Chapter (3)

Context-Free Languages

Dr. Mohammed AbdelFattah

Context-Free Languages

$\{a^n b^n : n \geq 0\}$ $\{ww^R\}$

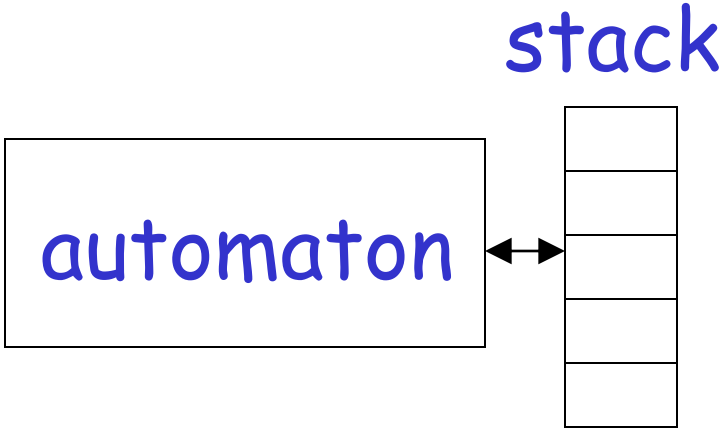
Regular Languages

a^*b^* $(a+b)^*$

Context-Free Languages

Context-Free Grammars

Pushdown Automata



Pushdown Automata
PDAs

Reference

For this lecture, Please contact with the following links:

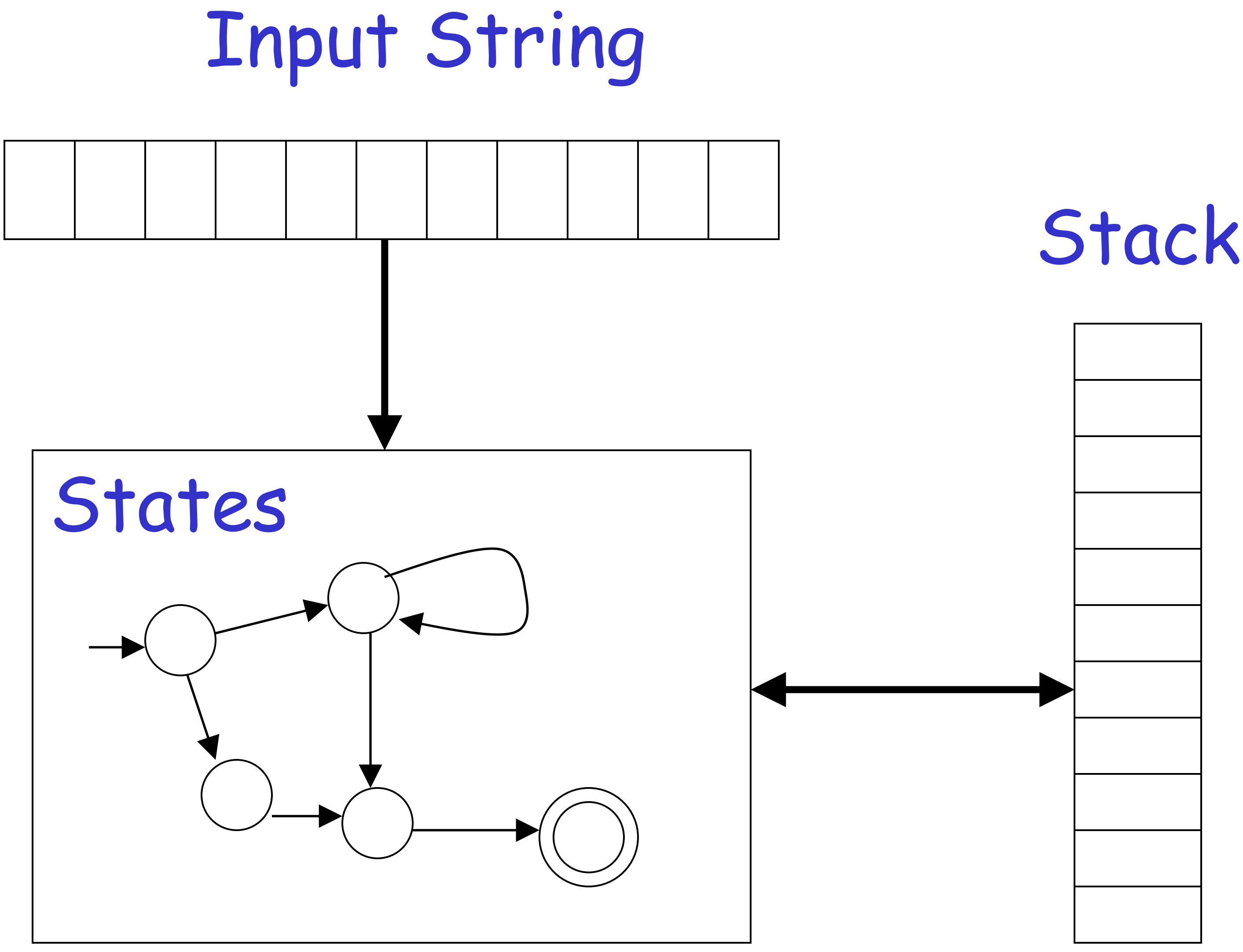
Part1:

https://www.youtube.com/watch?v=XqX-BH5XA9Y

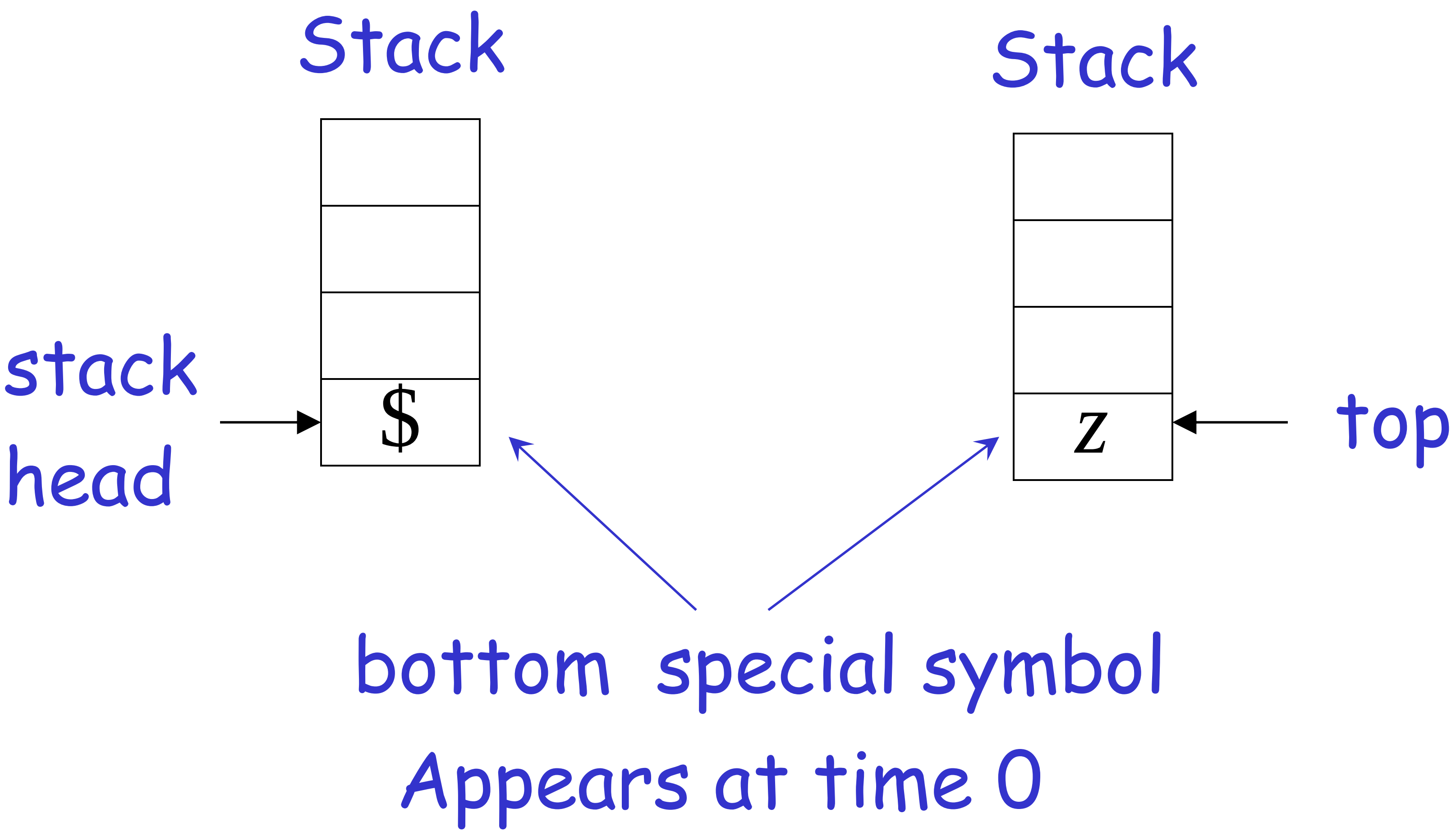
Part2:

https://www.youtube.com/watch?v=XlizUFH1M6w

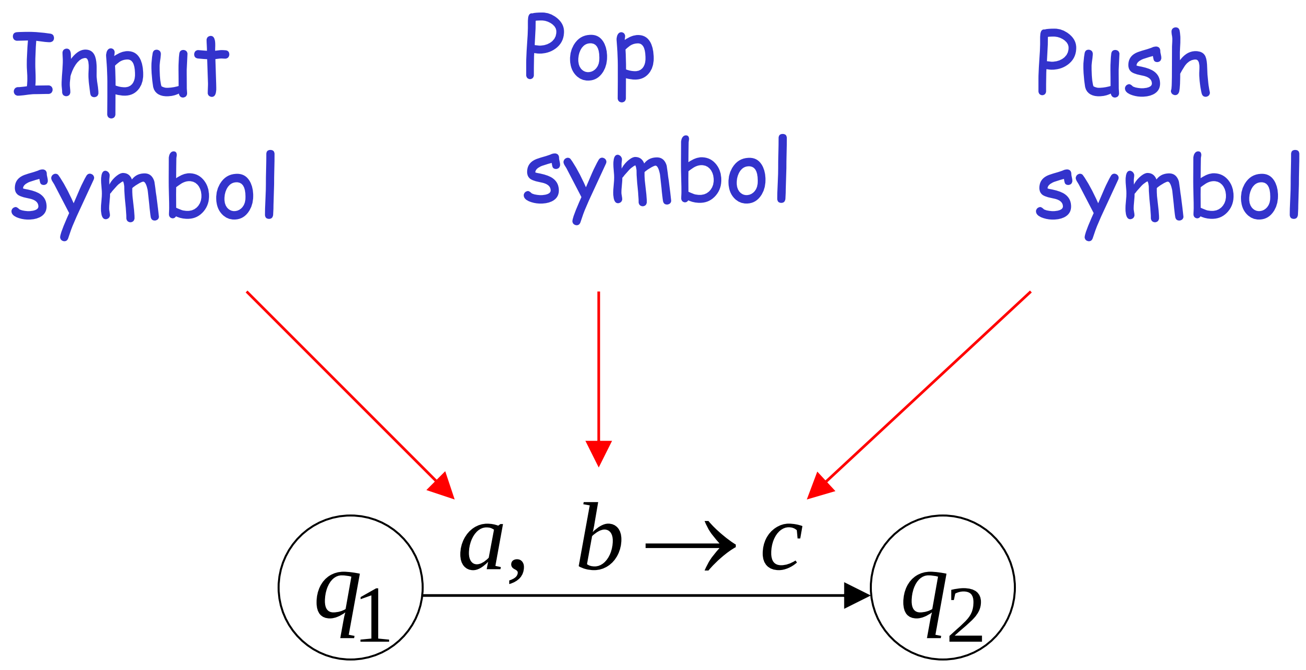
Pushdown Automaton -- PDA



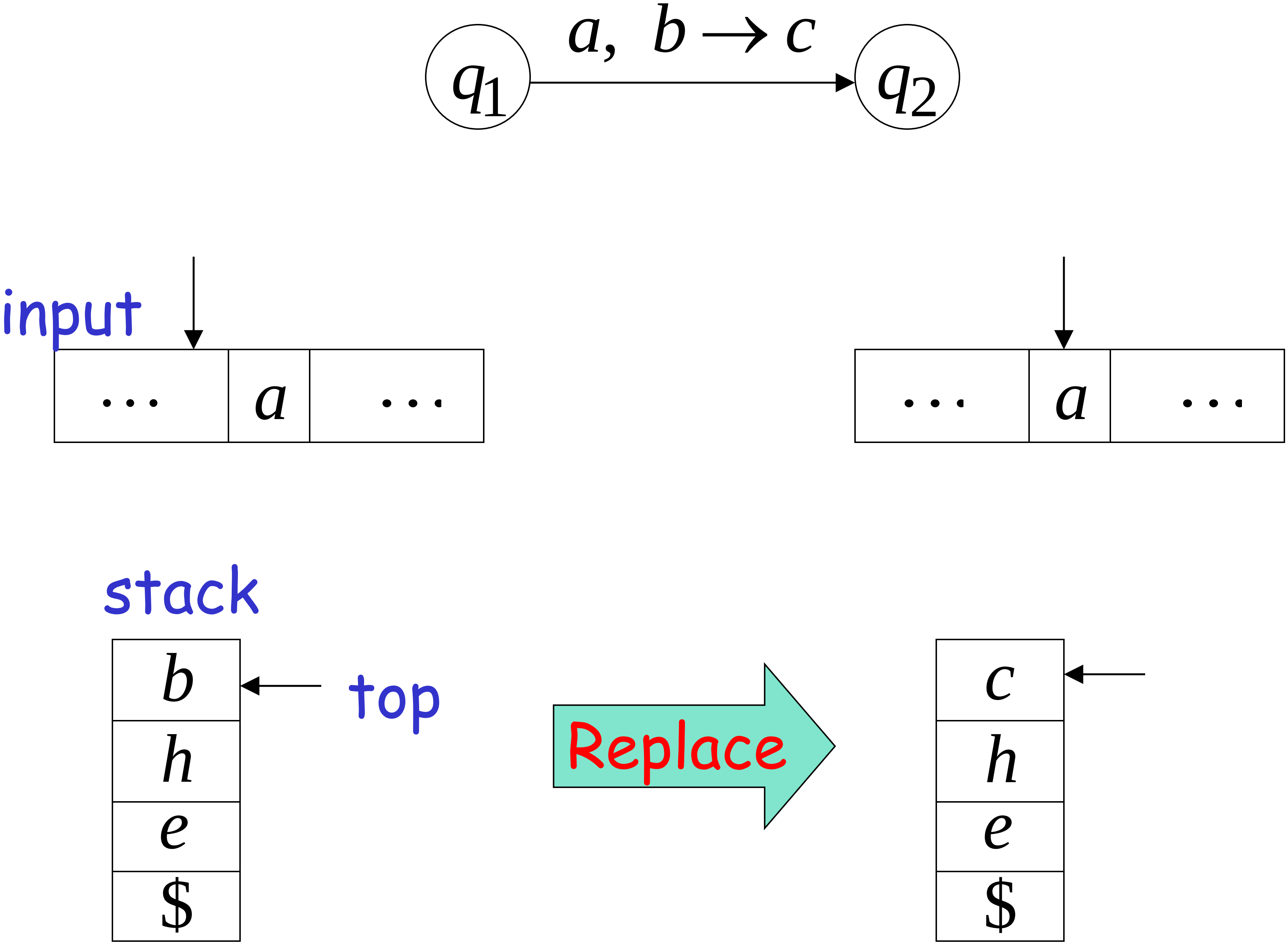
Initial Stack Symbol



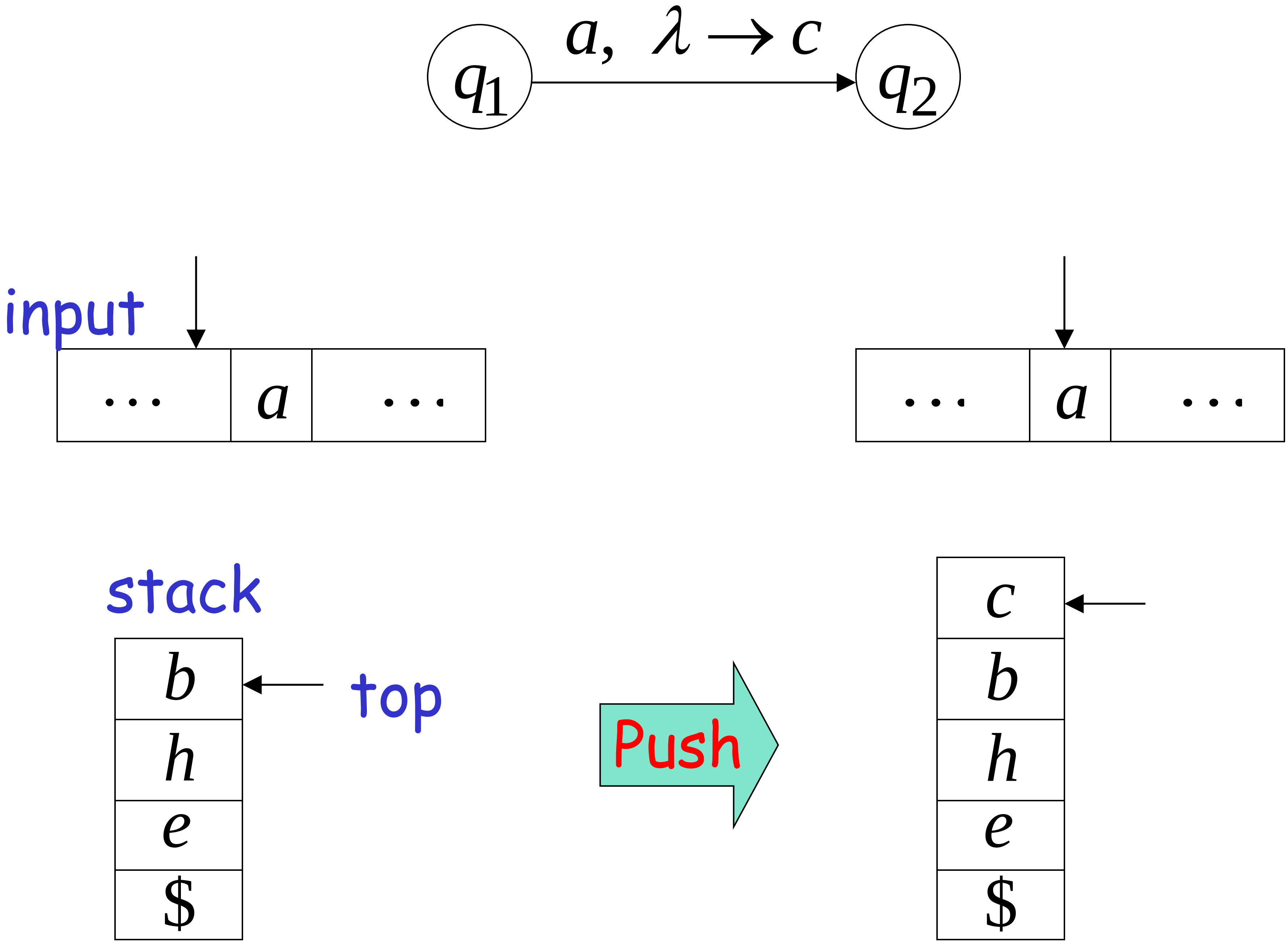
The States

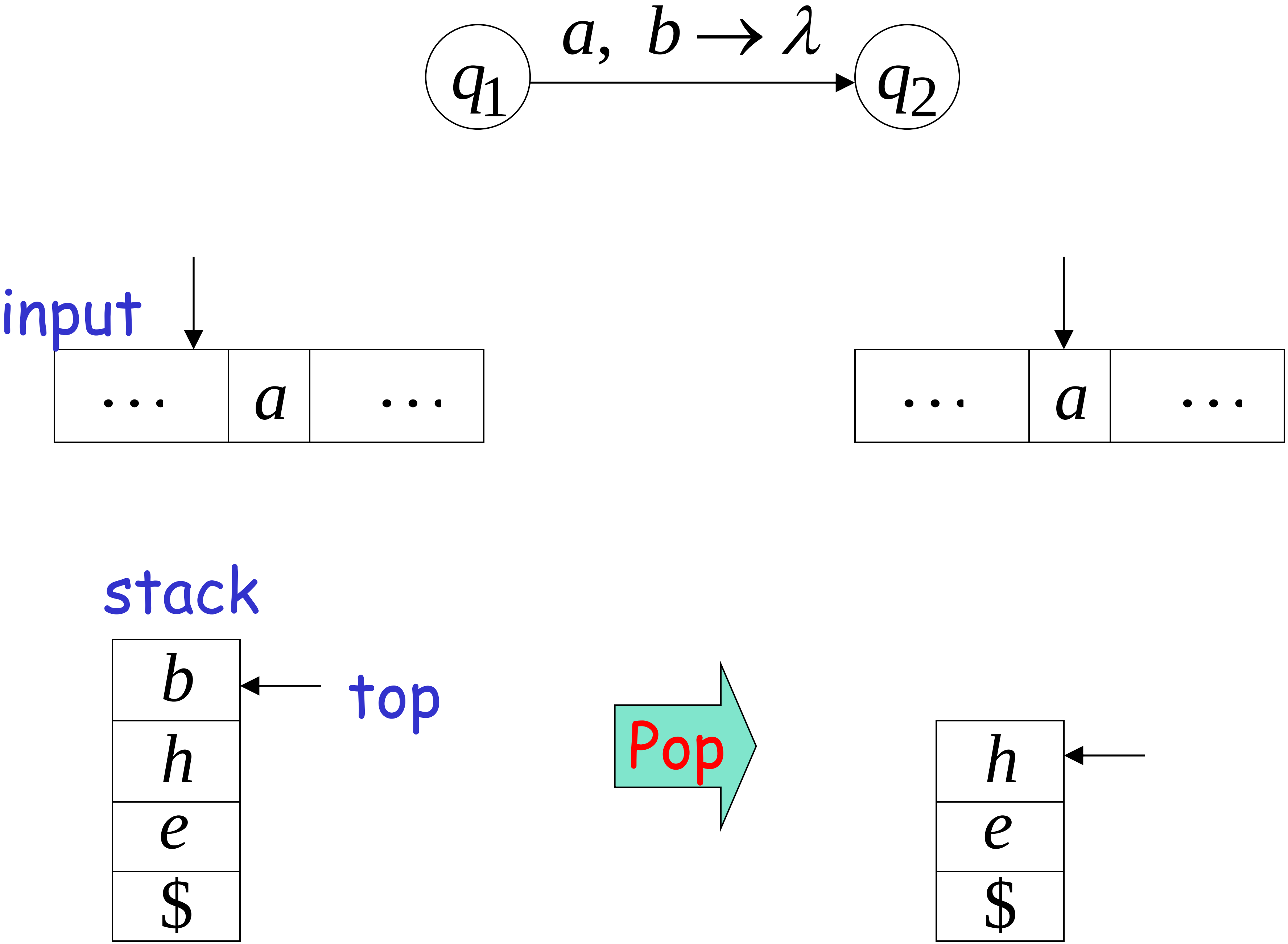


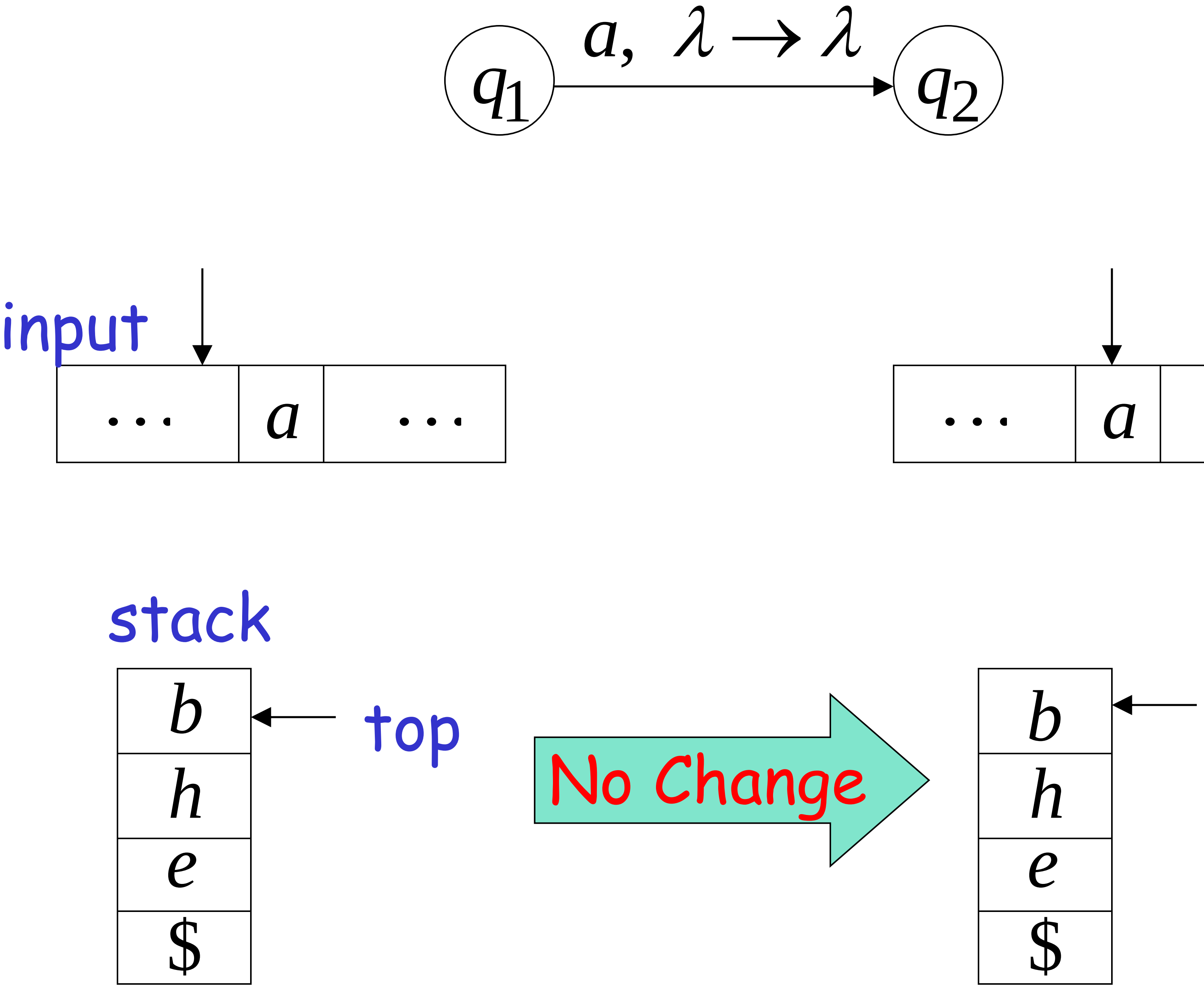
$$q_1 \xrightarrow{a,\ b \to c} q_2$$



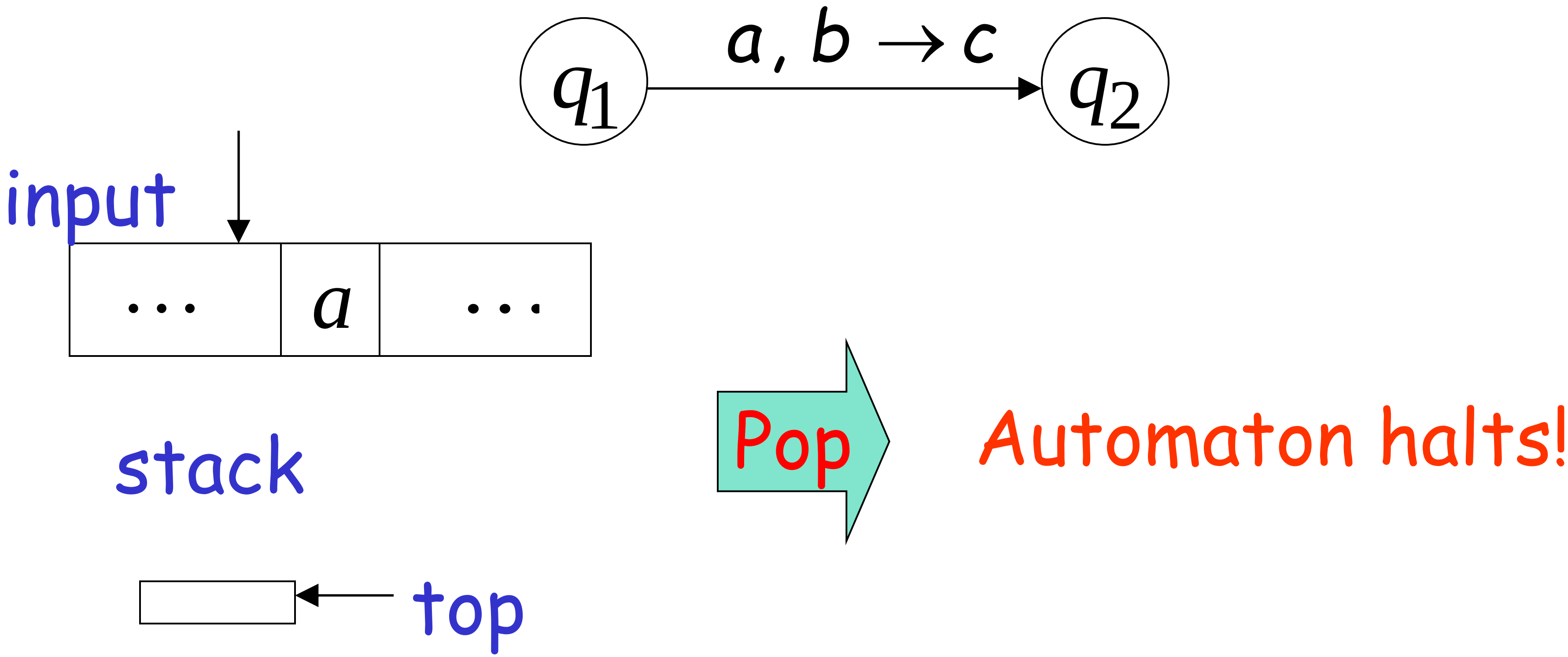
$a, \ b \to c$
q_1
q_2
input
$\cdots$
a
$\cdots$
$\cdots$
a
$\cdots$
stack
b
h
e
$\$$
top
Replace
c
h
e
$\$$


