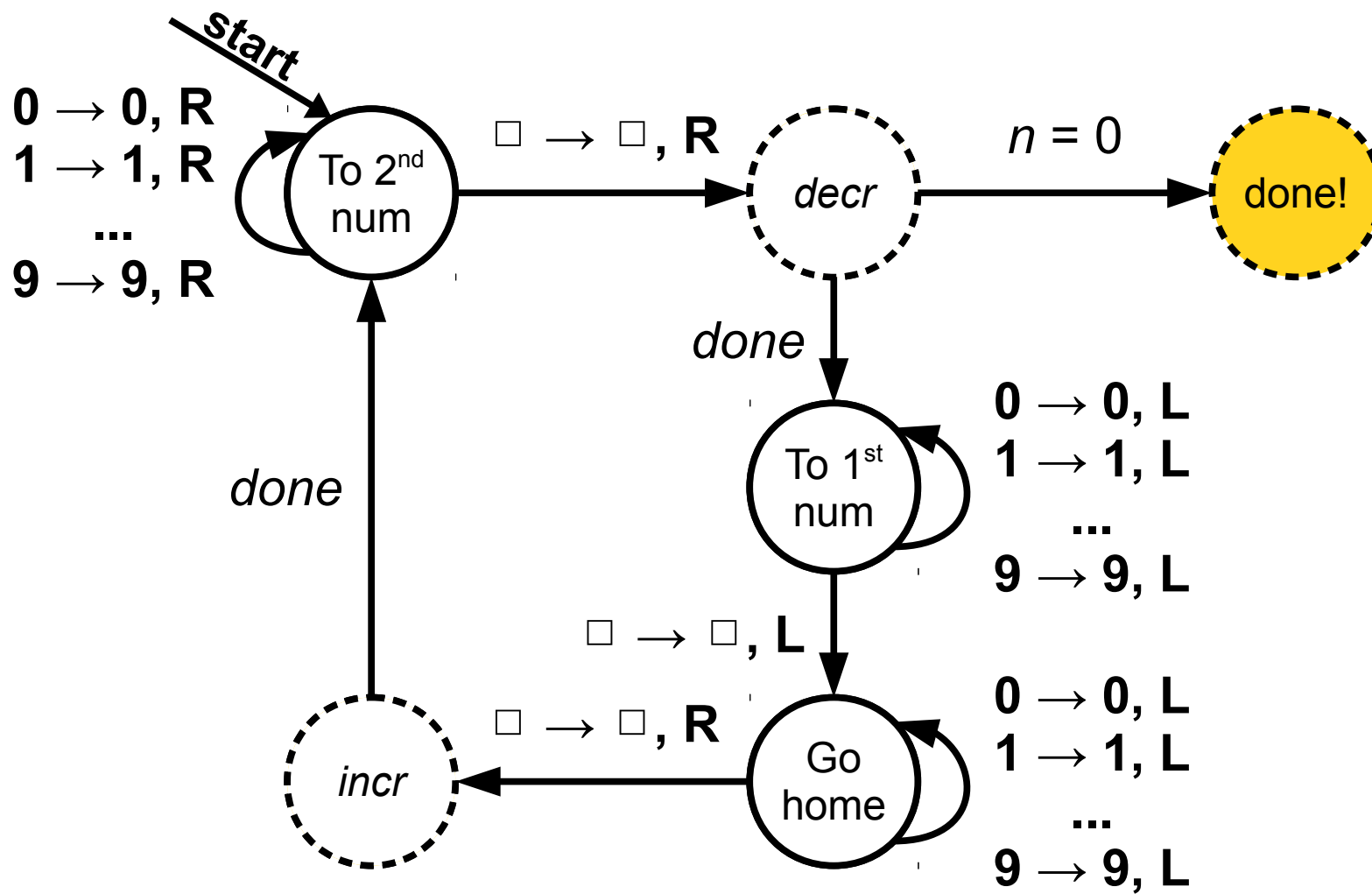


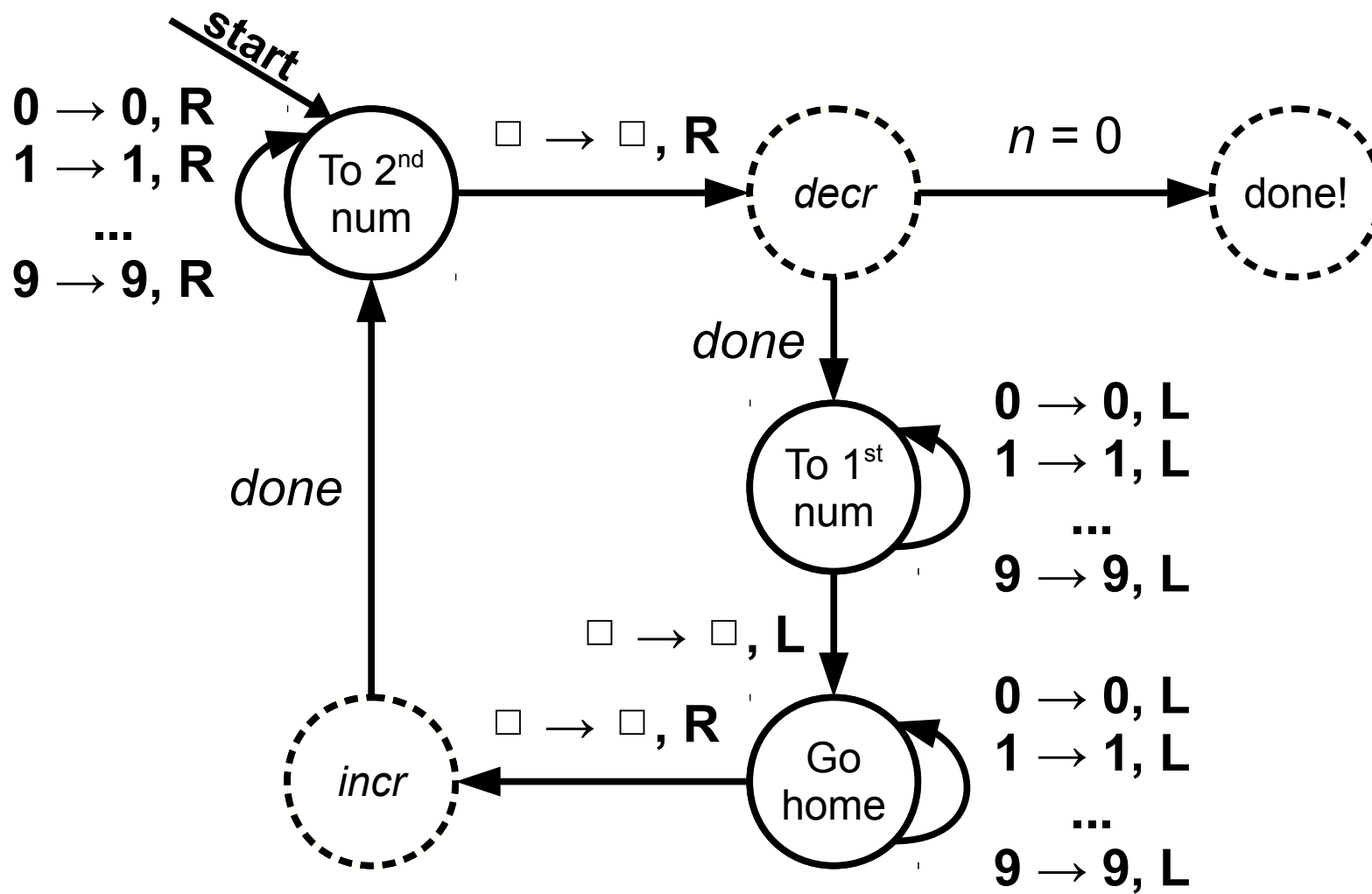






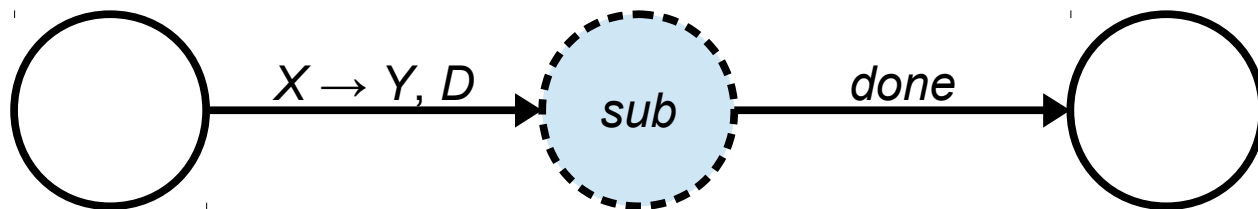






Using Subroutines

- Once you've built a subroutine, you can wire it into another TM with something that, schematically, looks like this:



- Intuitively, this corresponds to transitioning to the start state of the subroutine, then replacing the “done” state of the subroutine with the state at the end of the transition.

Bonus Fun Read

- Check out the short story
“[The Feeling of Power](#)” by Isaac Asimov,
set in a world where people have
forgotten how to add and subtract.