$a, \ \lambda \rightarrow c$
q_1
q_2
input
$\cdots$
a
$\cdots$
stack
top
b
h
e
$\$$
Push
c
b
h
e
$\$$


q_1
$a,\ b \to \lambda$
q_2
input
$\cdots$ a $\cdots$
$\cdots$ a $\cdots$
stack
b
h
e
$\$$
top
Pop
h
e
$\$$


$a, \lambda \to \lambda$
q_1
q_2
input
$\cdots$ a $\cdots$
$\cdots$ a $\cdots$
stack
top
No Change
b
h
e
$\$$
b
h
e
$\$$

Pop from Empty Stack

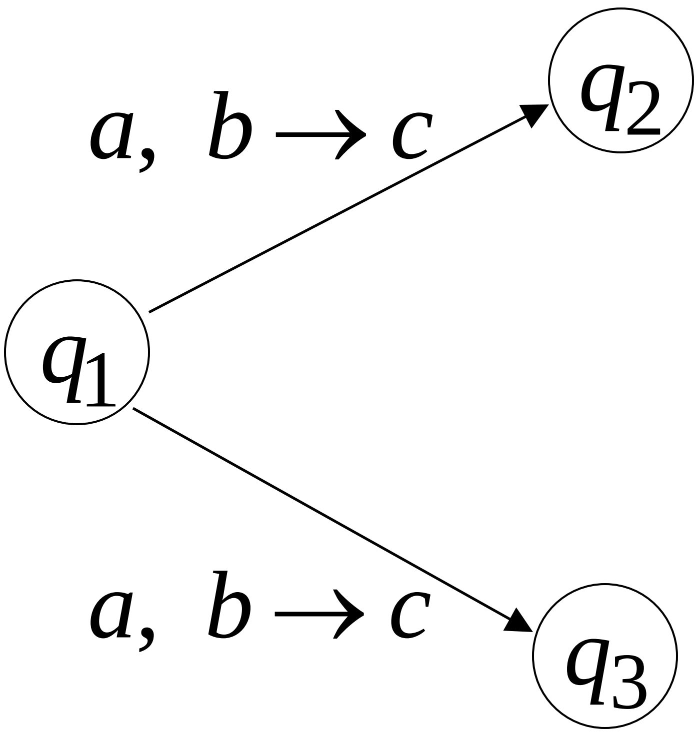


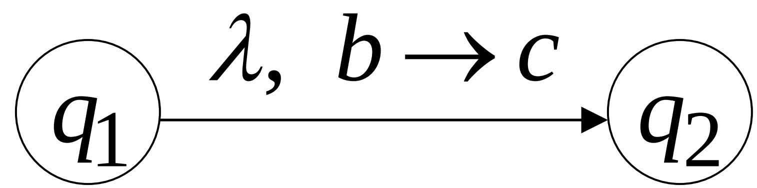
If the automaton attempts to pop from empty stack then it halts and rejects input

Non-Determinism

PDAs are non-deterministic

Allowed non-deterministic transitions



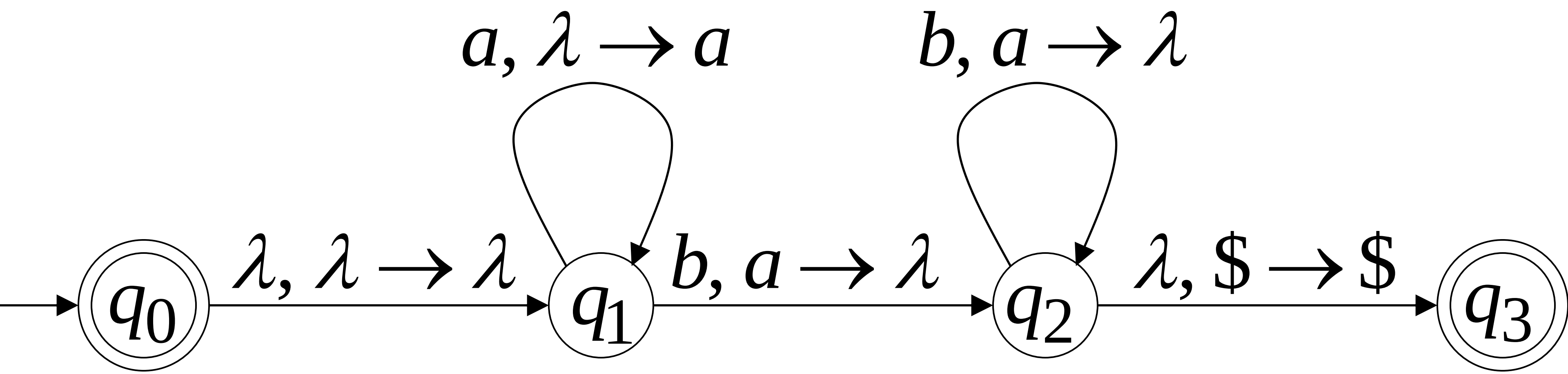


λ – transition

Example PDA

PDA M:

$$L(M) = \{a^n b^n : n \geq 0\}$$



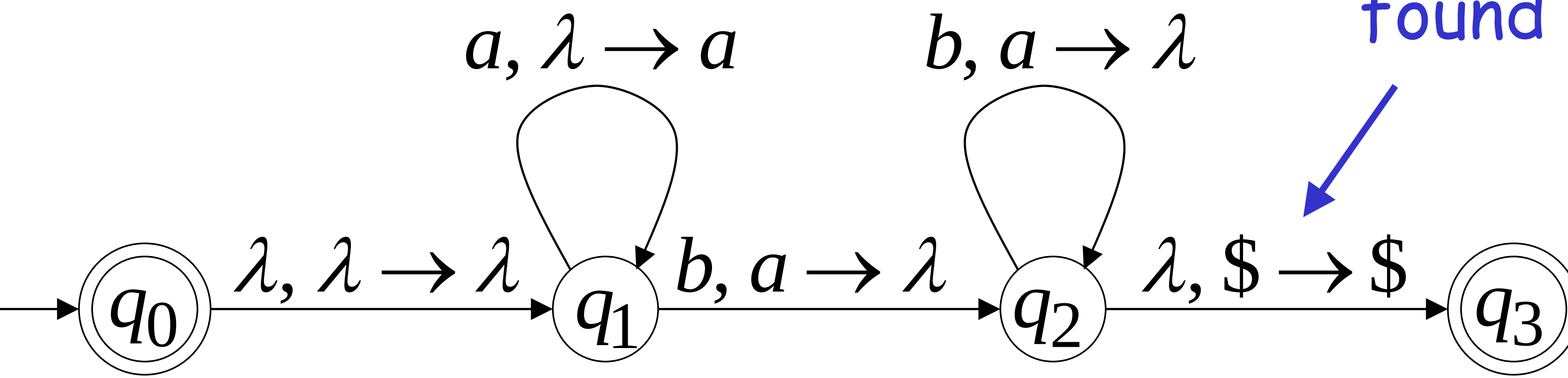
$$L(M) = \{a^n b^n : n \geq 0\}$$

Basic Idea:

1. Push the a's on the stack

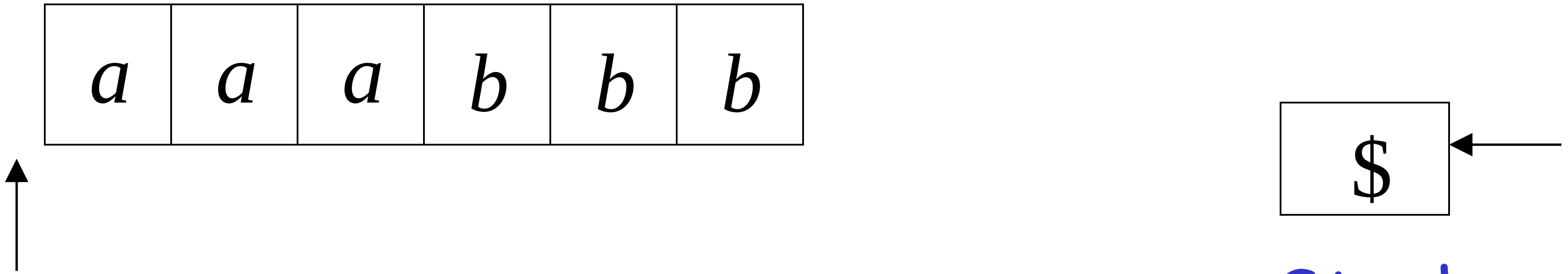
2. Match the b's on input with a's on stack

3. Match found



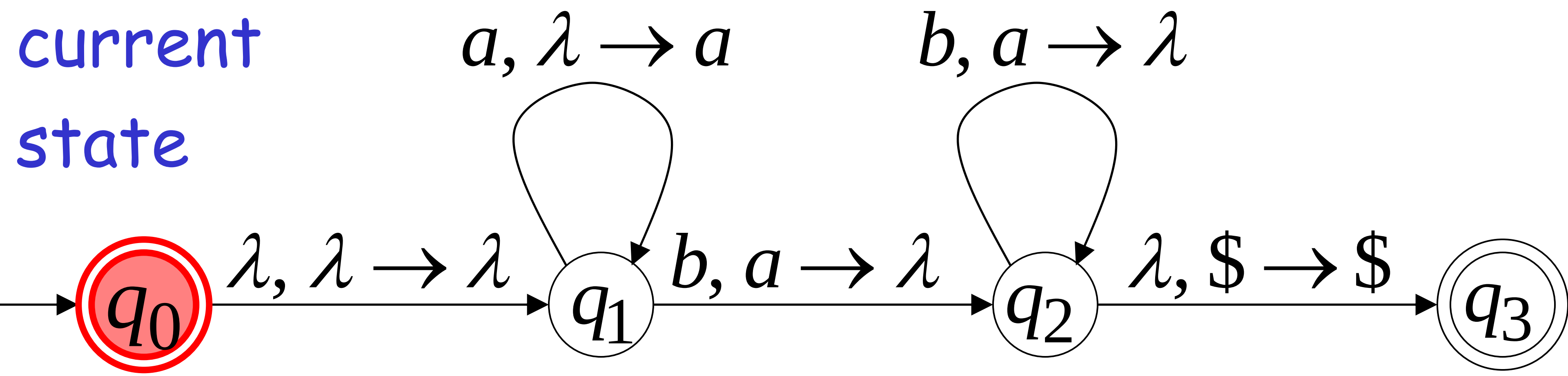
Execution Example: Time 0

Input



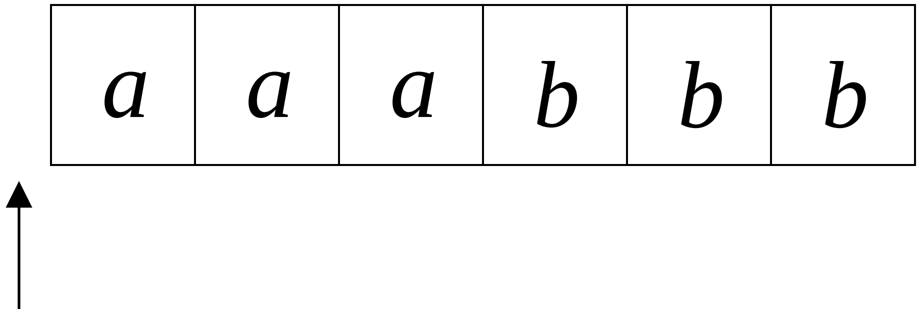
Stack

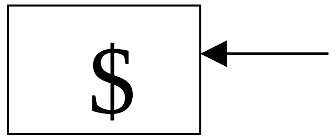
current state



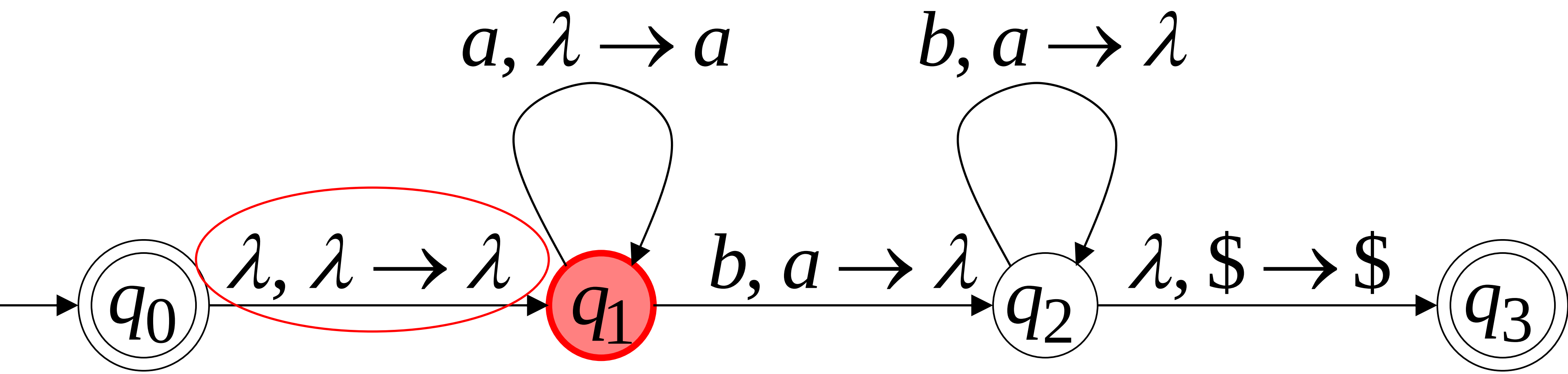
Time 1

Input



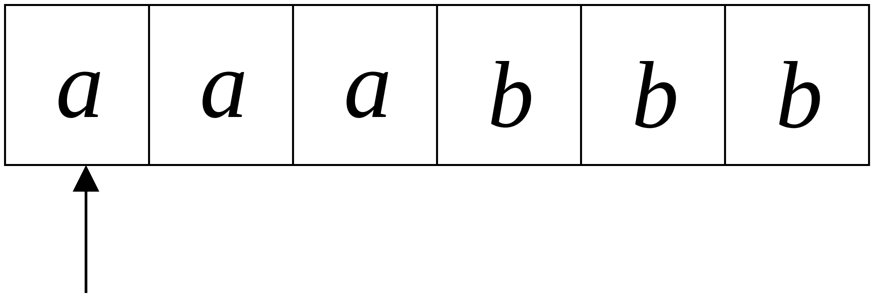


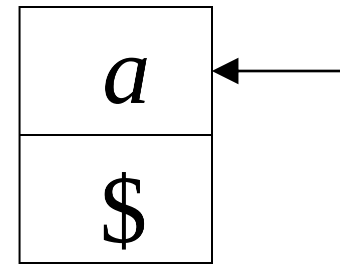
Stack



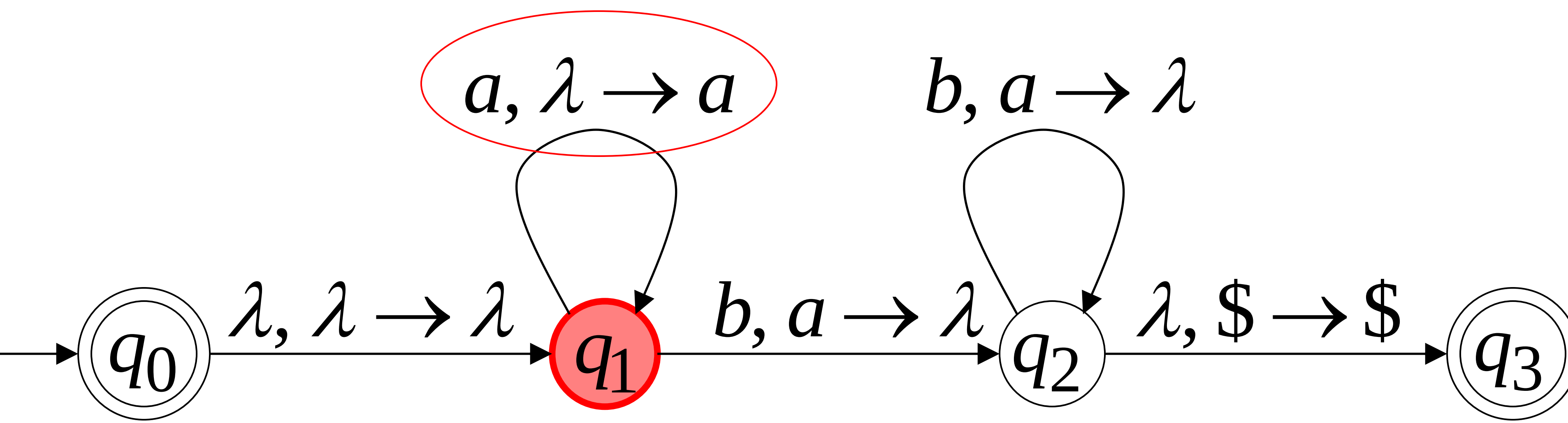
Time 2

Input

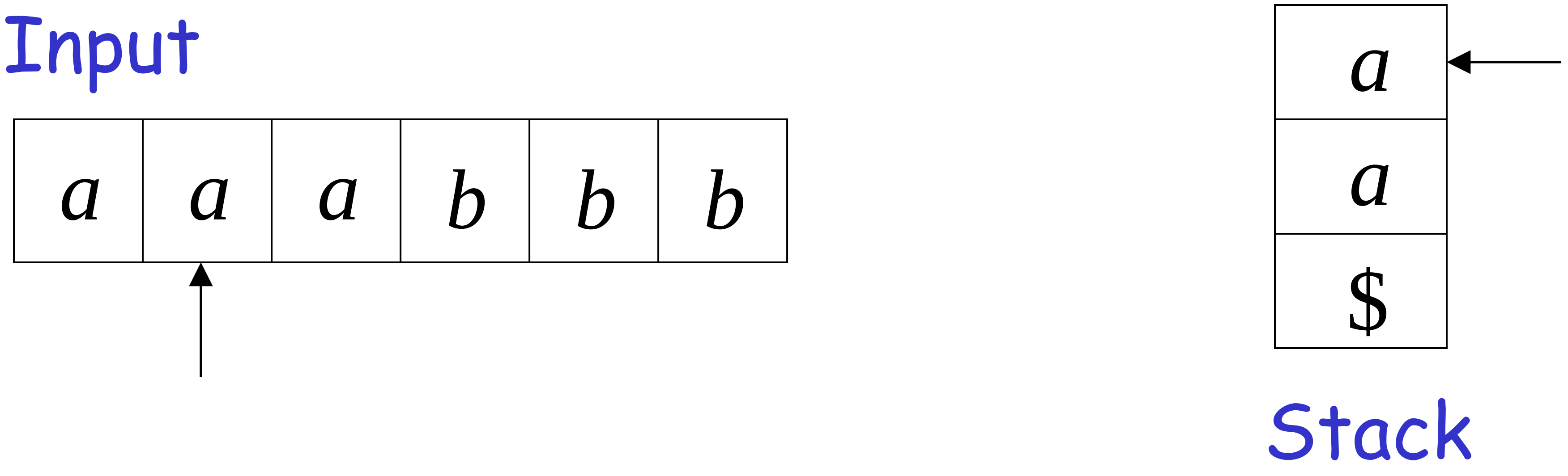


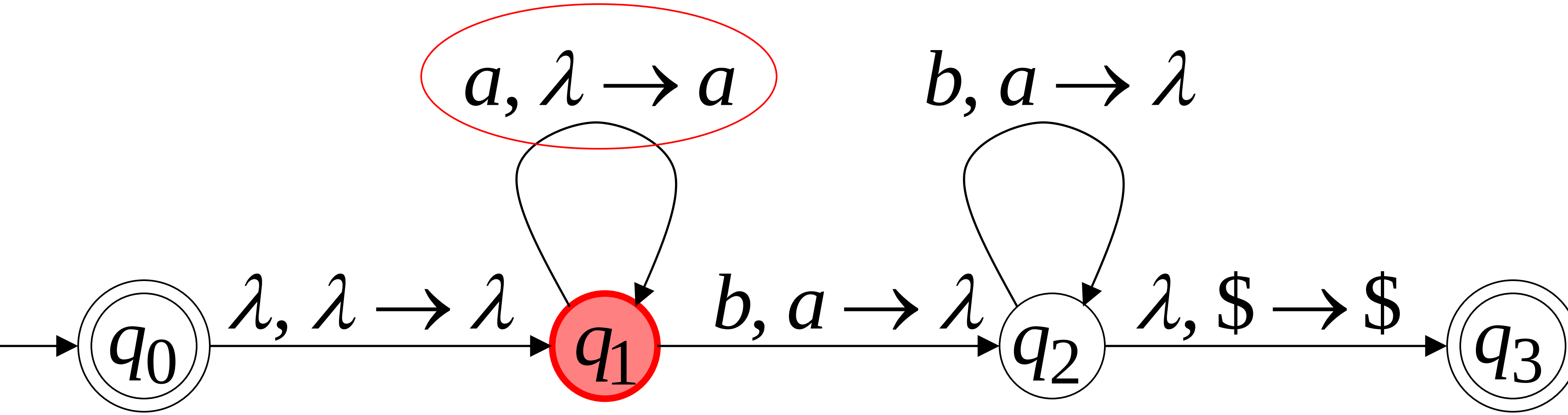


Stack



Time 3


Input
a a a b b b
a
a
$\$$
Stack


$a, \lambda \rightarrow a$
$b, a \rightarrow \lambda$
$\lambda, \lambda \rightarrow \lambda$
$b, a \rightarrow \lambda$
$\lambda, \$ \rightarrow \$$
q_0
q_1
q_2
q_3

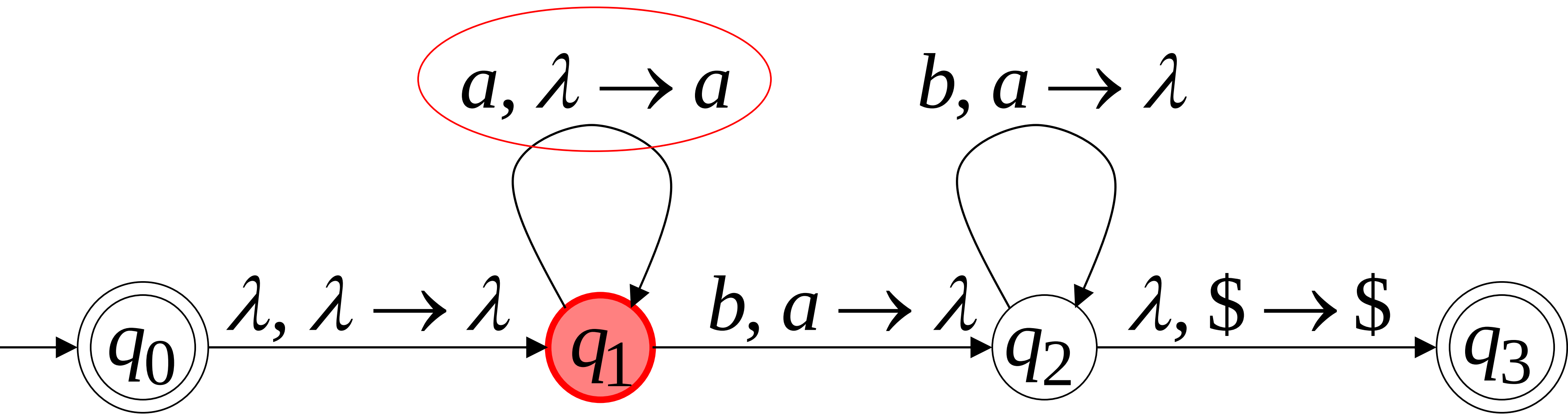
Time 4

Input

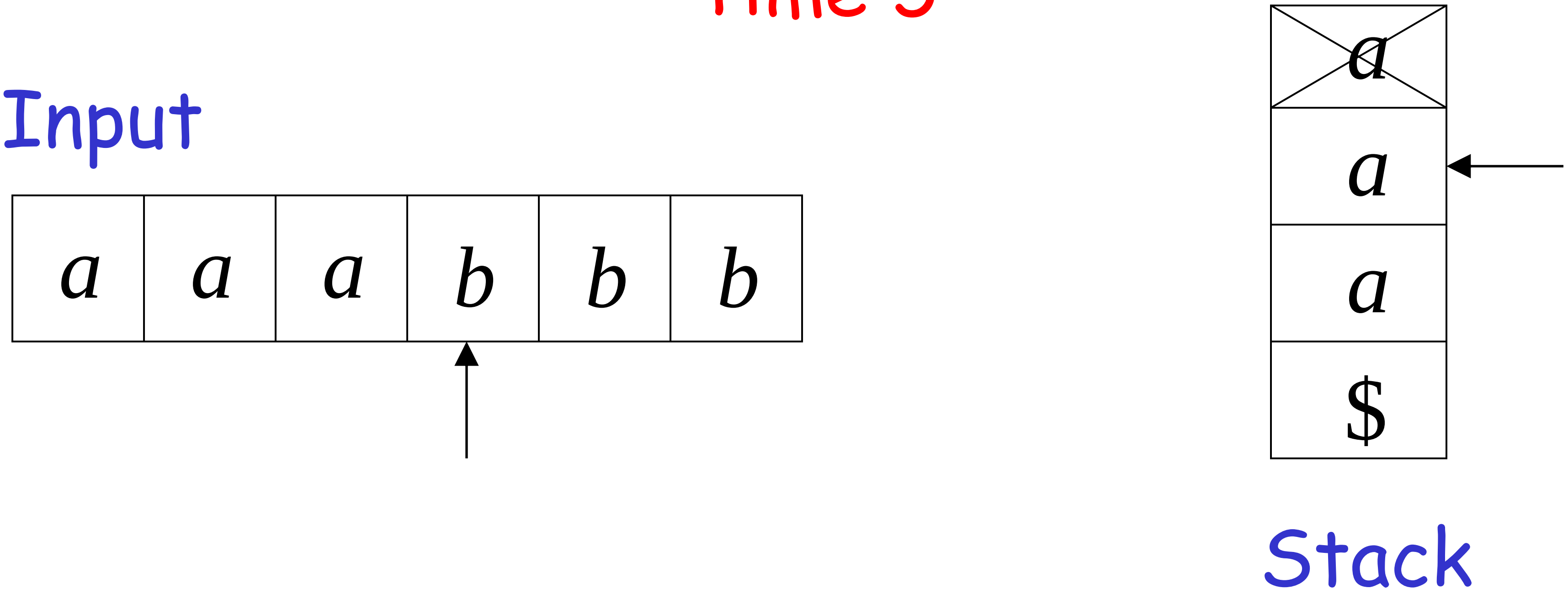
a	a	a	b	b	b

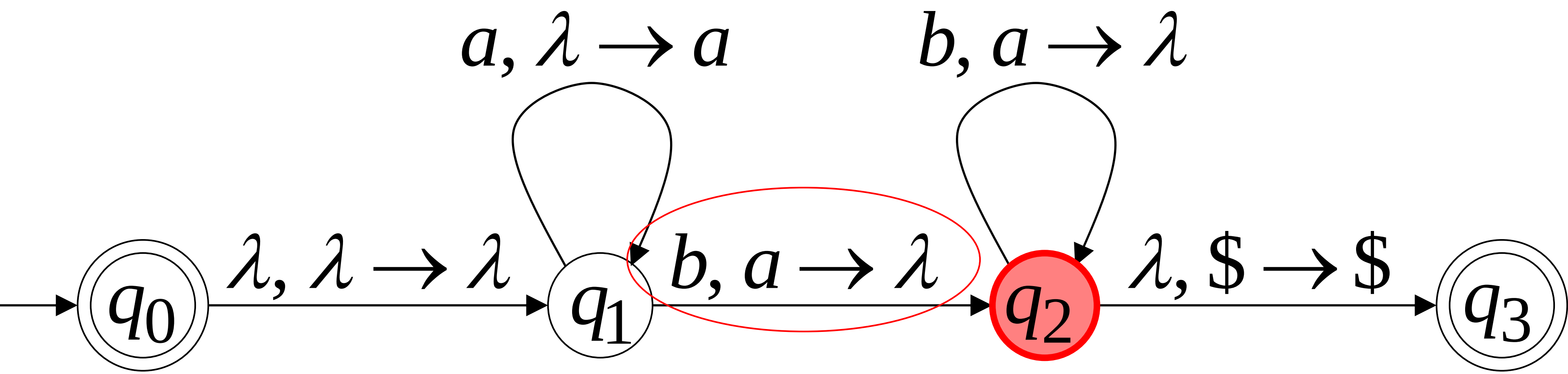
a
a
a
$\$$

Stack

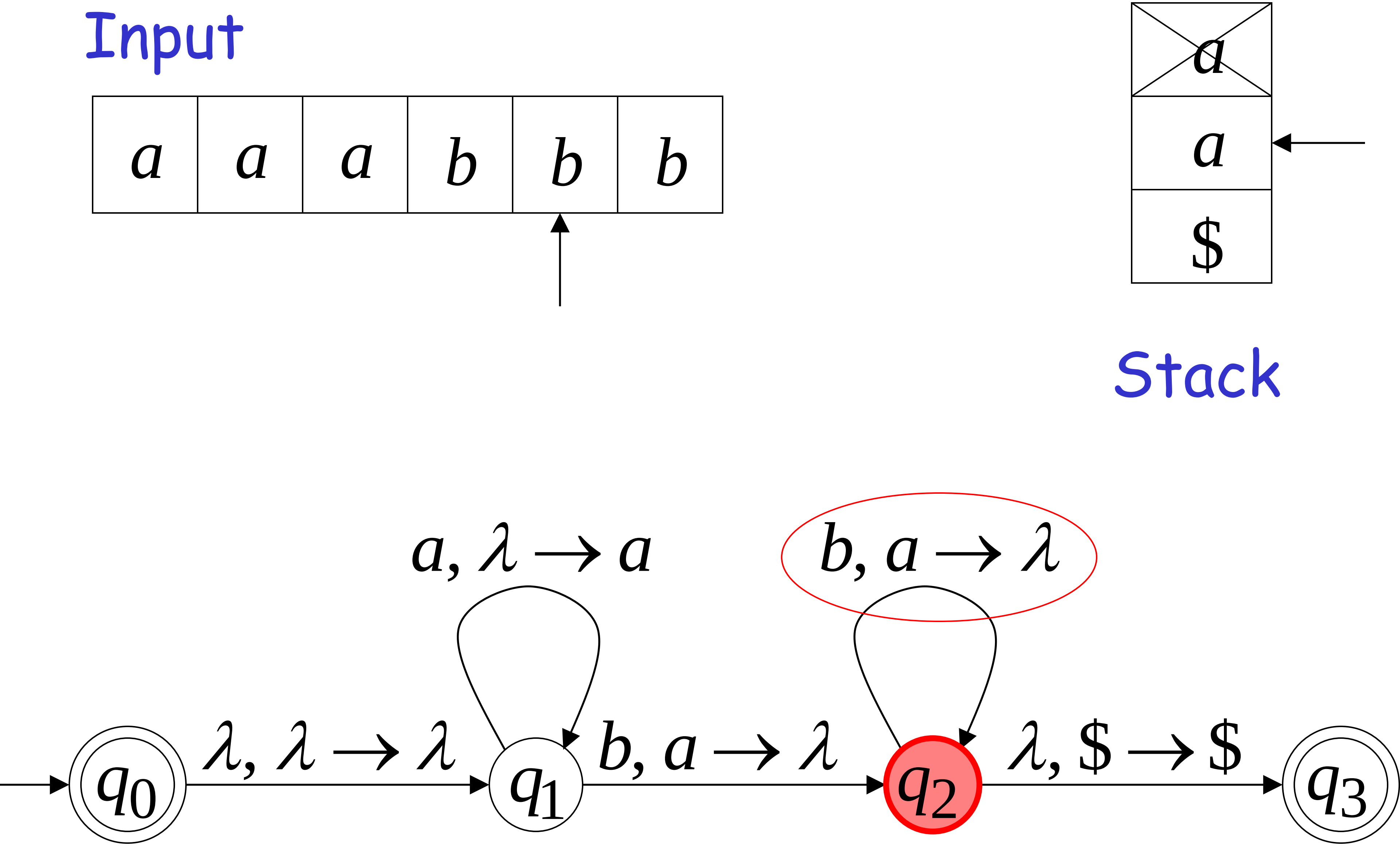


Time 5

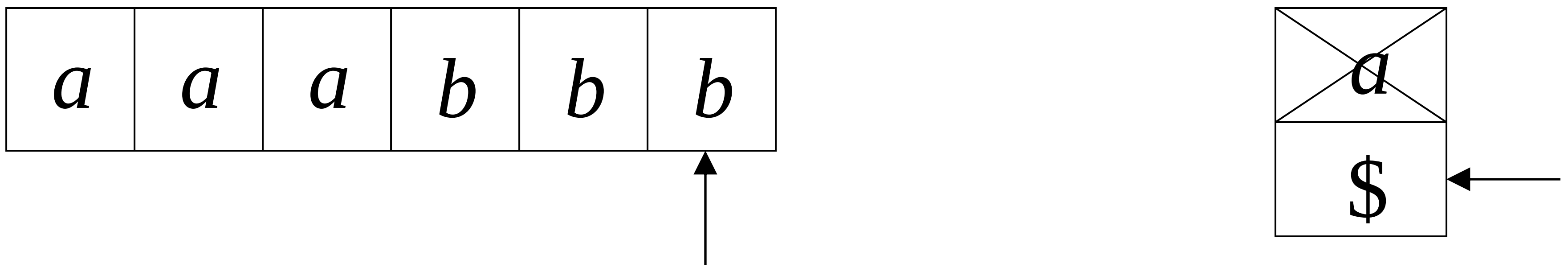

Input
a
a
a
b
b
b
a
a
a
$
Stack


$a, \lambda \to a$
$b, a \to \lambda$
$\lambda, \lambda \to \lambda$
$b, a \to \lambda$
$\lambda, \$ \to \$$
q_0
q_1
q_2
q_3

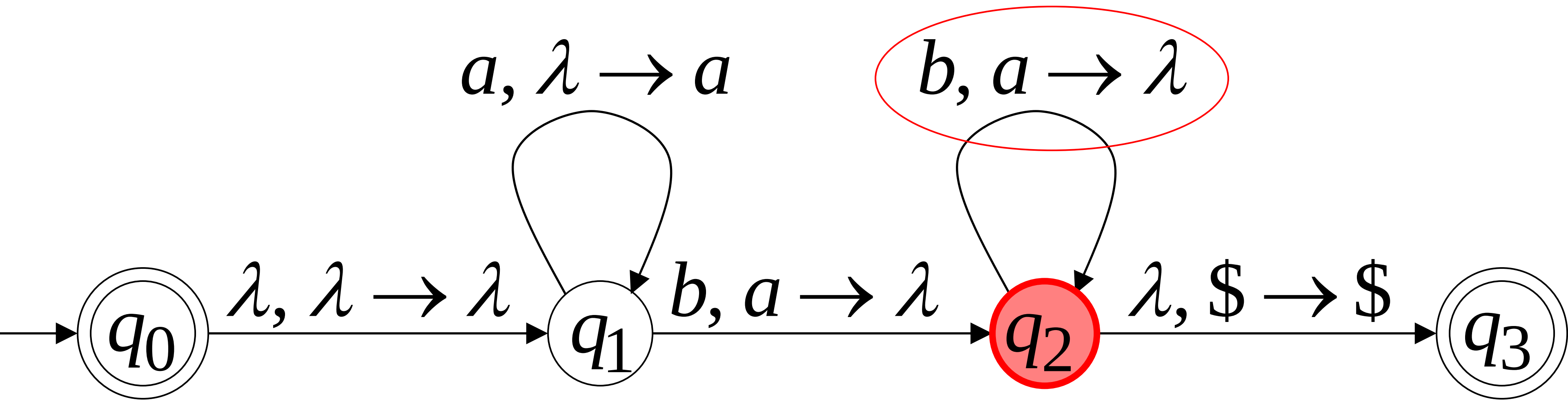
Time 6


Input
a a a b b b
$\cancel{a}$
a
$\$$
Stack
$a, \lambda \to a$
$b, a \to \lambda$
$\lambda, \lambda \to \lambda$
$b, a \to \lambda$
$\lambda, \$ \to \$$
q_0
q_1
q_2
q_3

Time 7

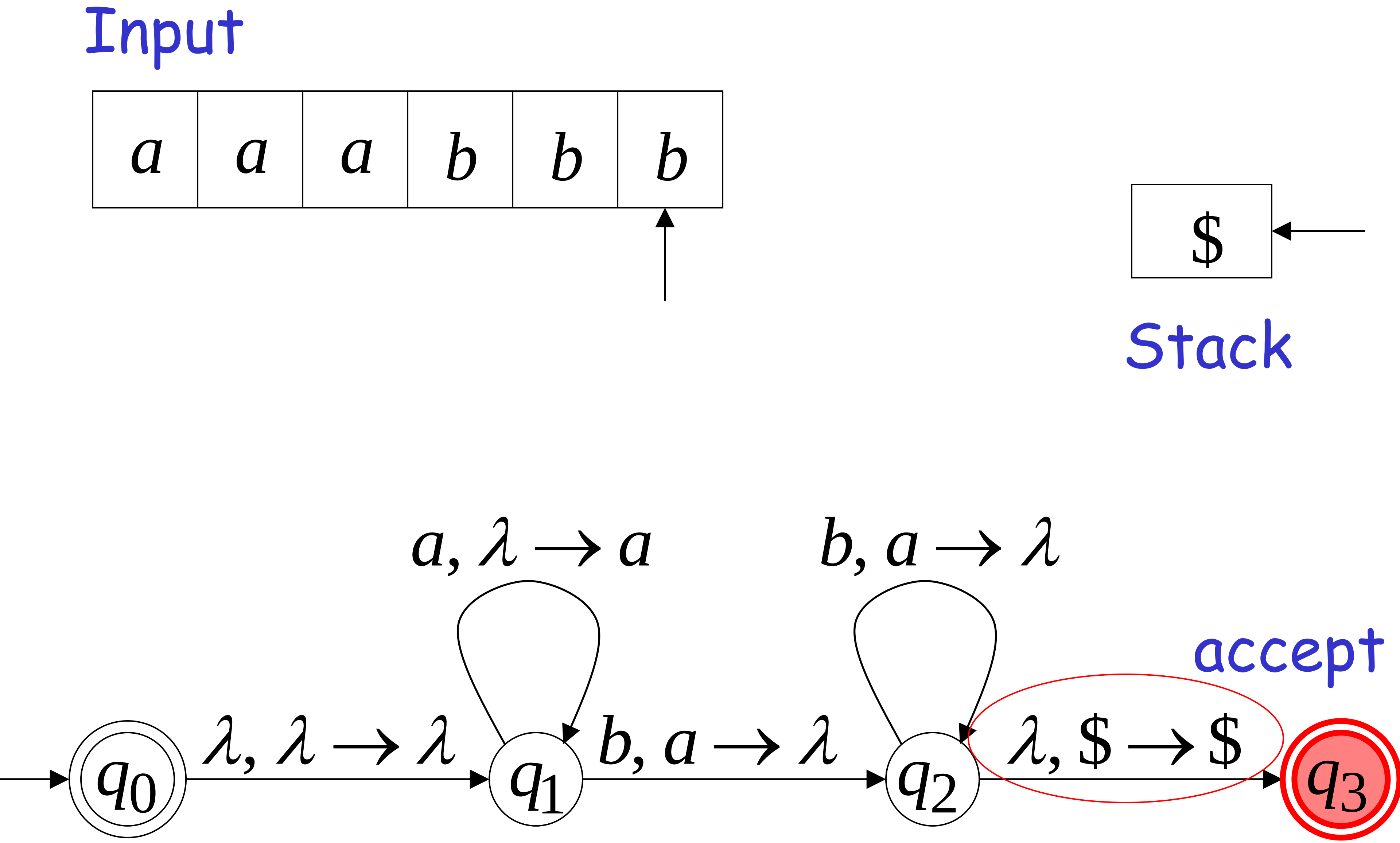


Input
a
a
a
b
b
b
$\cancel{a}$
$\$$
Stack



$a, \lambda \to a$
$b, a \to \lambda$
$\lambda, \lambda \to \lambda$
$b, a \to \lambda$
$\lambda, \$ \to \$$
q_0
q_1
q_2
q_3

Time 8



Input
a a a b b b
$\$$
Stack
$a, \lambda \to a$
$b, a \to \lambda$
accept
$\lambda, \lambda \to \lambda$
$b, a \to \lambda$
$\lambda, \$ \to \$$
q_0
q_1
q_2
q_3

A string is accepted if there is
a computation such that:

All the input is consumed

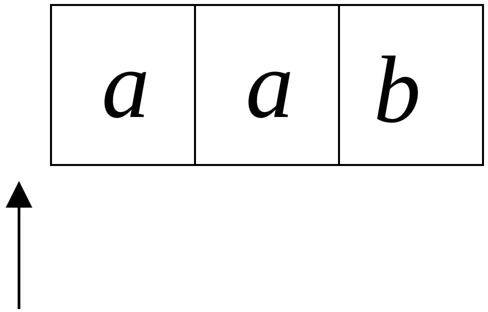
AND

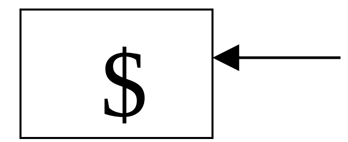
The last state is an accepting state

we do not care about the stack contents
at the end of the accepting computation

Rejection Example: Time 0

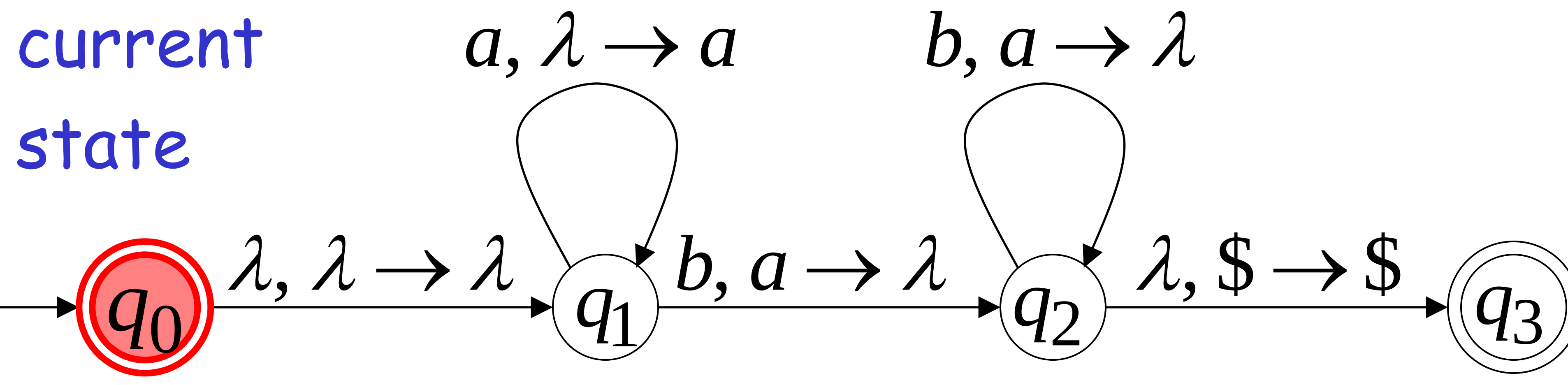
Input





Stack

current state



Rejection Example: Time 1

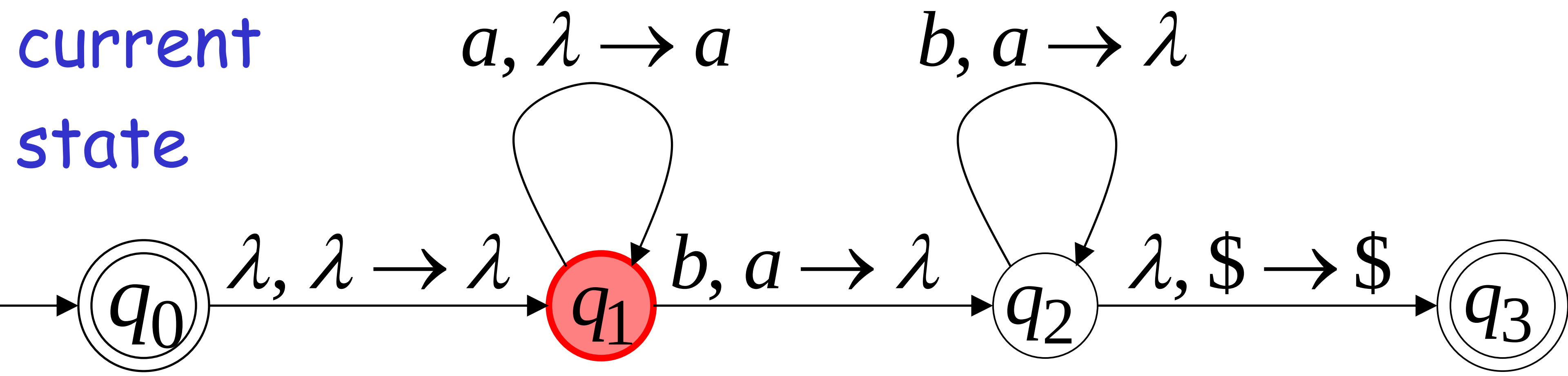
Input

a	a	b

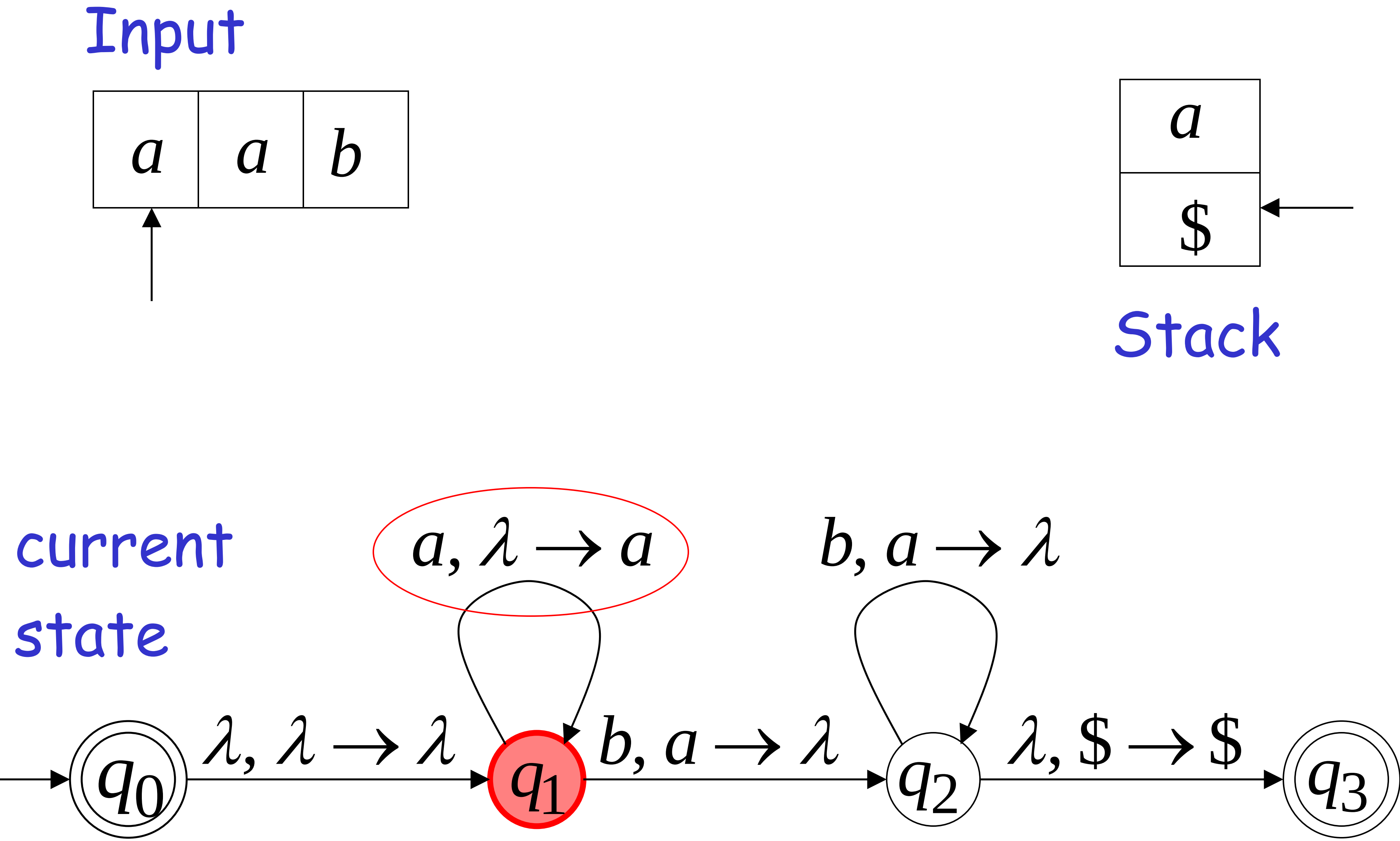
$\$$

Stack

current state



Rejection Example: Time 2



Rejection Example: Time 3

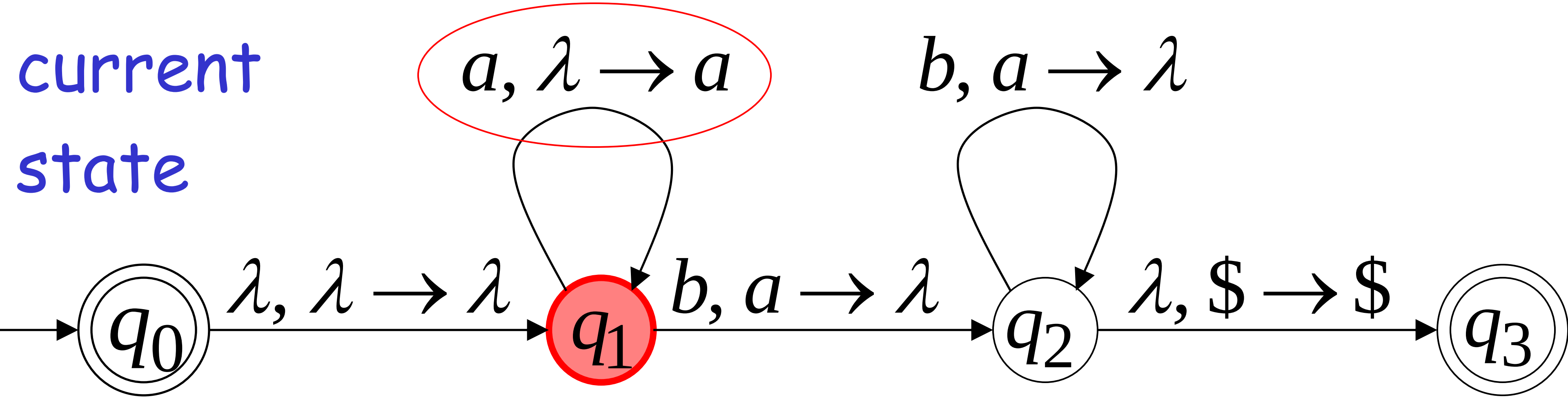
Input

a	a	b

a
a
$\$$

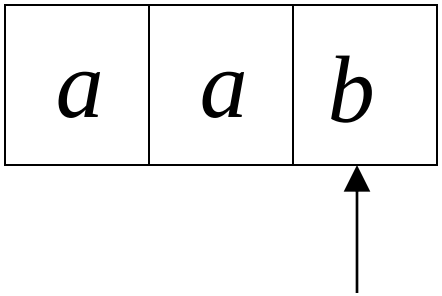
Stack

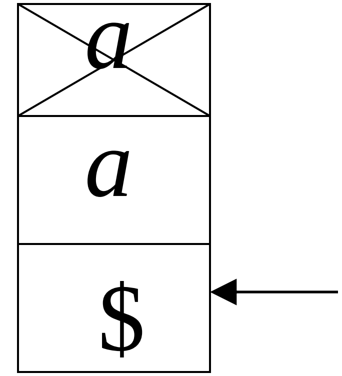
current state



Rejection Example: Time 4

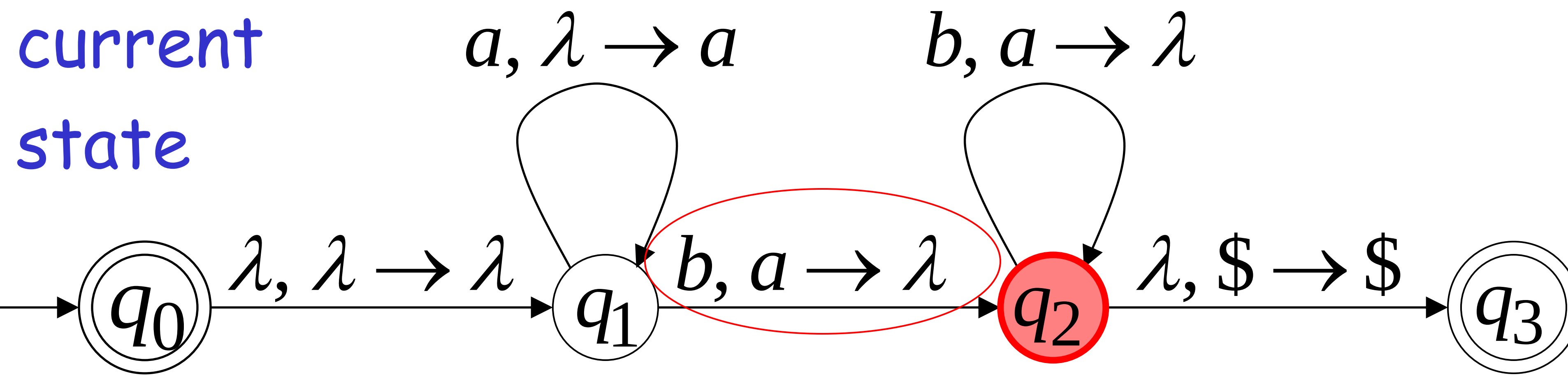
Input





Stack

current state



Rejection Example: Time 4

Input

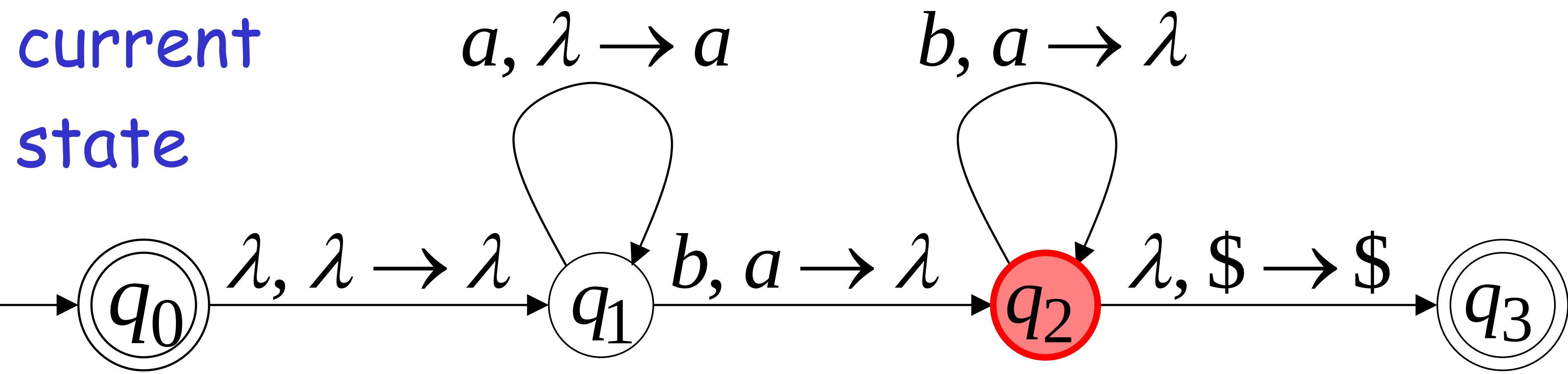
a	a	b

$\cancel{a}$
a
$\$$

Stack

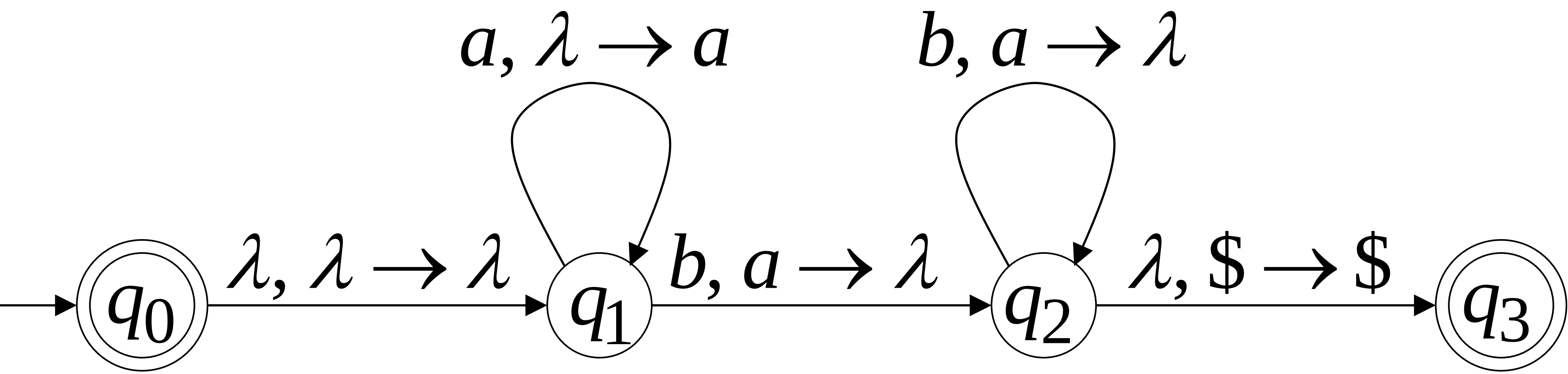
reject

current state



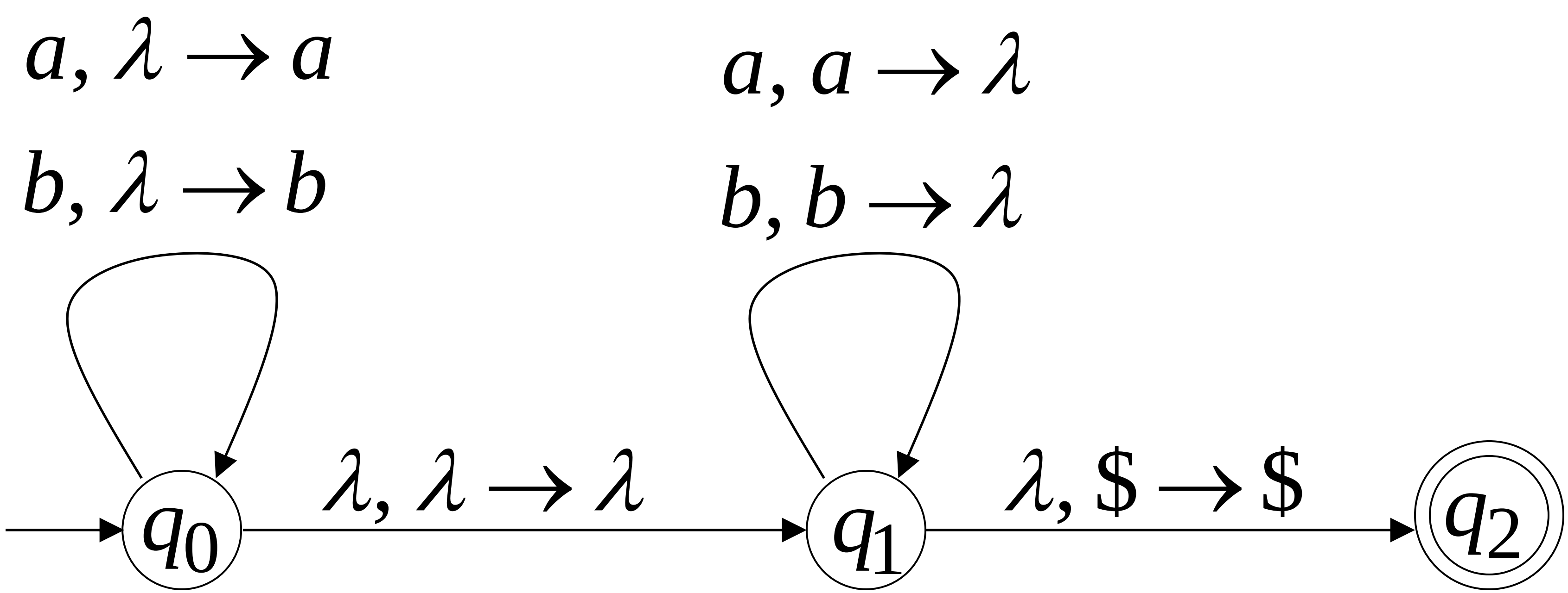
There is no accepting computation for aab

The string aab is rejected by the PDA

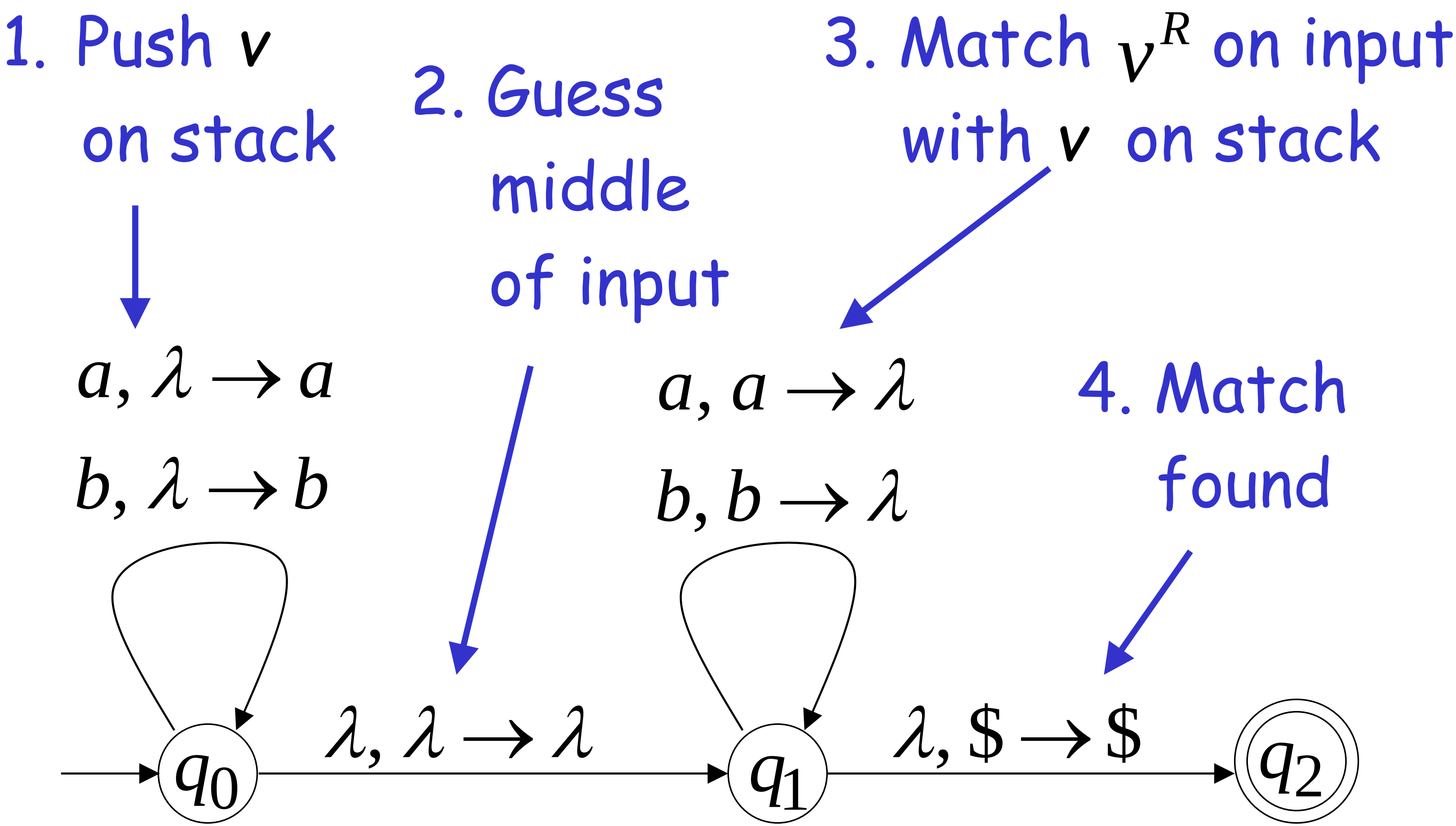


Another PDA example

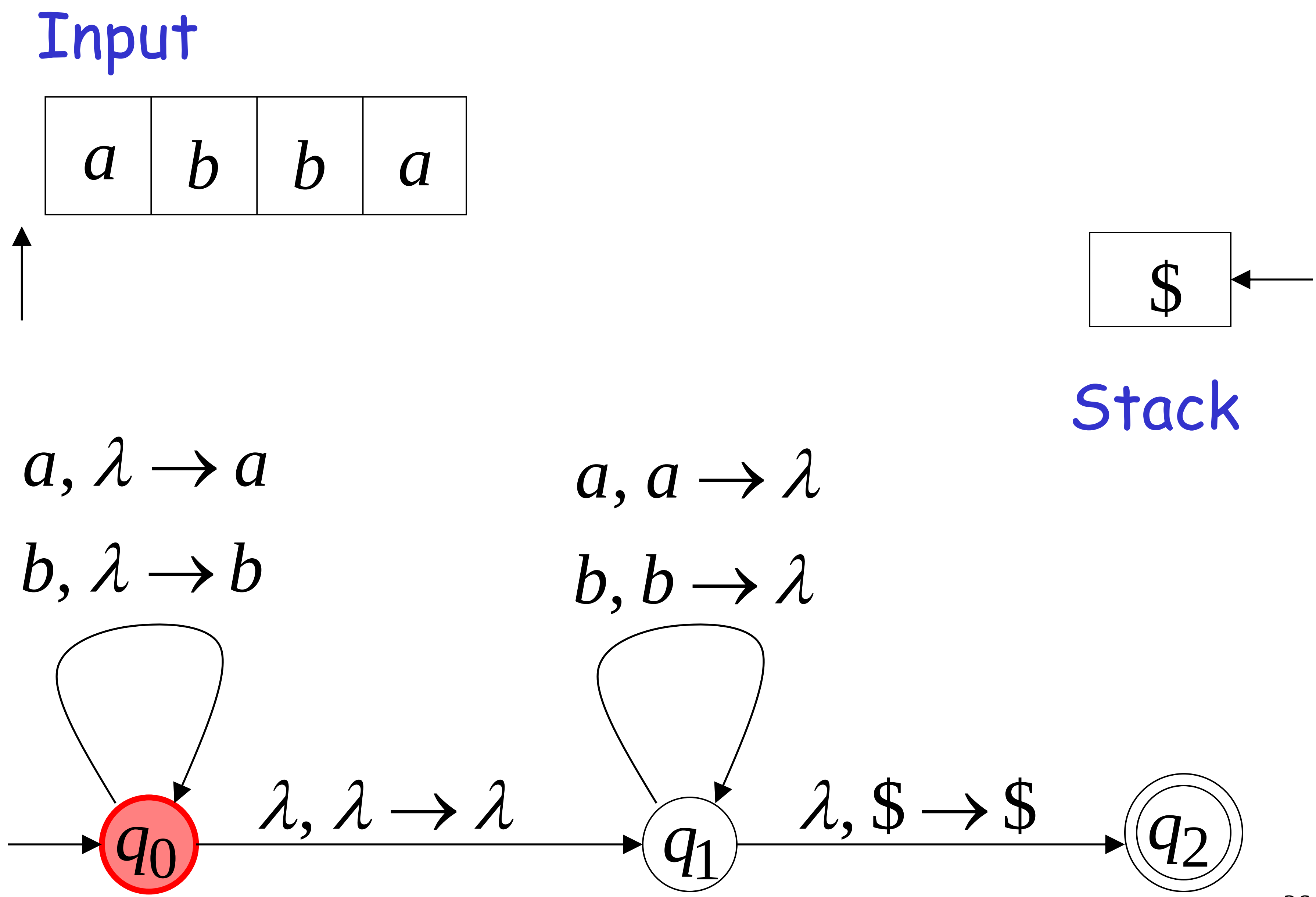
PDA M: $\quad L(M) = \{vv^R : v \in \{a,b\}^*\}$



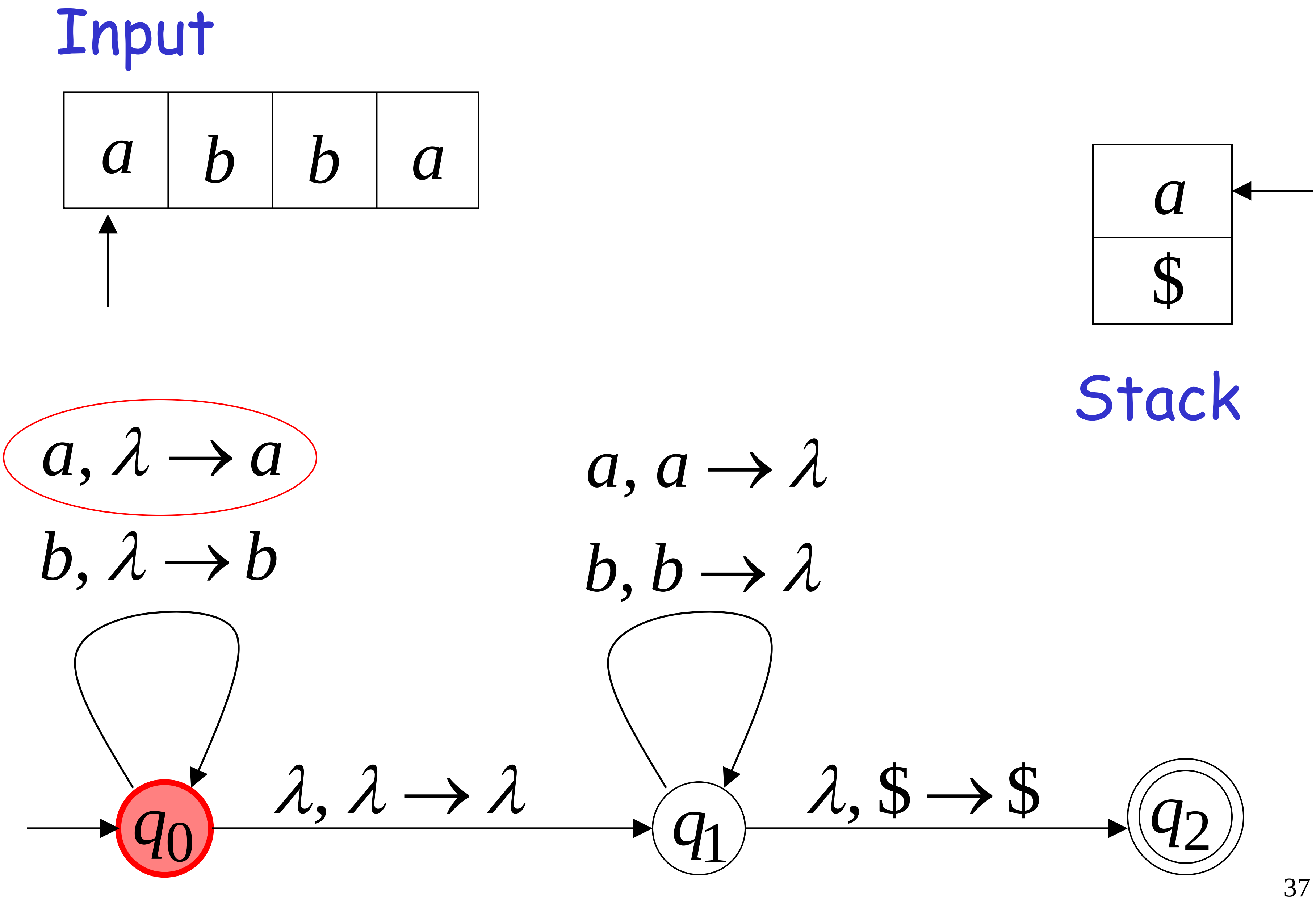
Basic Idea: $L(M) = \{vv^R : v \in \{a,b\}^*\}$



Execution Example: Time 0

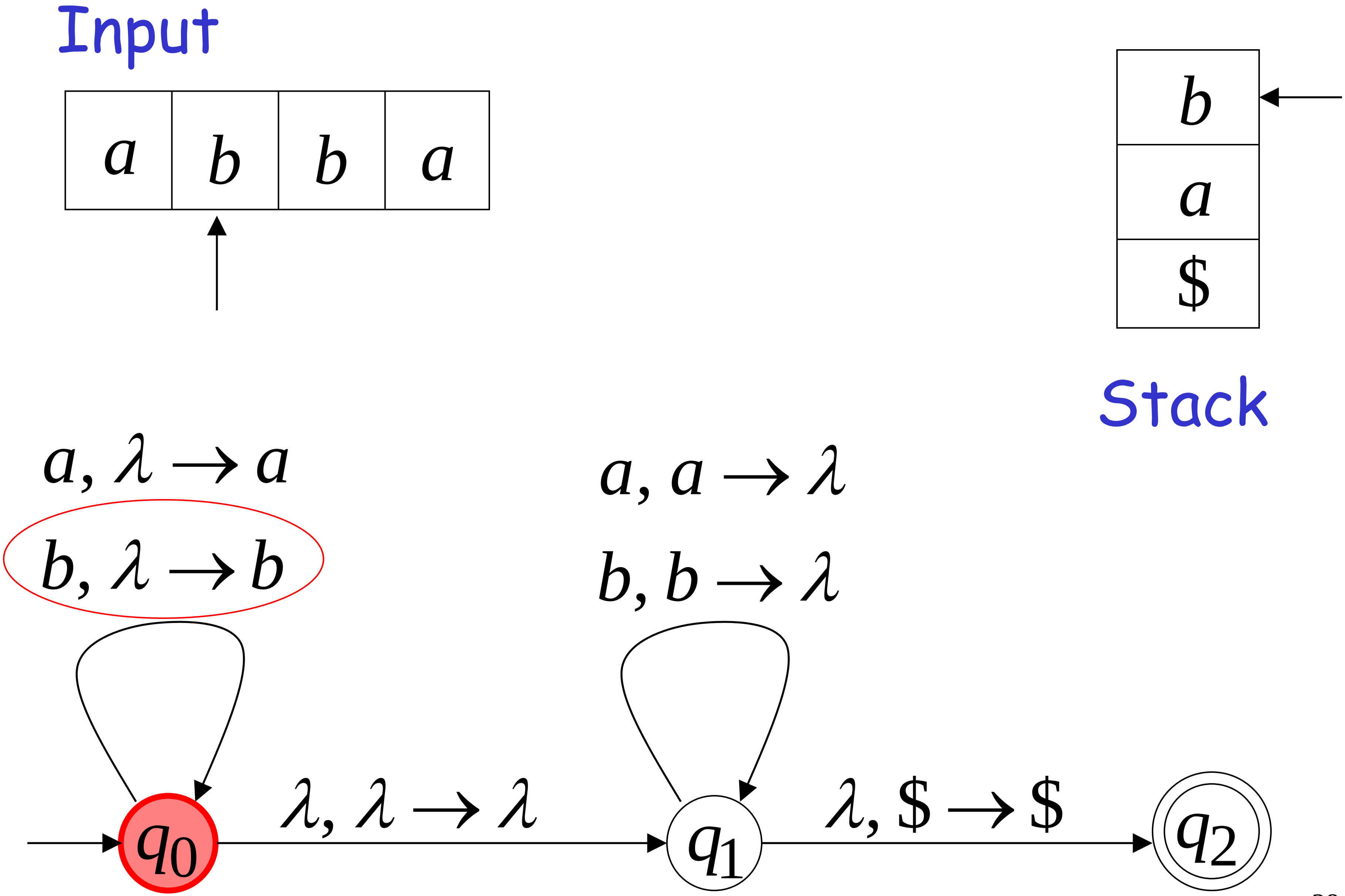


Time 1



Input
a b b a
a
$
Stack
$a, \lambda \rightarrow a$
$b, \lambda \rightarrow b$
$a, a \rightarrow \lambda$
$b, b \rightarrow \lambda$
$\lambda, \lambda \rightarrow \lambda$
$\lambda, \$ \rightarrow \$$
q_0
q_1
q_2

Time 2



Time 3

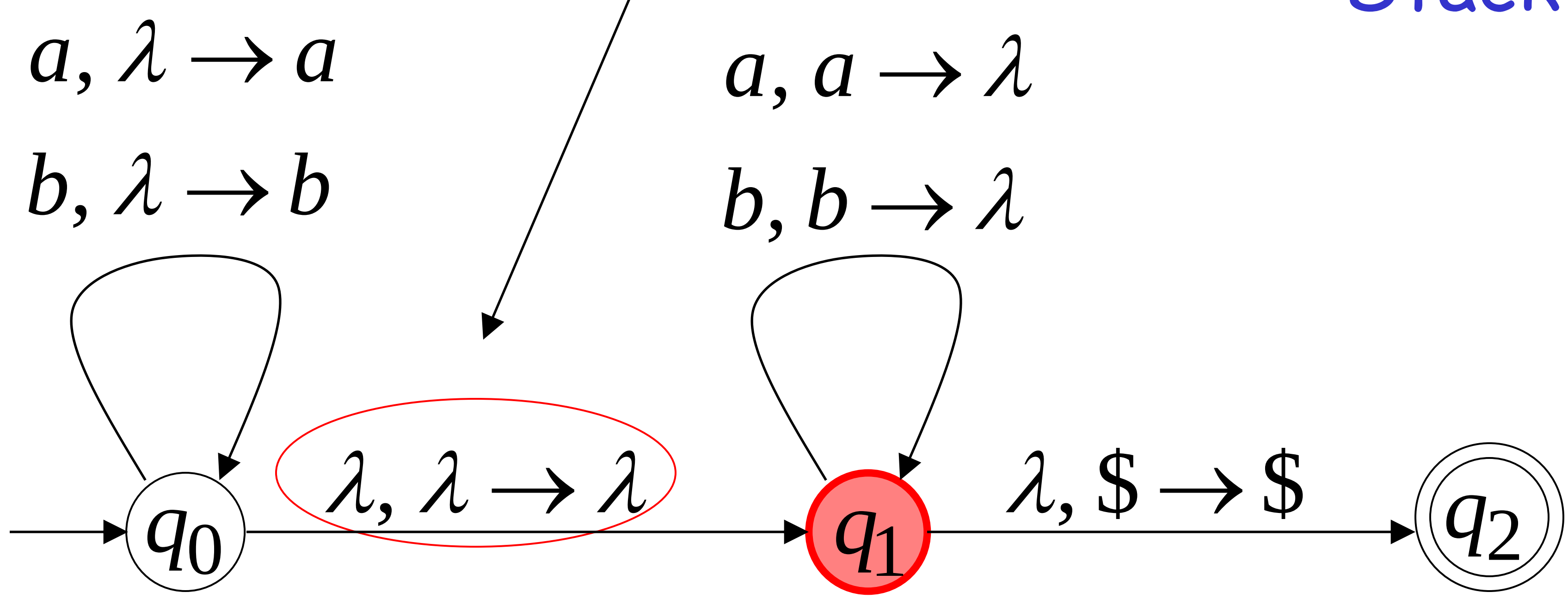
Input

a	b	b	a

Guess the middle of string

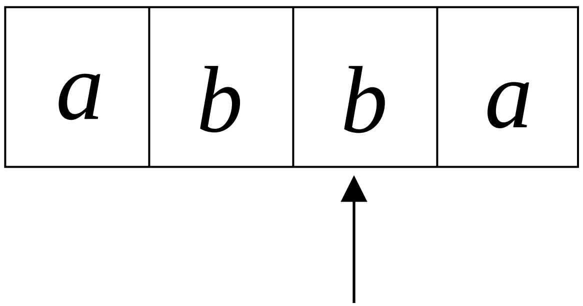
b
a
$\$$

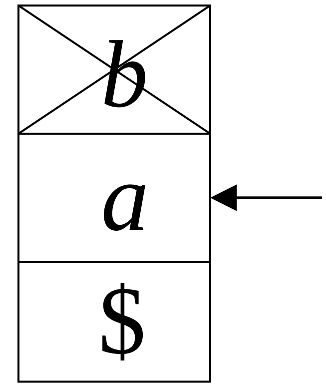
Stack



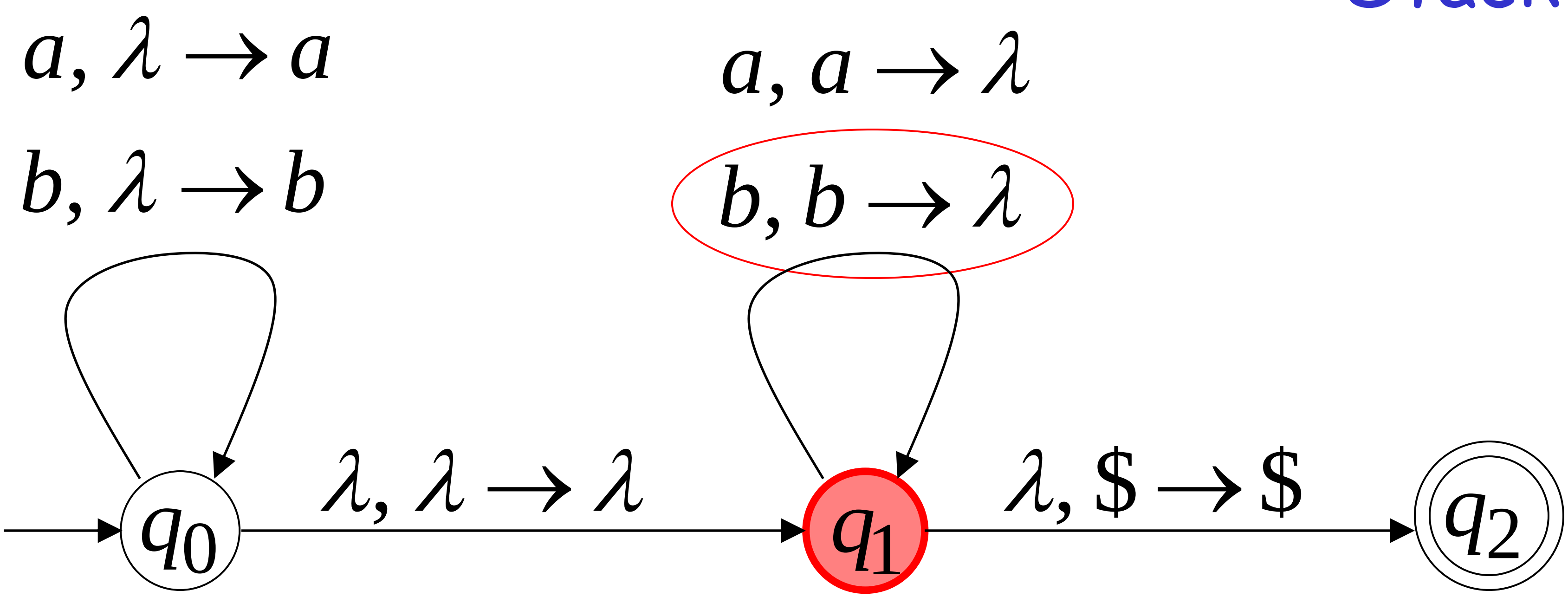
Time 4

Input

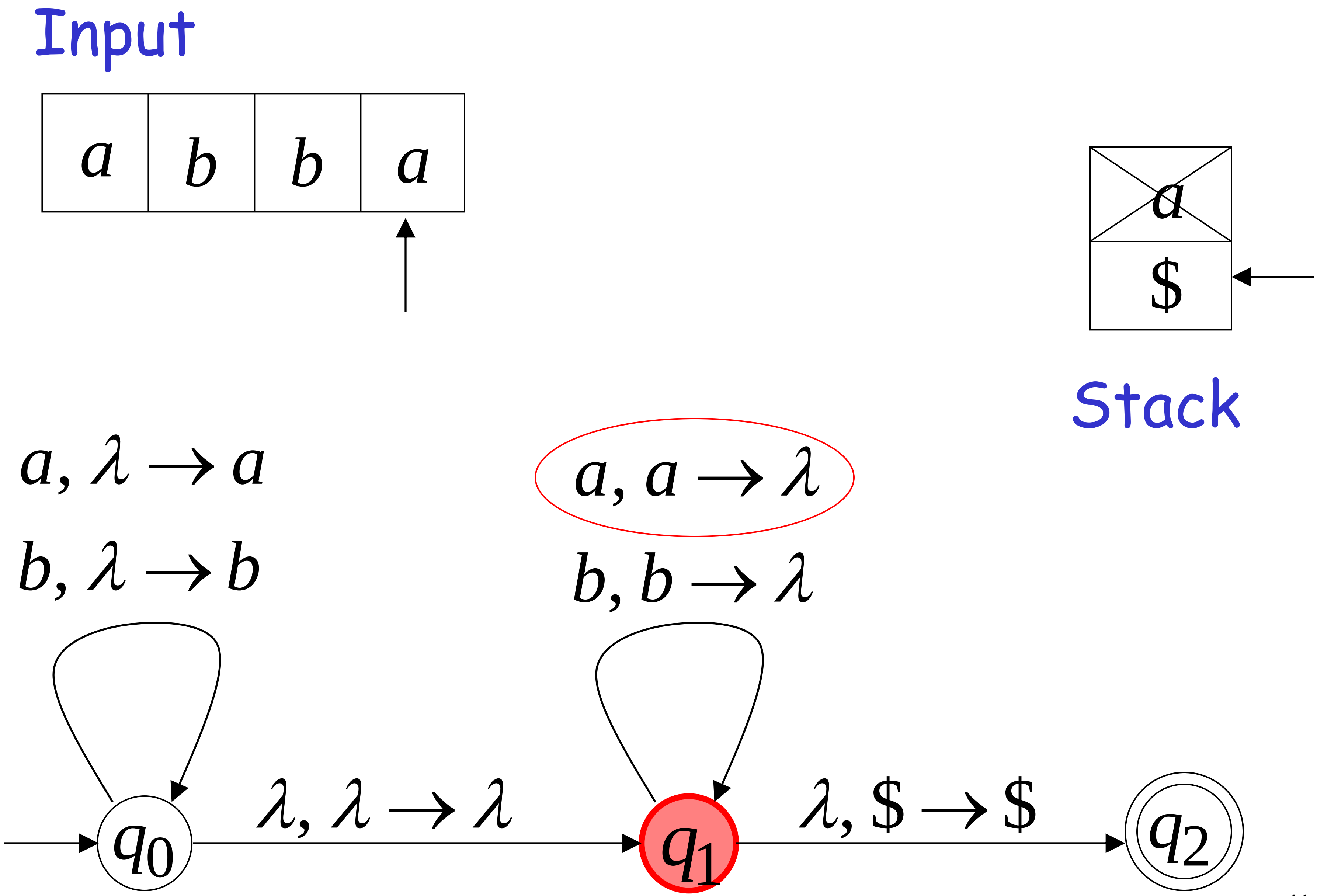




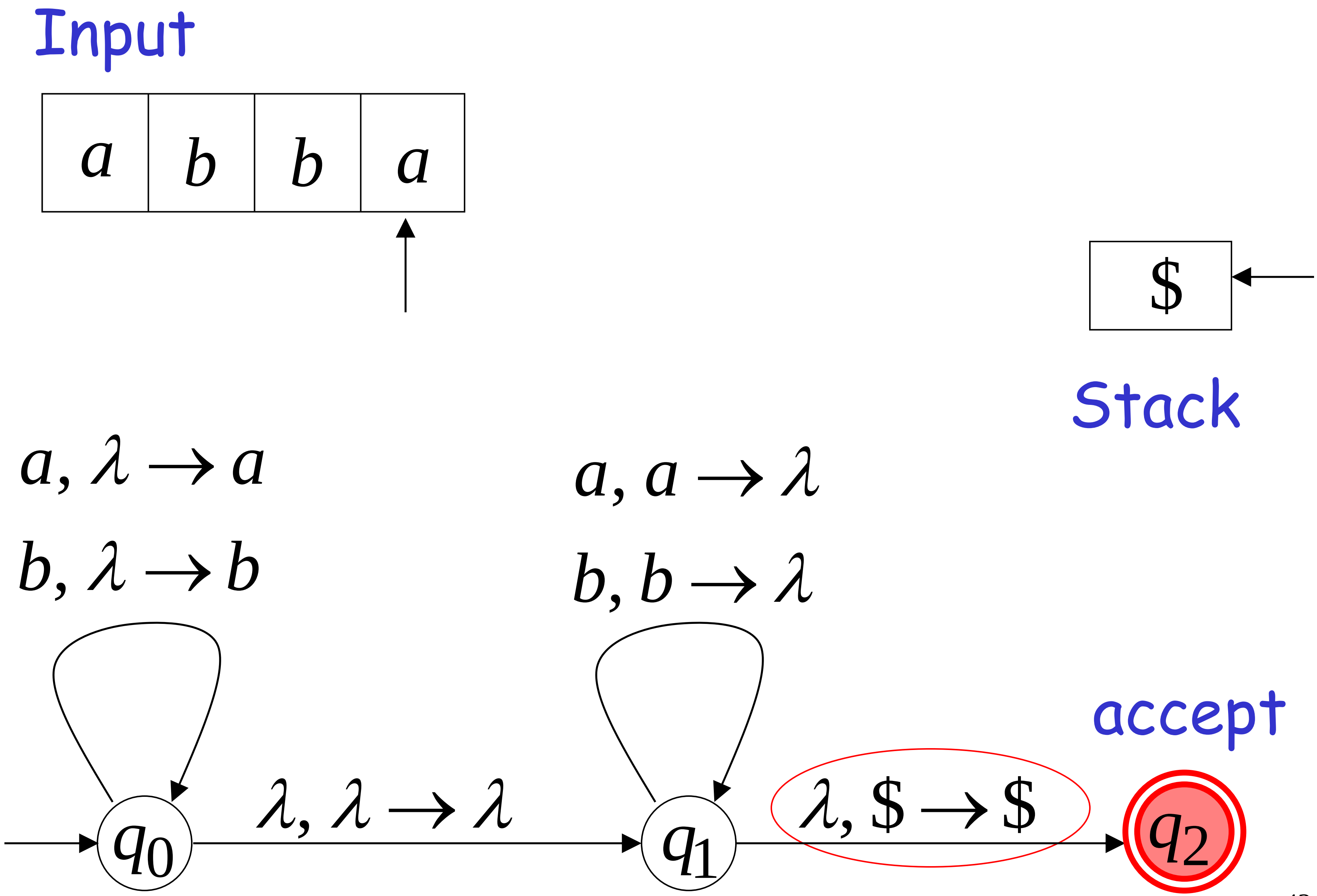
Stack



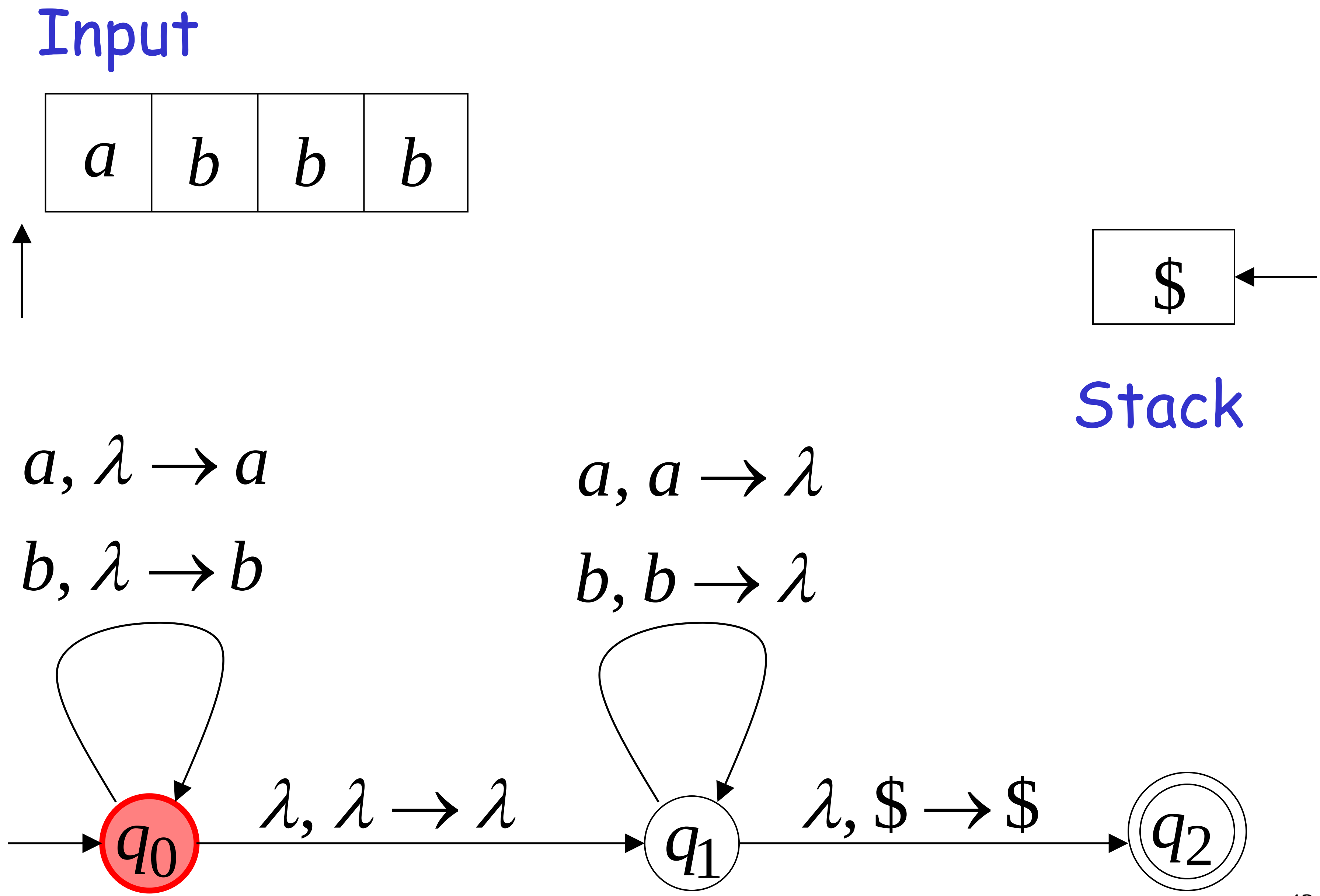
Time 5


Input
a
b
b
a
a
$
Stack
$a, \lambda \rightarrow a$
$b, \lambda \rightarrow b$
$a, a \rightarrow \lambda$
$b, b \rightarrow \lambda$
$\lambda, \lambda \rightarrow \lambda$
$\lambda, \$ \rightarrow \$$
q_0
q_1
q_2

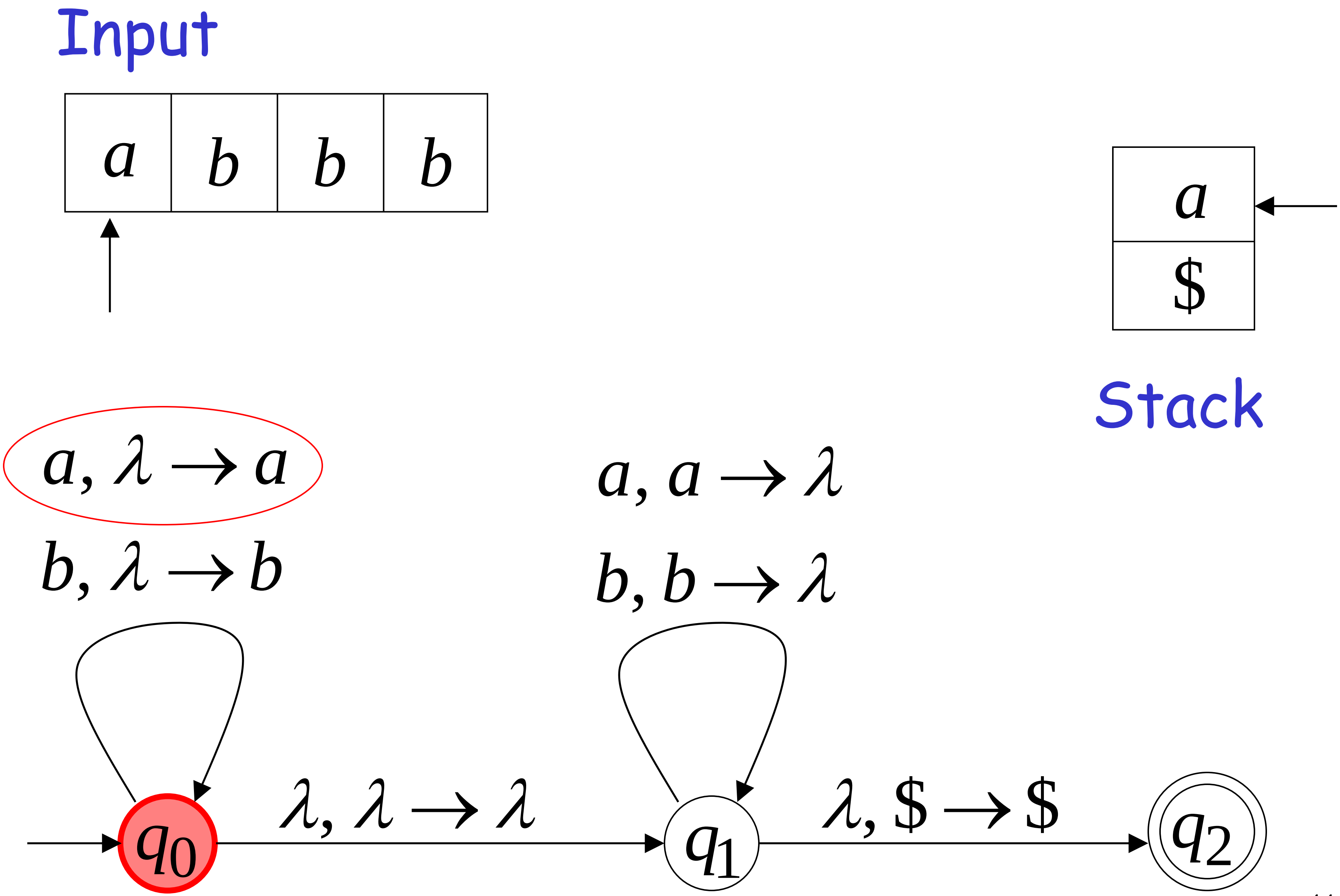
Time 6



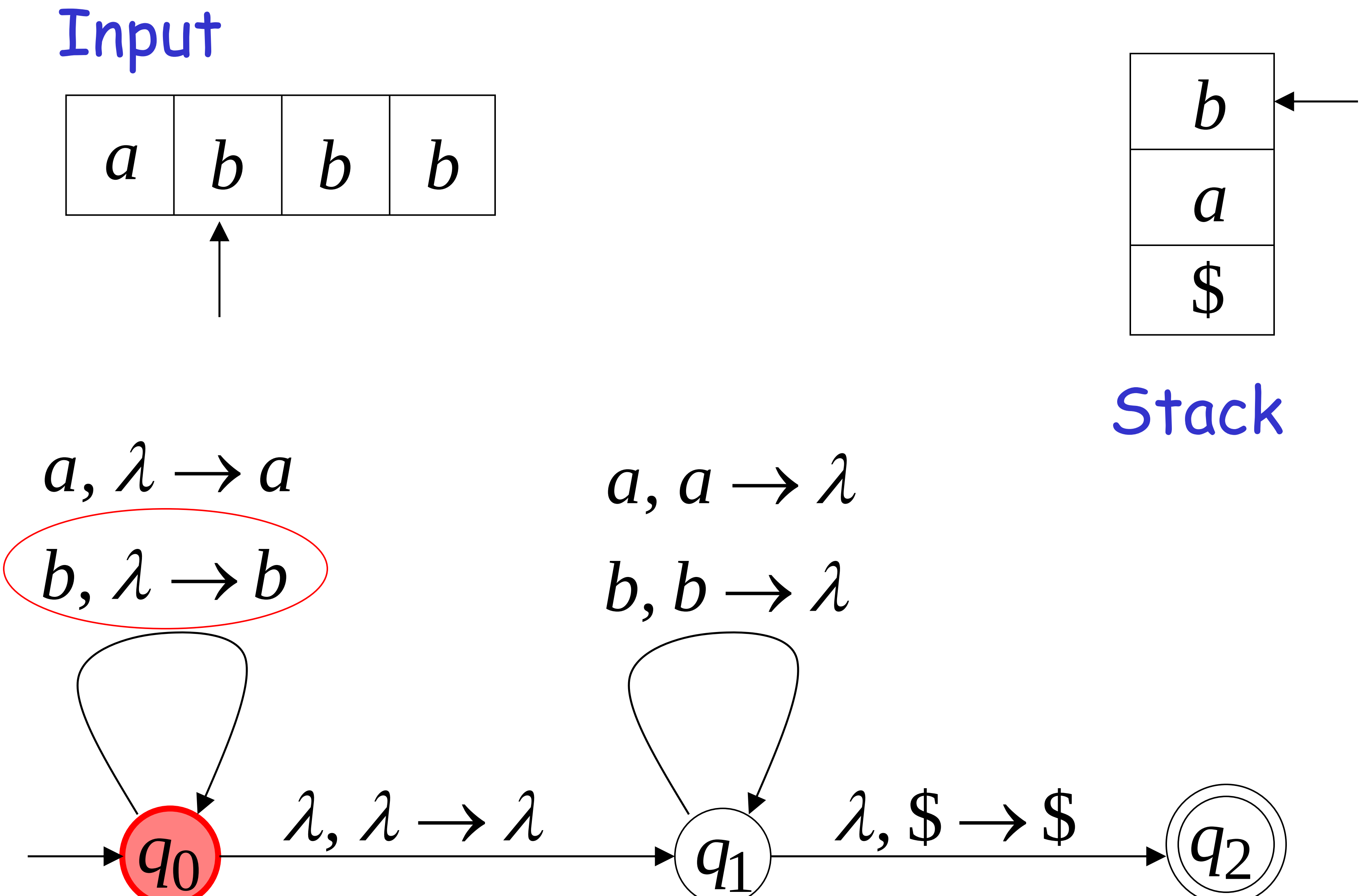
Rejection Example: Time 0



Time 1



Time 2



Time 3

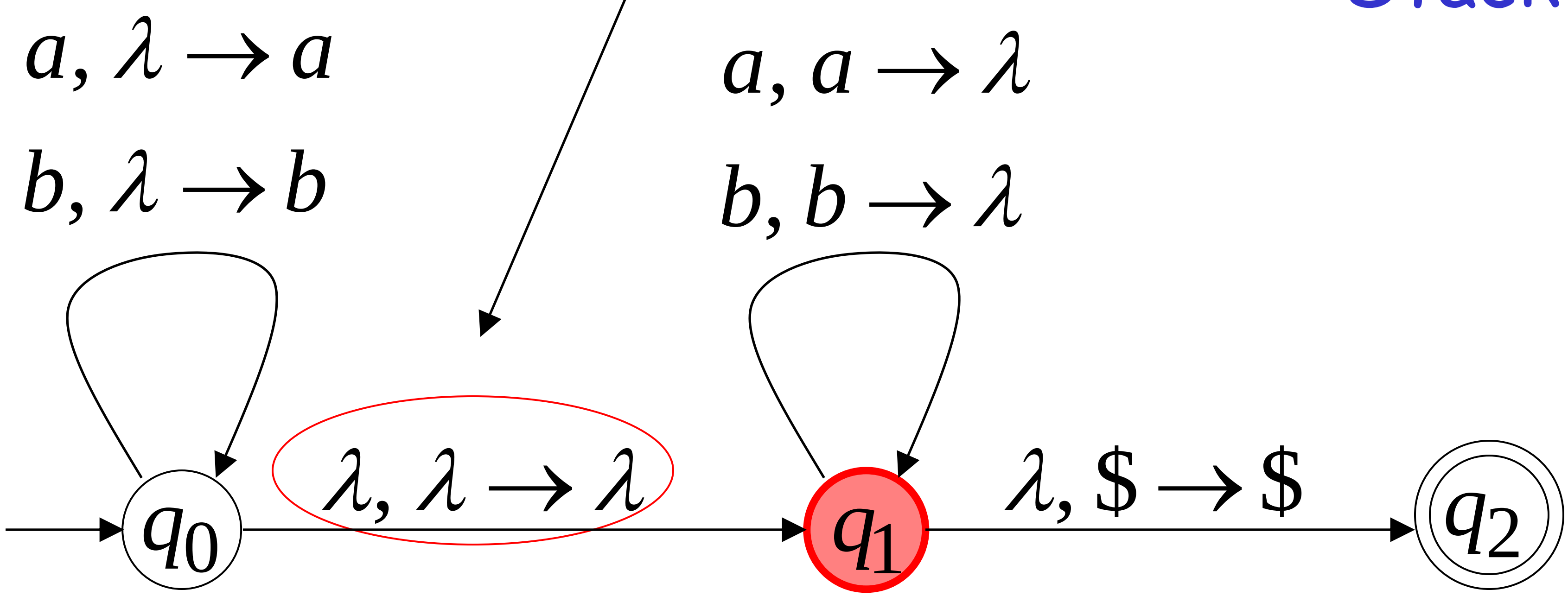
Input

a	b	b	b

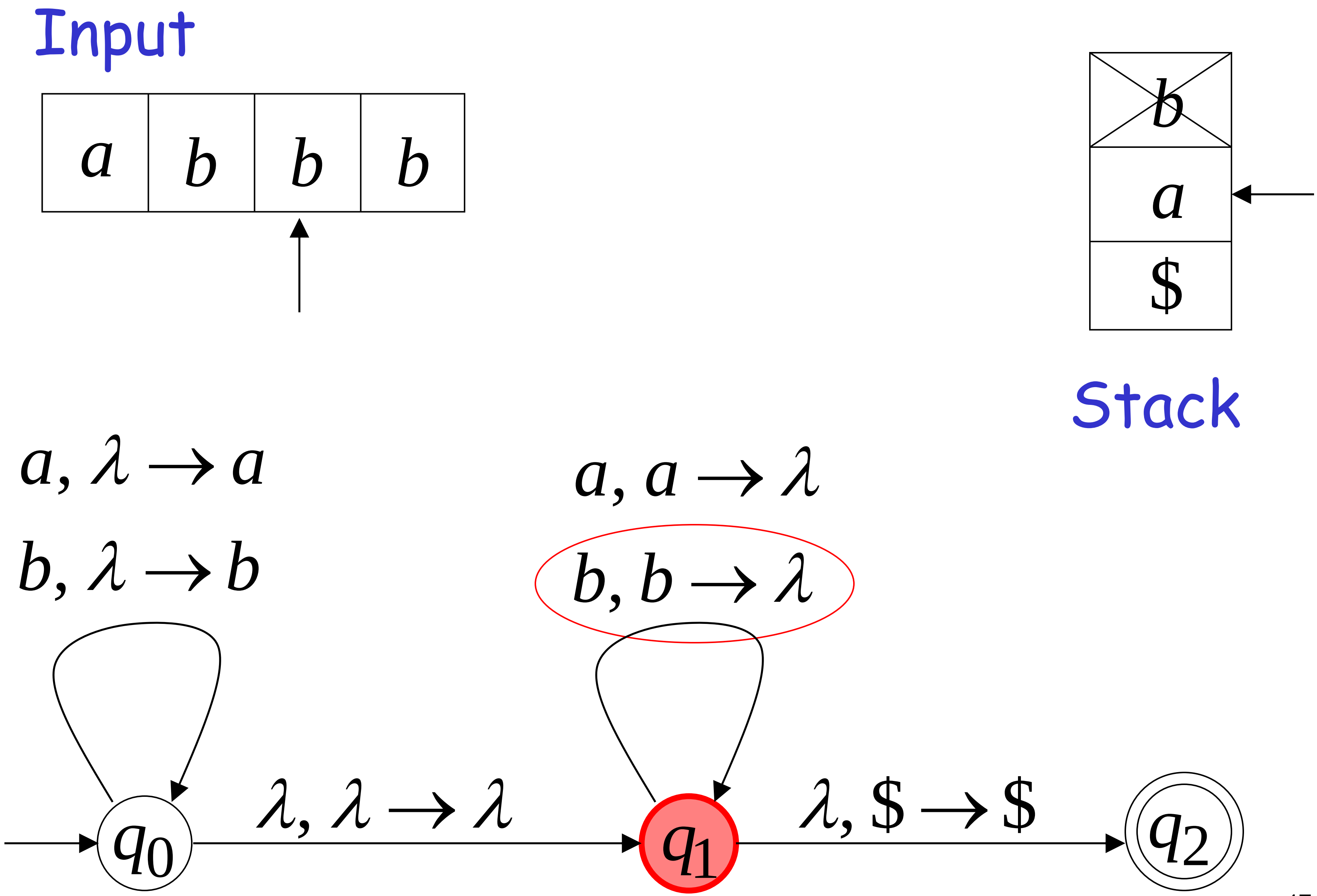
Guess the middle of string

b
a
$\$$

Stack



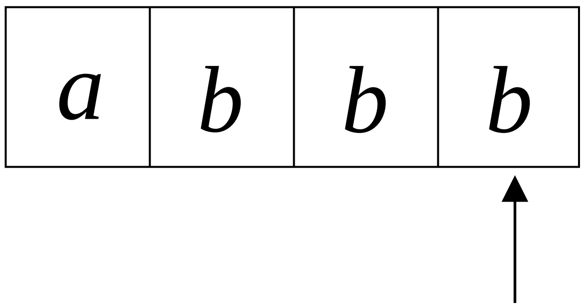
Time 4



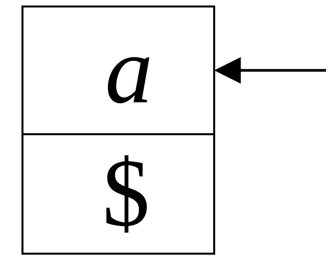
Time 5

There is no possible transition.

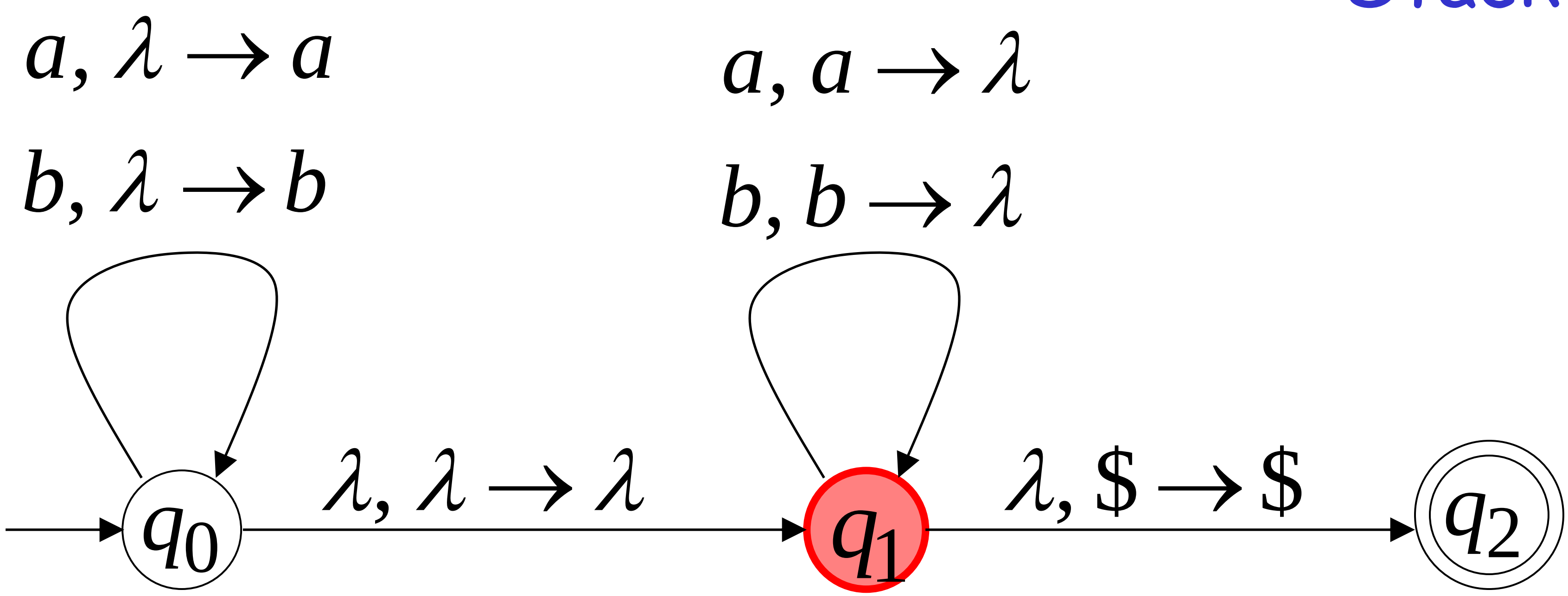
Input



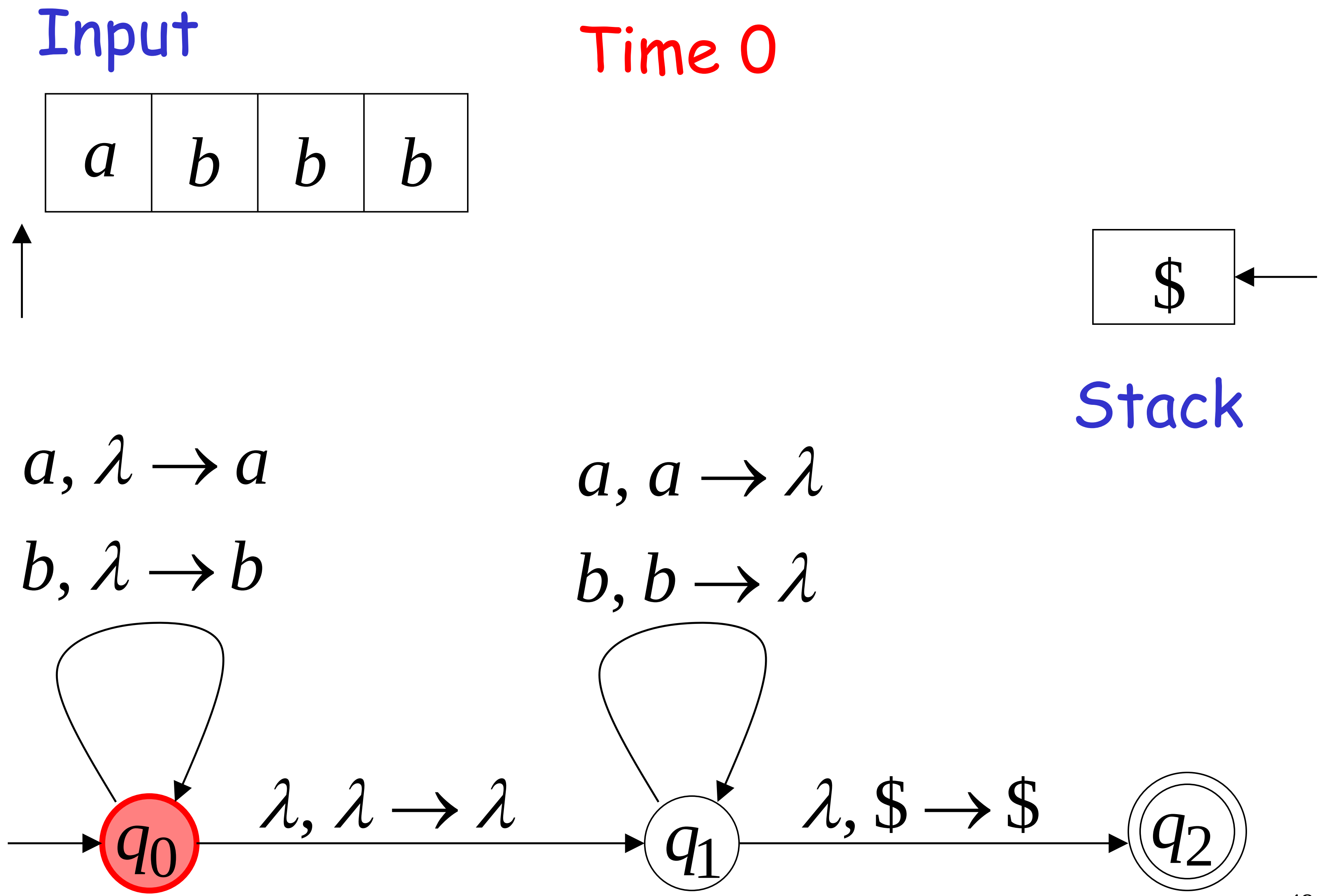
Input is not consumed



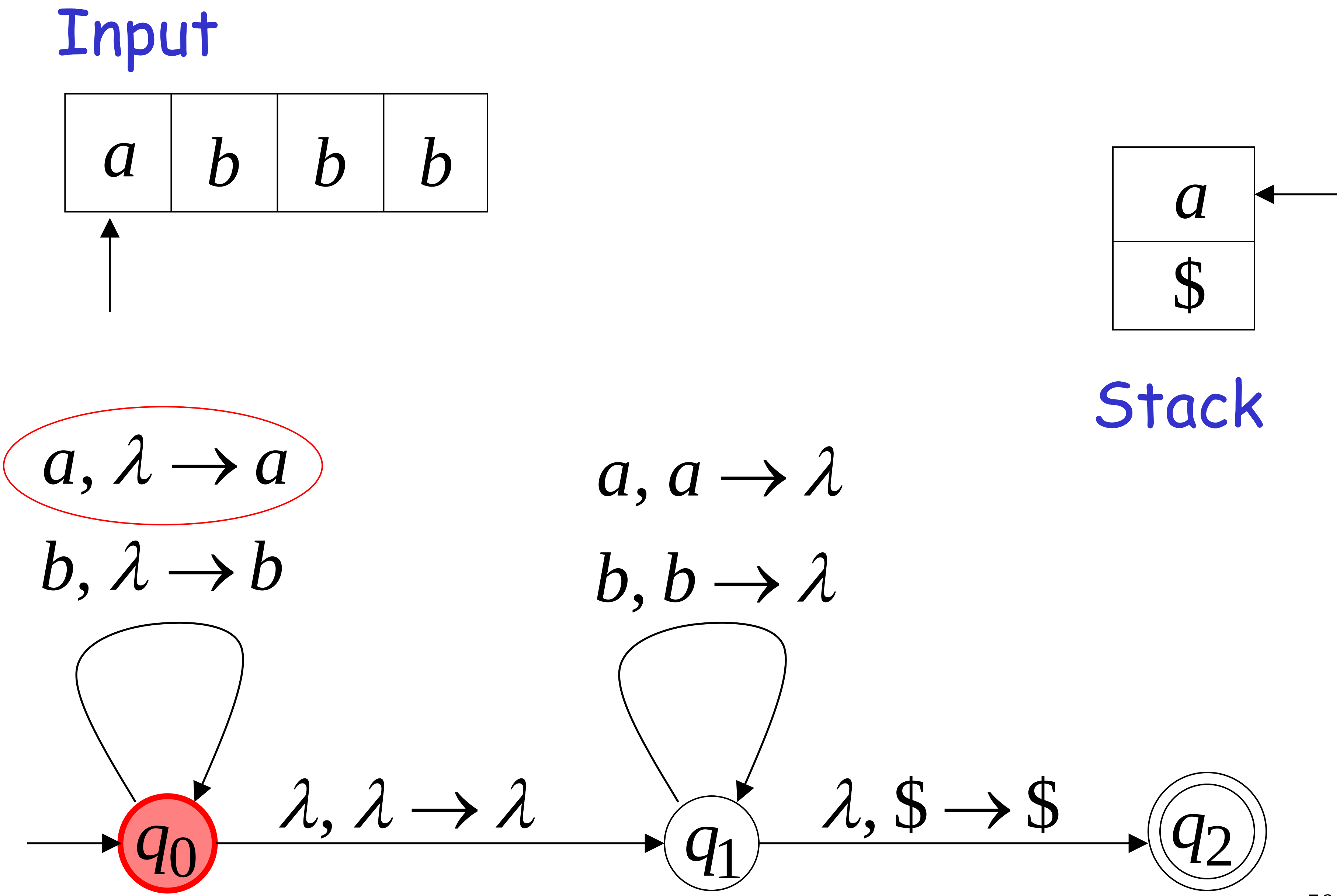
Stack



Another computation on same string:

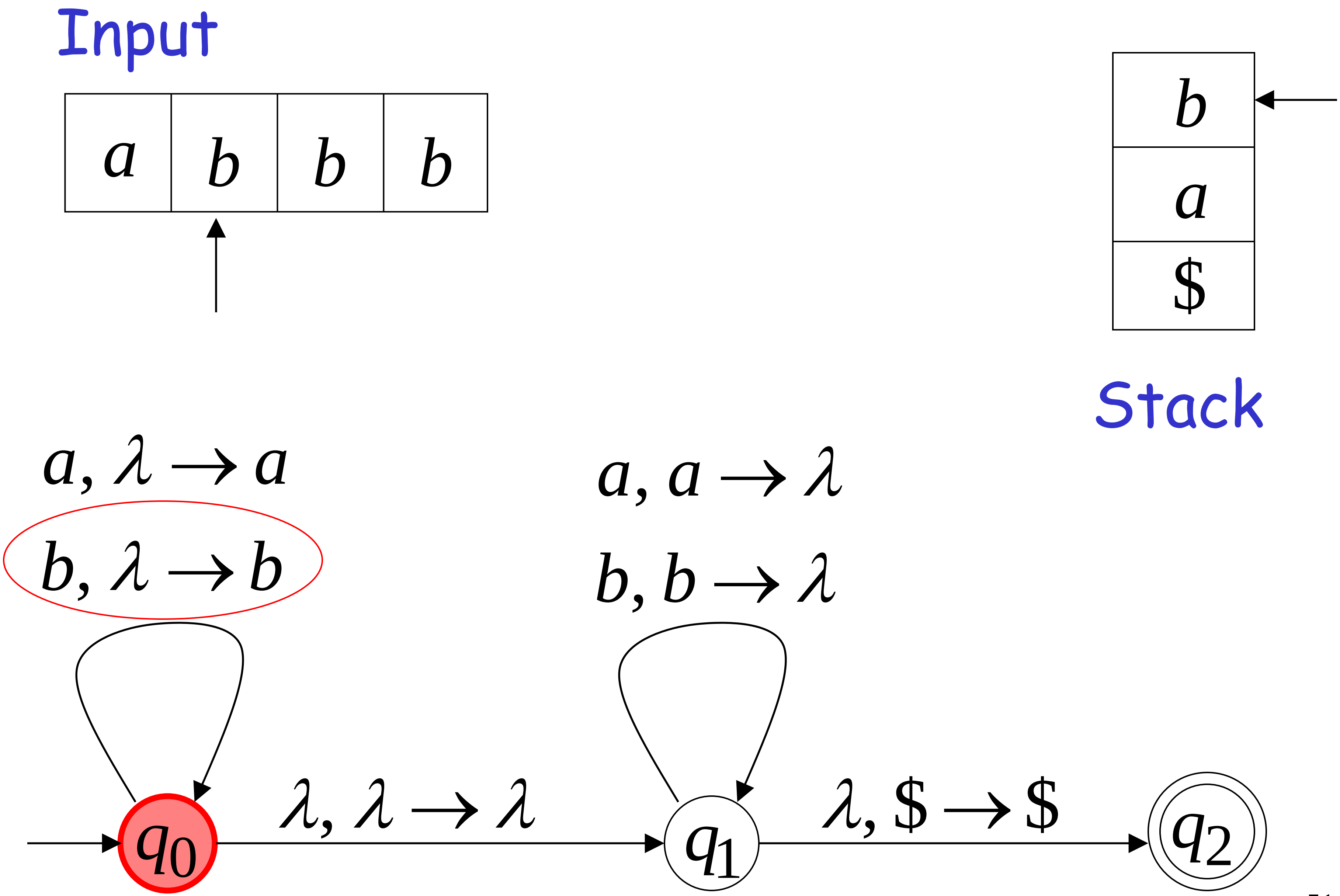


Time 1

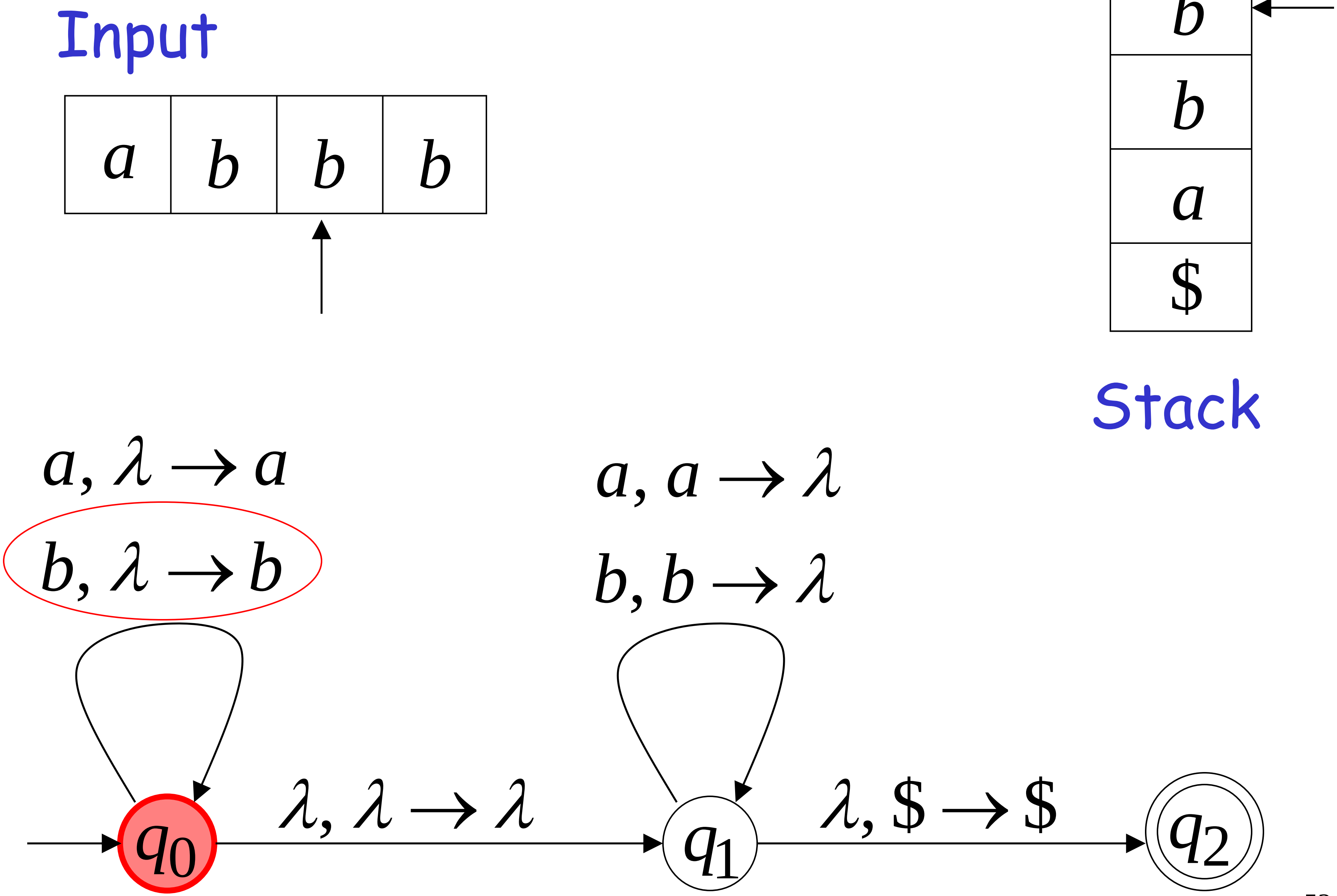


Input
a b b b
a
$\$$
Stack
$a, \lambda \rightarrow a$
$b, \lambda \rightarrow b$
$a, a \rightarrow \lambda$
$b, b \rightarrow \lambda$
$\lambda, \lambda \rightarrow \lambda$
$\lambda, \$ \rightarrow \$$
q_0
q_1
q_2

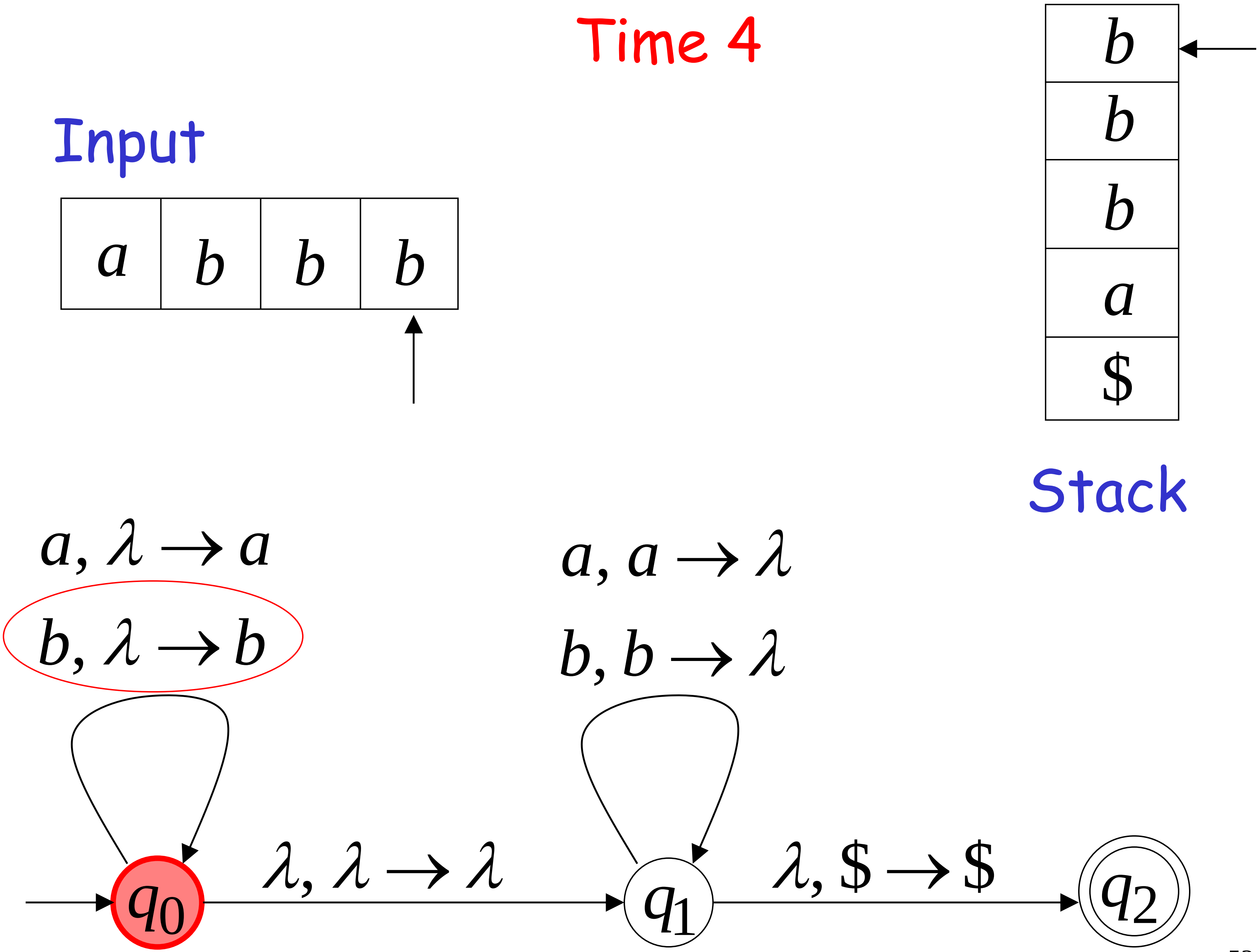
Time 2



Time 3


Input
a
b
b
b
b
b
a
$
Stack
$a, \lambda \rightarrow a$
$b, \lambda \rightarrow b$
$a, a \rightarrow \lambda$
$b, b \rightarrow \lambda$
$\lambda, \lambda \rightarrow \lambda$
$\lambda, \$ \rightarrow \$$
q_0
q_1
q_2

Time 4



Time 5

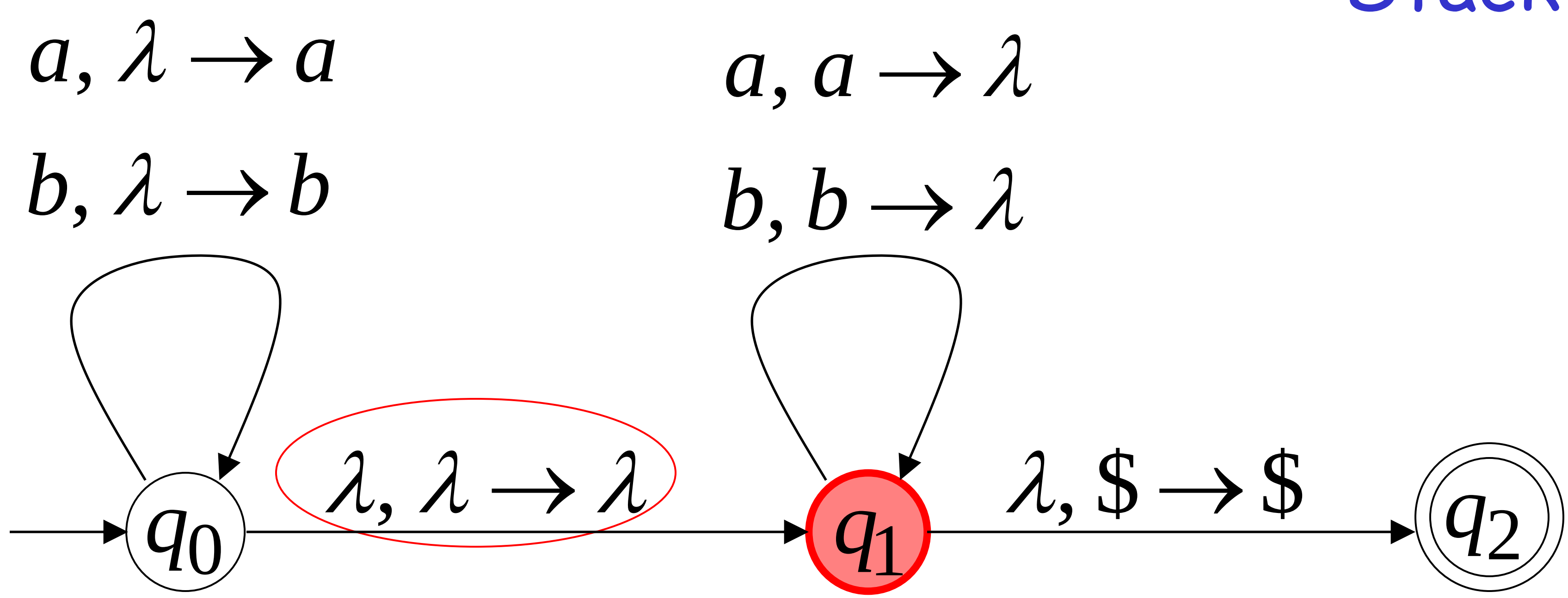
Input

a	b	b	b

No accept state is reached

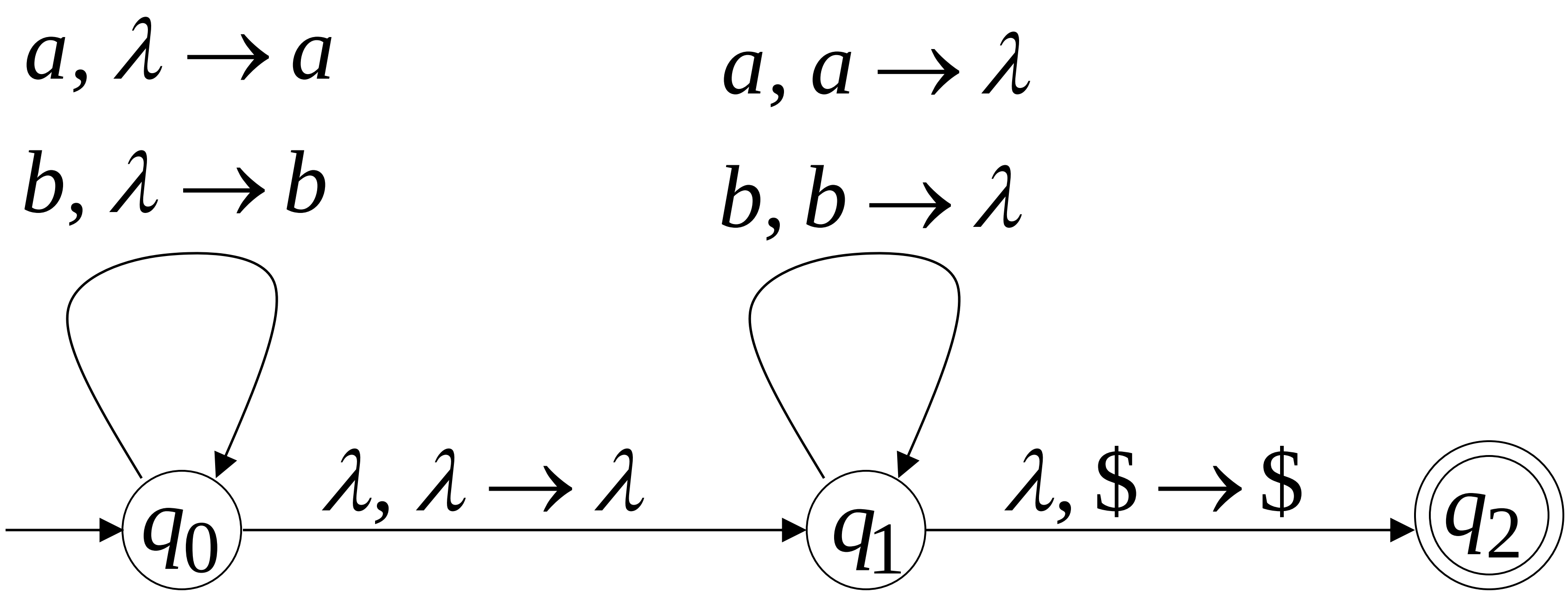
b
b
b
a
$\$$

Stack

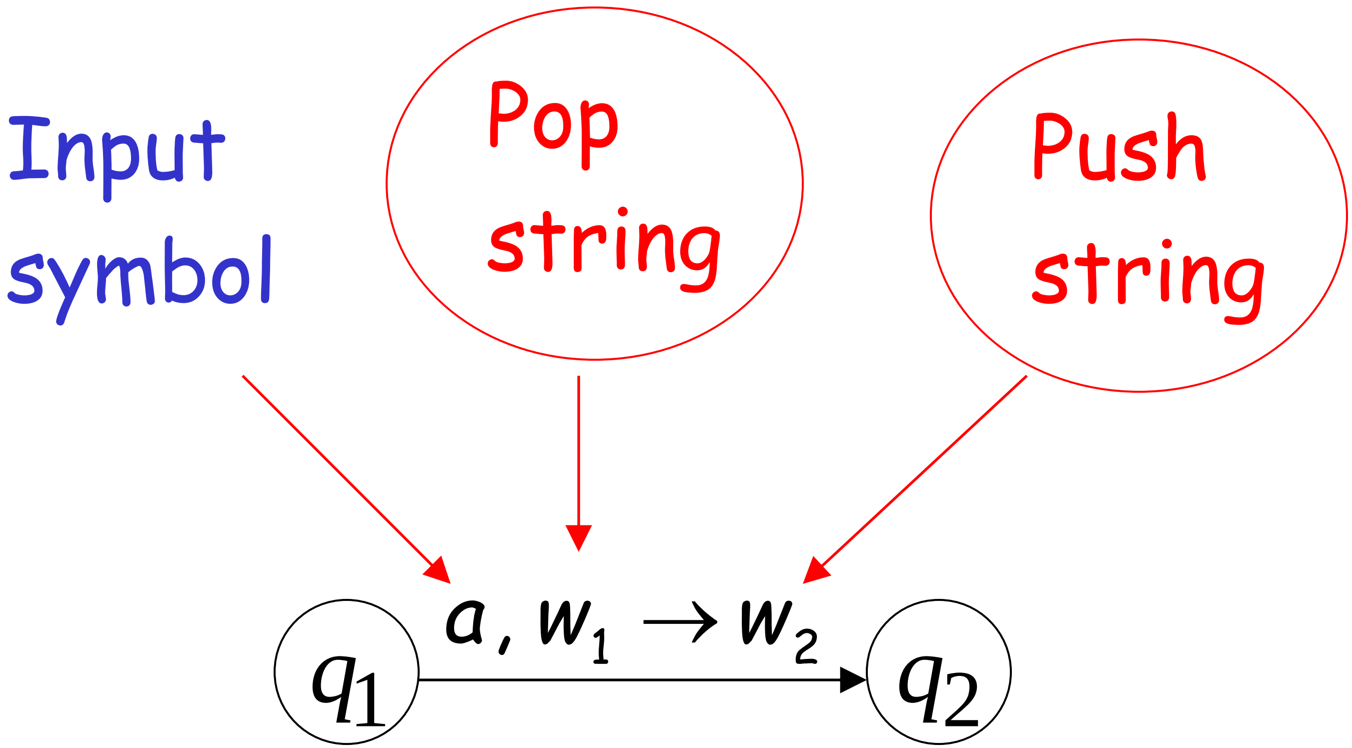


There is no computation that accepts string $abbb$

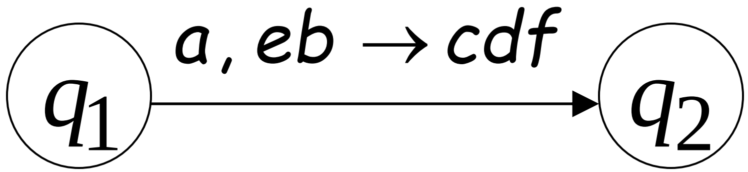
$$abbb \notin L(M)$$

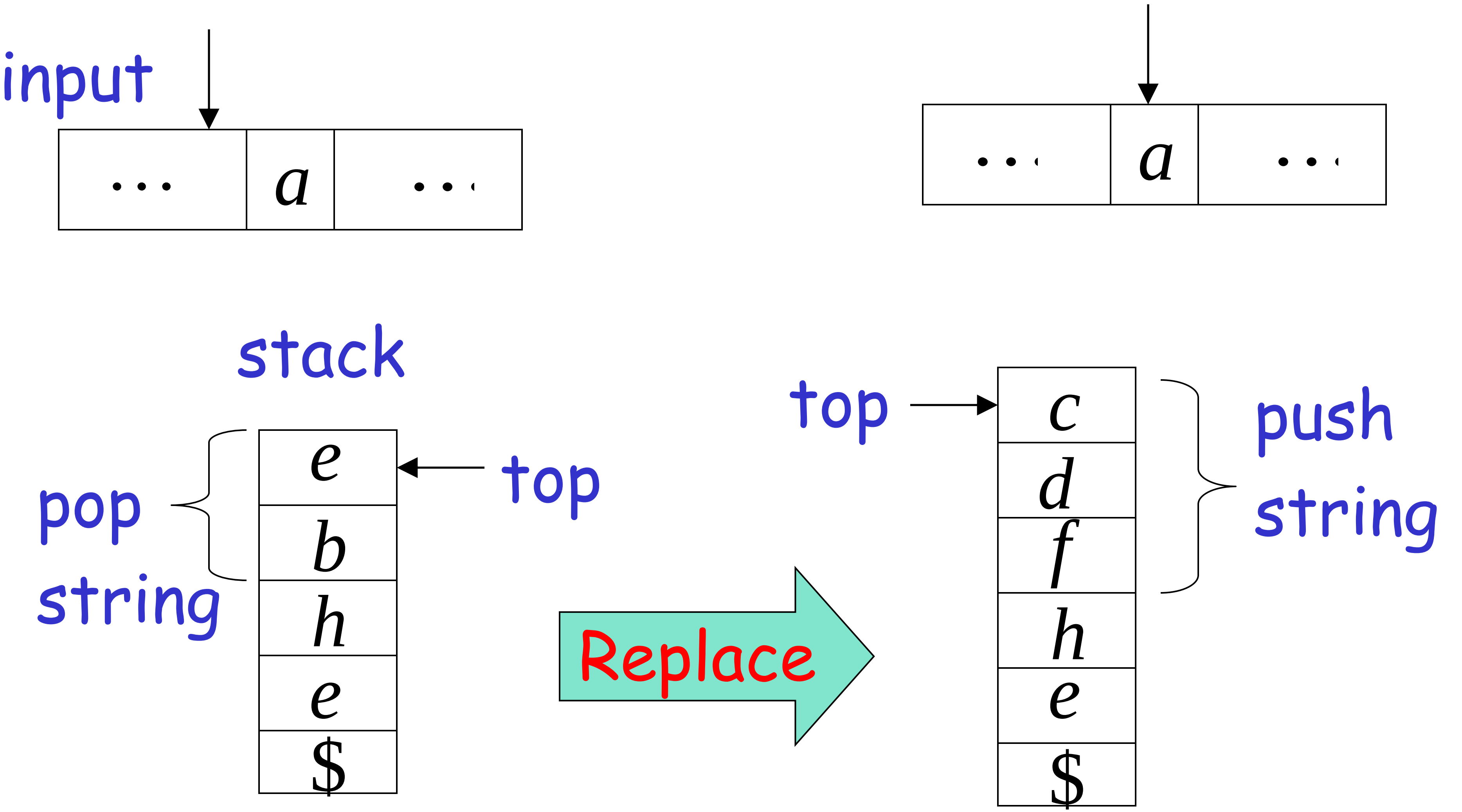


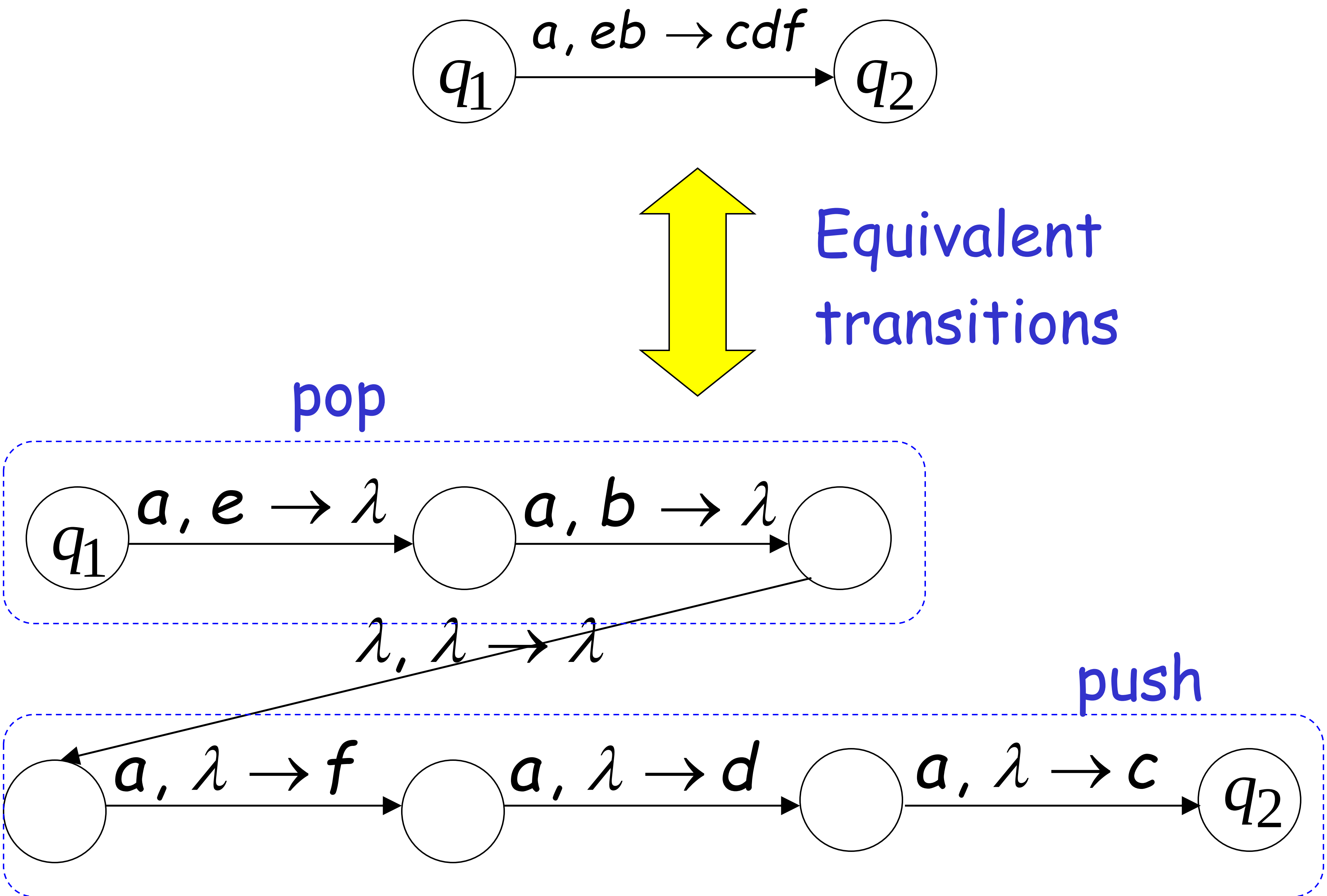
Pushing & Popping Strings



Example:


q_1
$a, eb \to cdf$
q_2


input
a
a
stack
pop string
e
b
h
e
$
top
Replace
top
c
d
f
h
e
$
push string



$q_1 \xrightarrow{a,\ eb \to cdf} q_2$
Equivalent transitions
pop
$a,\ e \to \lambda$
$a,\ b \to \lambda$
$\lambda,\ \lambda \to \lambda$
push
$a,\ \lambda \to f$
$a,\ \lambda \to d$
$a,\ \lambda \to c$
q_1
q_2

Another PDA example

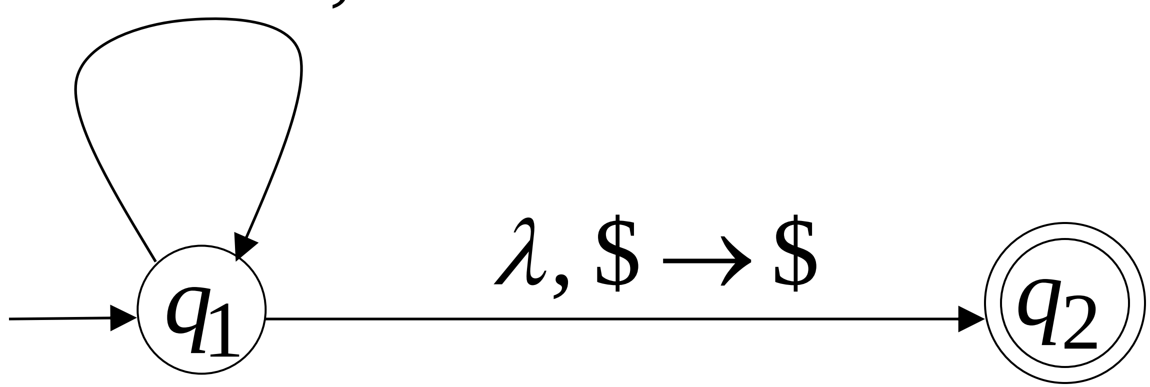
$$L(M) = \{w \in \{a,b\}^* : \quad n_a(w) = n_b(w)\}$$

PDA M

$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$

$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$

$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$



Execution Example: Time 0

Input

a	b	b	b	a	a

$\$$

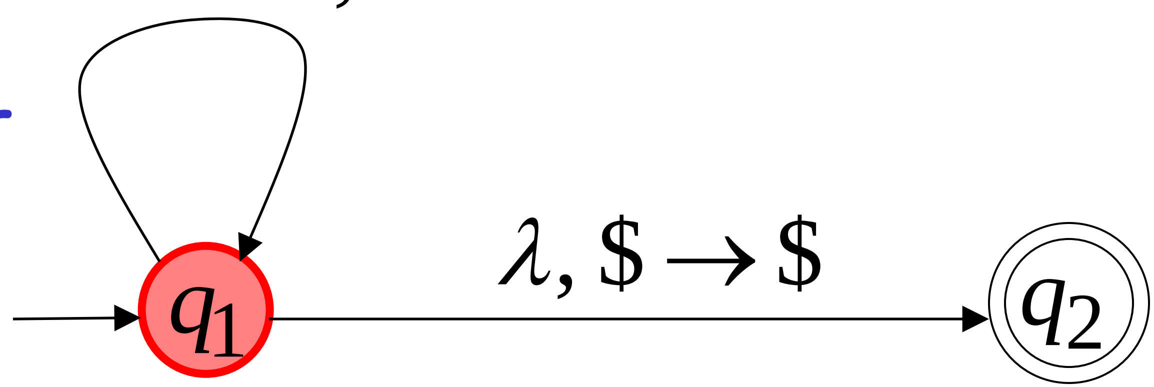
Stack

$$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$$

$$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$$

$$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$$

current state



Time 1

Input

a	b	b	b	a	a

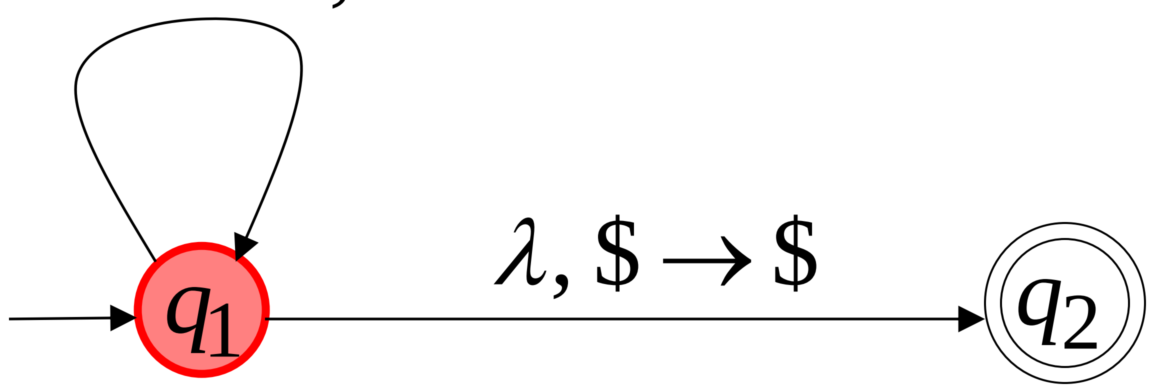
Stack

0
$\$$

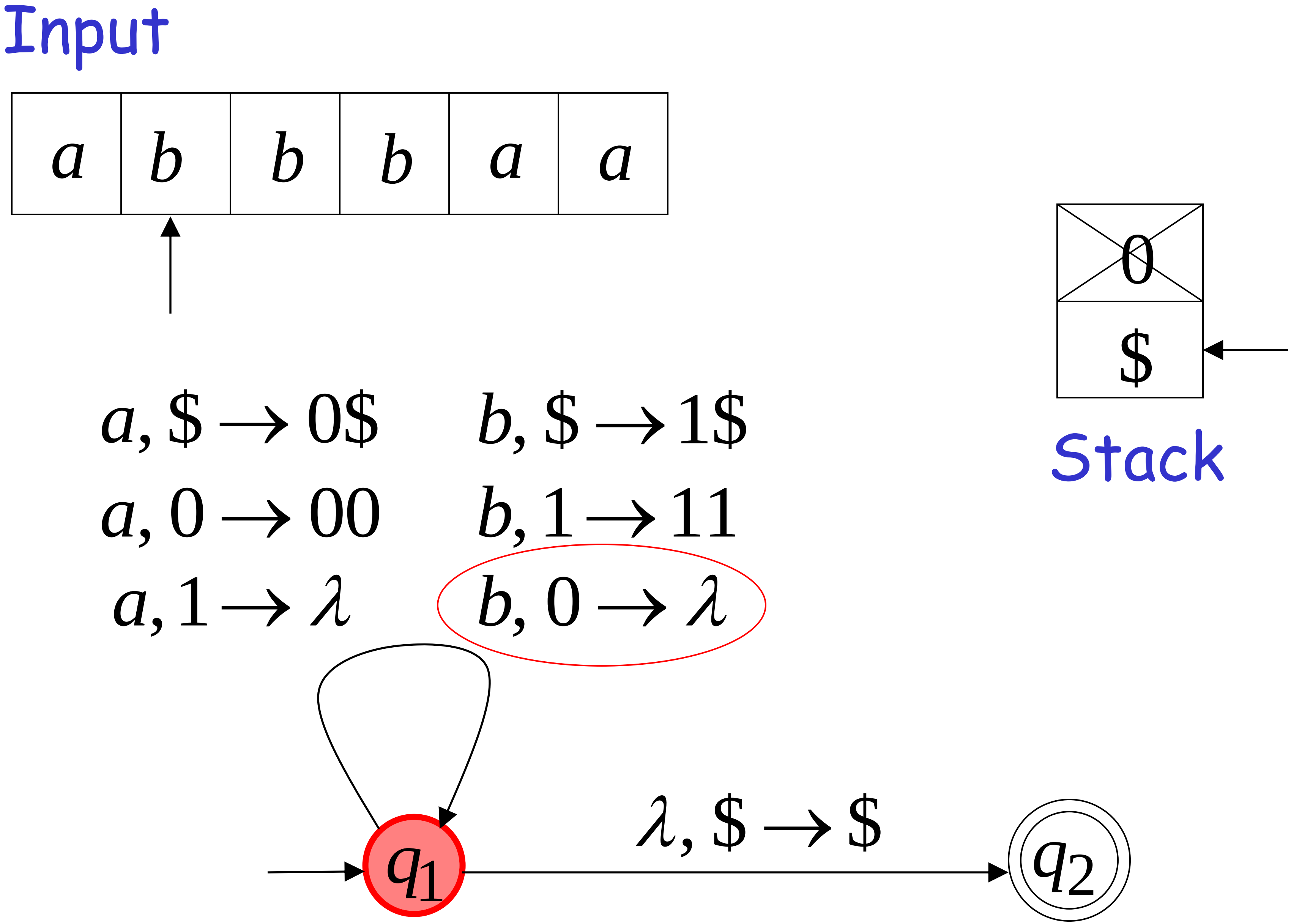
$$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$$

$$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$$

$$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$$



Time 3



Input
a b b b a a
0
$
Stack
$a, \$ \rightarrow 0\$$
$b, \$ \rightarrow 1\$$
$a, 0 \rightarrow 00$
$b, 1 \rightarrow 11$
$a, 1 \rightarrow \lambda$
$b, 0 \rightarrow \lambda$
$\lambda, \$ \rightarrow \$$
q_1
q_2

Time 4

Input

a	b	b	b	a	a

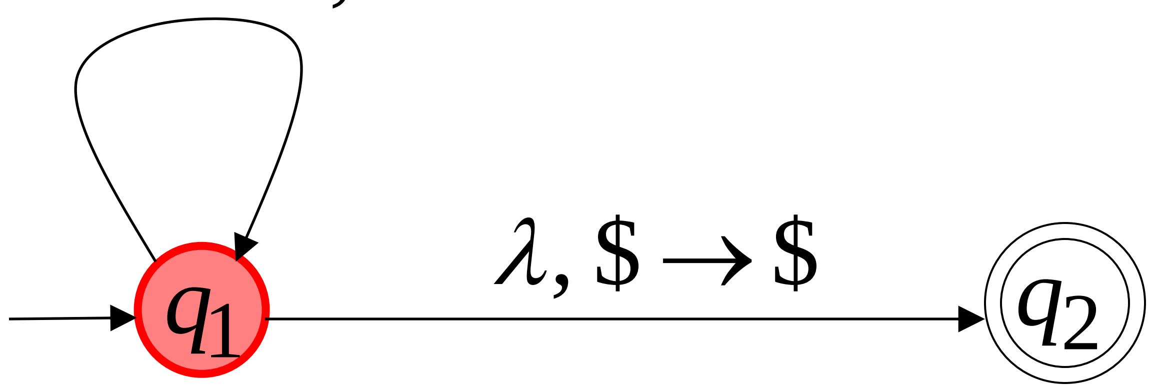
1
$\$$

Stack

$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$

$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$

$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$



Time 5

Input

a	b	b	b	a	a

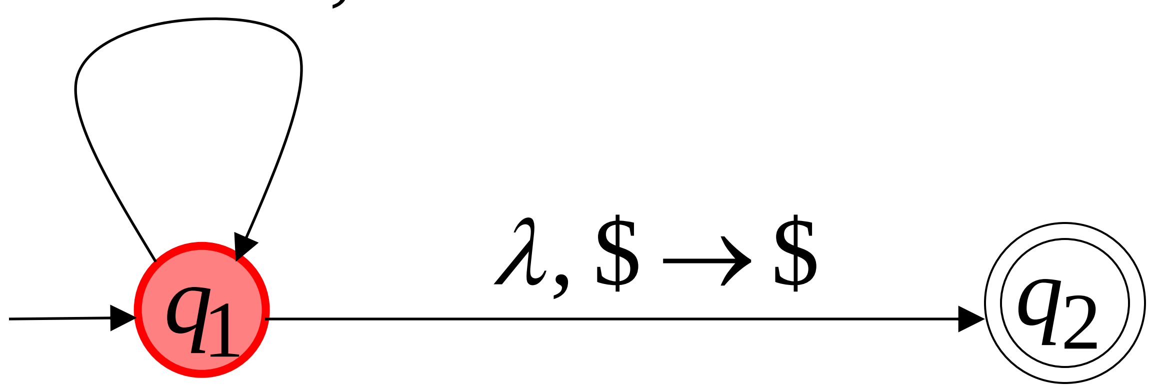
Stack

1
1
$

$$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$$

$$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$$

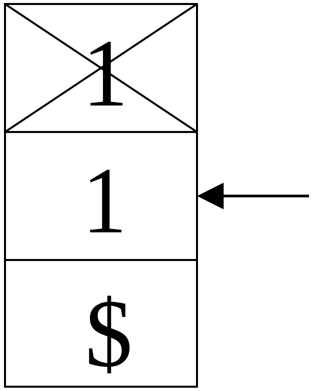
$$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$$



Time 6

Input

a	b	b	b	a	a

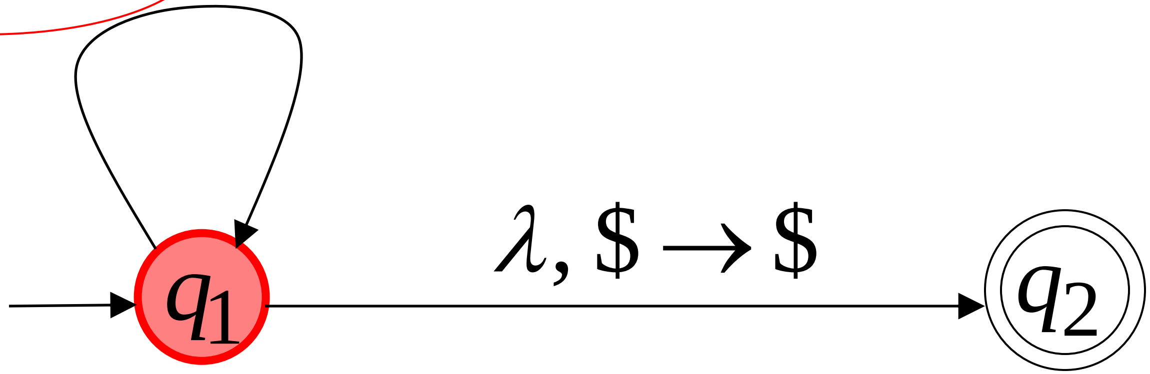


Stack

$a, \$ \rightarrow 0\$$ $\quad$ $b, \$ \rightarrow 1\$$

$a, 0 \rightarrow 00$ $\quad$ $b, 1 \rightarrow 11$

$a, 1 \rightarrow \lambda$ $\quad$ $b, 0 \rightarrow \lambda$



Time 7

Input

a	b	b	b	a	a

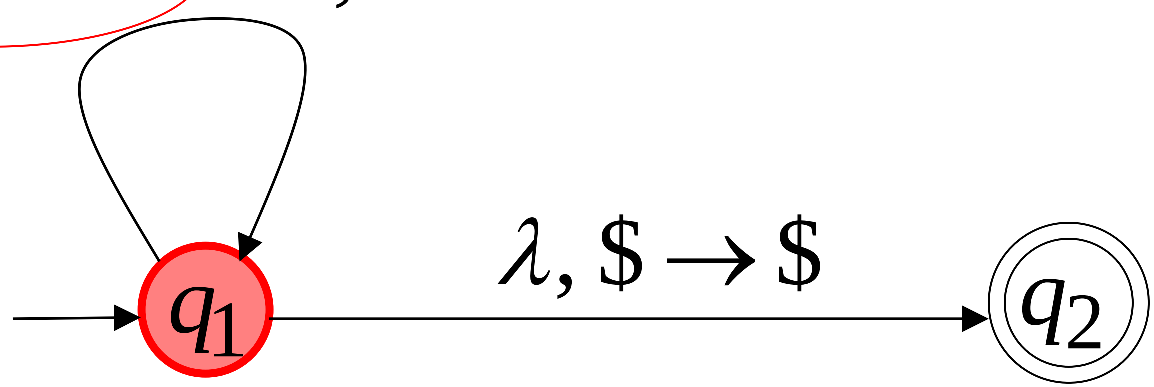
Stack

$\cancel{1}$
$\$$

$$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$$

$$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$$

$$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$$



Time 8

Input

a	b	b	b	a	a

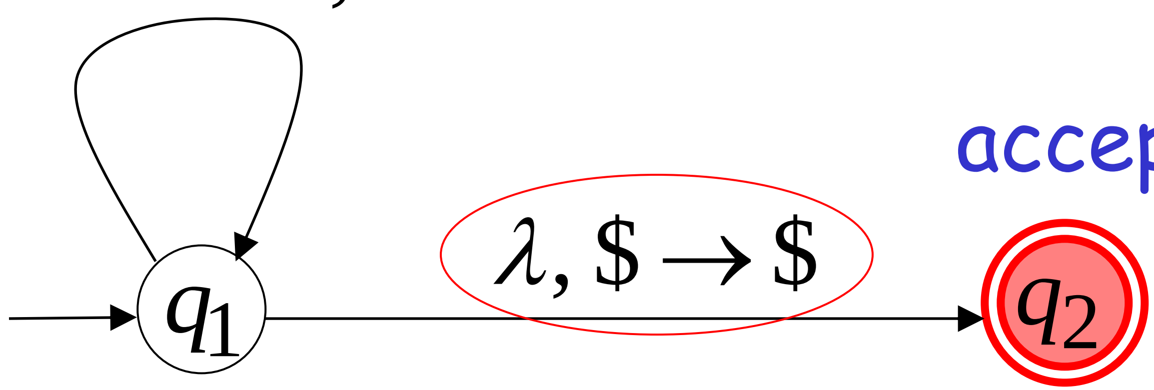
Stack

$\$$

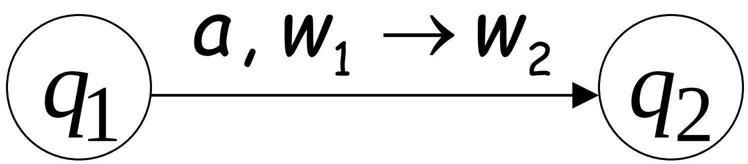
$$a, \$ \rightarrow 0\$ \qquad b, \$ \rightarrow 1\$$$

$$a, 0 \rightarrow 00 \qquad b, 1 \rightarrow 11$$

$$a, 1 \rightarrow \lambda \qquad b, 0 \rightarrow \lambda$$

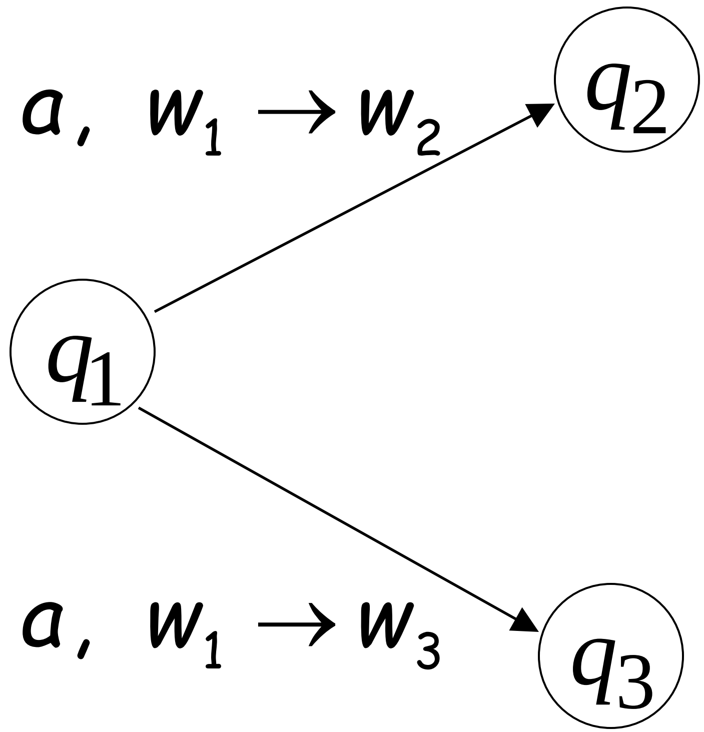


Formalities for PDAs



Transition function:

$$\delta(q_1, a, w_1) = \{(q_2, w_2)\}$$



Transition function:

$$\delta(q_1, a, w_1) = \{(q_2, w_2), \ (q_3, w_3)\}$$

Formal Definition

Pushdown Automaton (PDA)

$$M = (Q, \Sigma, \Gamma, \delta, q_0, z, F)$$

States

Input alphabet

Stack alphabet

Transition function

Initial state

Stack start symbol

Accept states

Example:Formal Definition for

$$L(M) = \{a^n b^n : n \geq 0\}$$

$$M = (Q, \Sigma, \Gamma, \delta, q_0, z, F)$$

-Q={q0, q1,q2,q3},

-∑={ a, b}

- Γ = {$, a, b}

- δ is a transitions function as

  δ(q0, ʌ, ʌ)= { (q1, ʌ) }

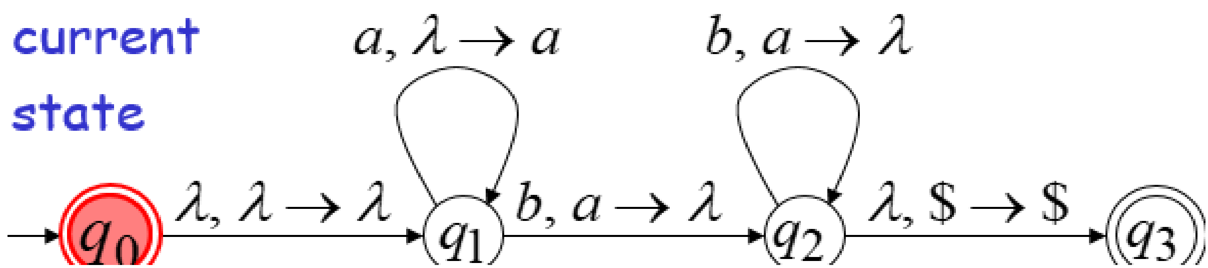
  δ(q1, a, ʌ)= { (q1, a) }

  δ(q1, b, a) ={ (q2, ʌ) }

  δ(q2, b, a)= { (q2, ʌ) }

  δ(q2, ʌ, $) = { (q3, $) }

- q0= q0,
- Z= $,
- F={q0,q3}.



Instantaneous Description

$$(q,\ u,\ s)$$

Current state

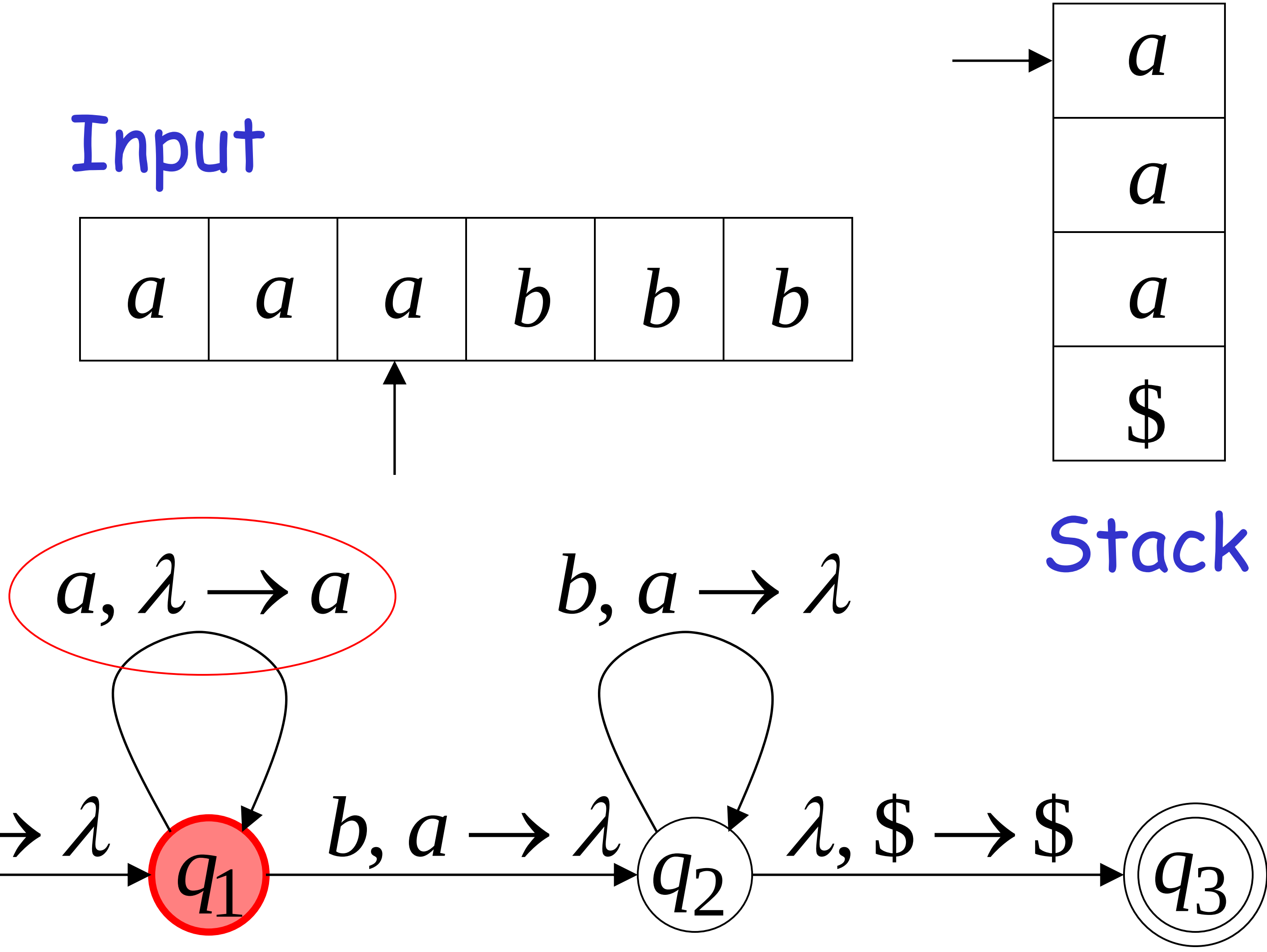
Remaining input

Current stack contents

Example: Instantaneous Description

$$(q_1, bbb, aaa\$)$$

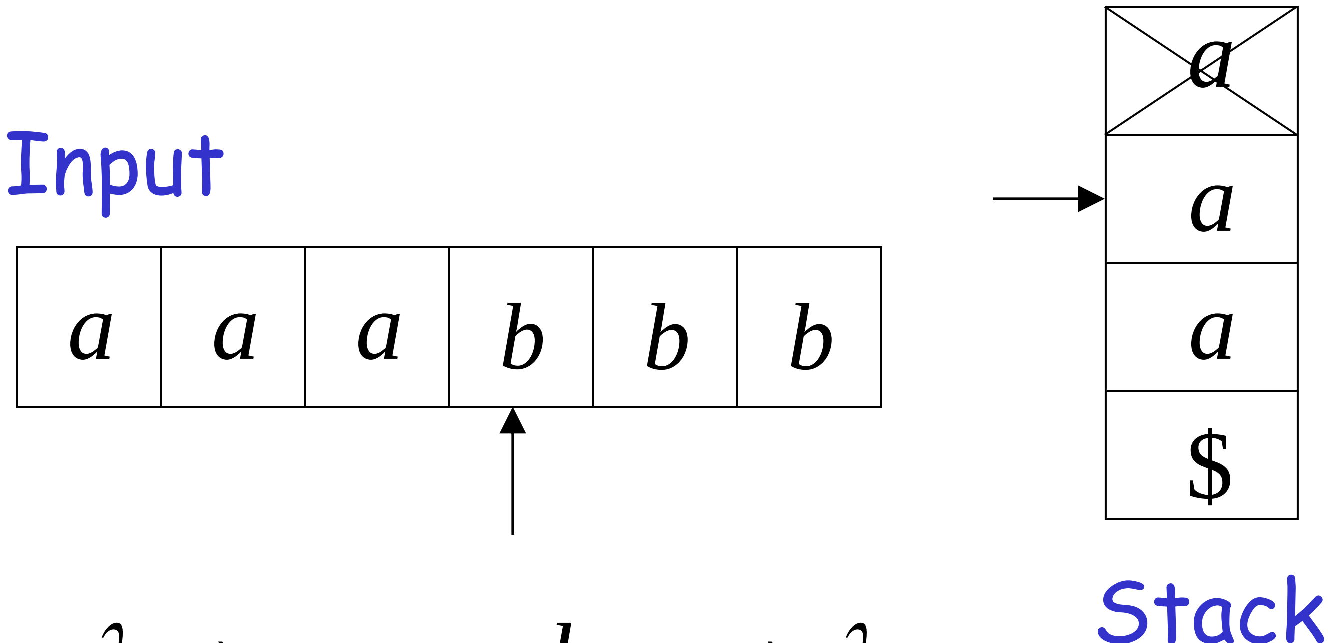
Time 4:

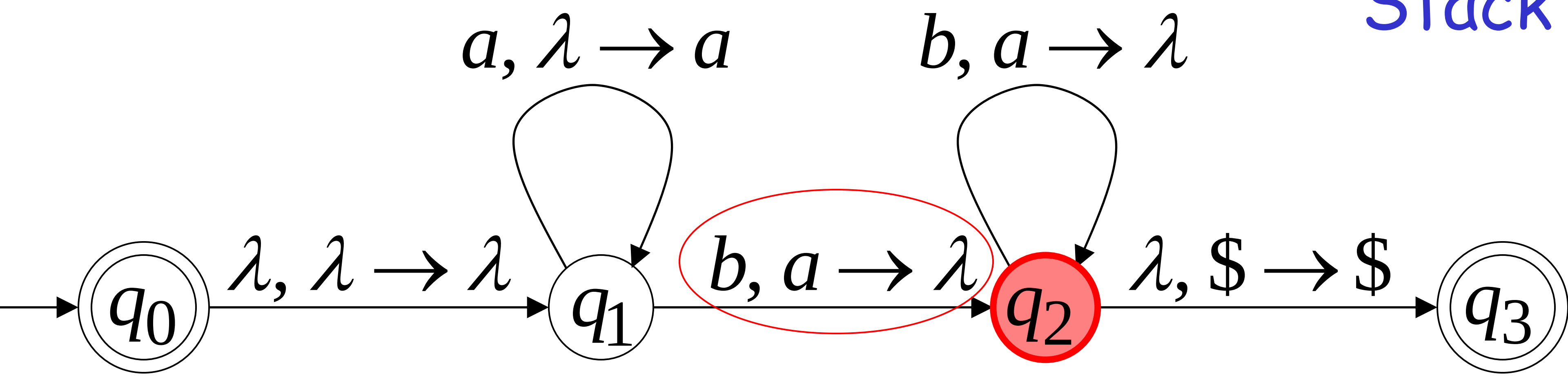


Example: Instantaneous Description

$$(q_2, bb, aa\$)$$

Time 5:





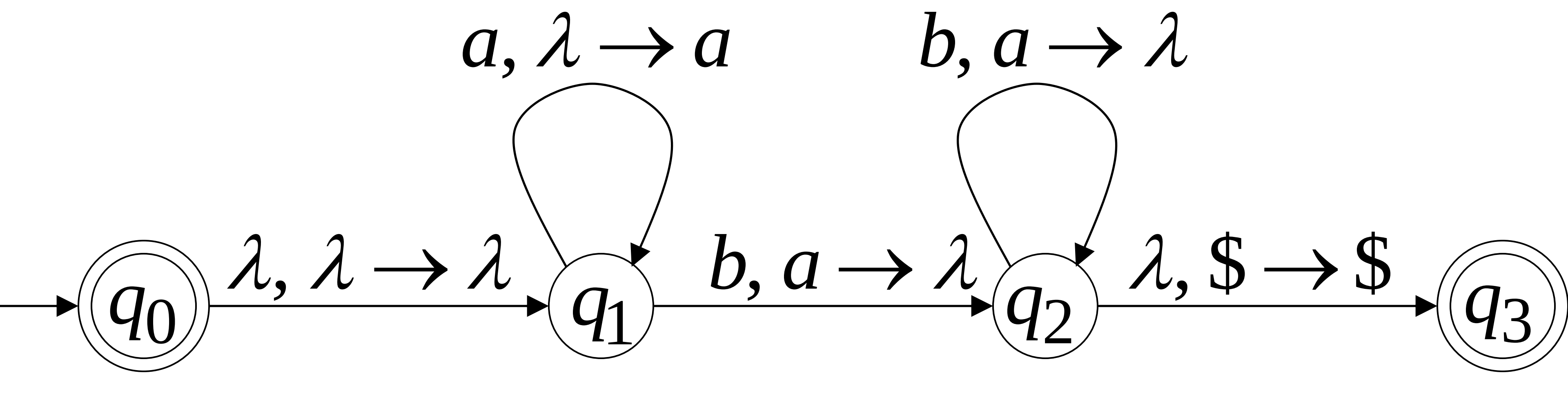
We write:

$$(q_1, bbb, aaa\$) \succ (q_2, bb, aa\$)$$

Time 4 Time 5

A computation:

$$(q_0, aaabbb, \$) \succ (q_1, aaabbb, \$) \succ$$
$$(q_1, aabbb, a\$) \succ (q_1, abbb, aa\$) \succ (q_1, bbb, aaa\$) \succ$$
$$(q_2, bb, aa\$) \succ (q_2, b, a\$) \succ (q_2, \lambda, \$) \succ (q_3, \lambda, \$)$$



$$(q_0, aaabbb, \$) \succ (q_1, aaabbb, \$) \succ$$
$$(q_1, aabbb, a\$) \succ (q_1, abbb, aa\$) \succ (q_1, bbb, aaa\$) \succ$$
$$(q_2, bb, aa\$) \succ (q_2, b, a\$) \succ (q_2, \lambda, \$) \succ (q_3, \lambda, \$)$$

For convenience we write:

$$(q_0, aaabbb, \$) \overset{*}{\succ} (q_3, \lambda, \$)$$

Language of PDA

Language $L(M)$ accepted by PDA M :

$$L(M) = \{w : \ (q_0, w, z) \overset{*}{\succ} (q_f, \lambda, s)\}$$

Initial state

Accept state

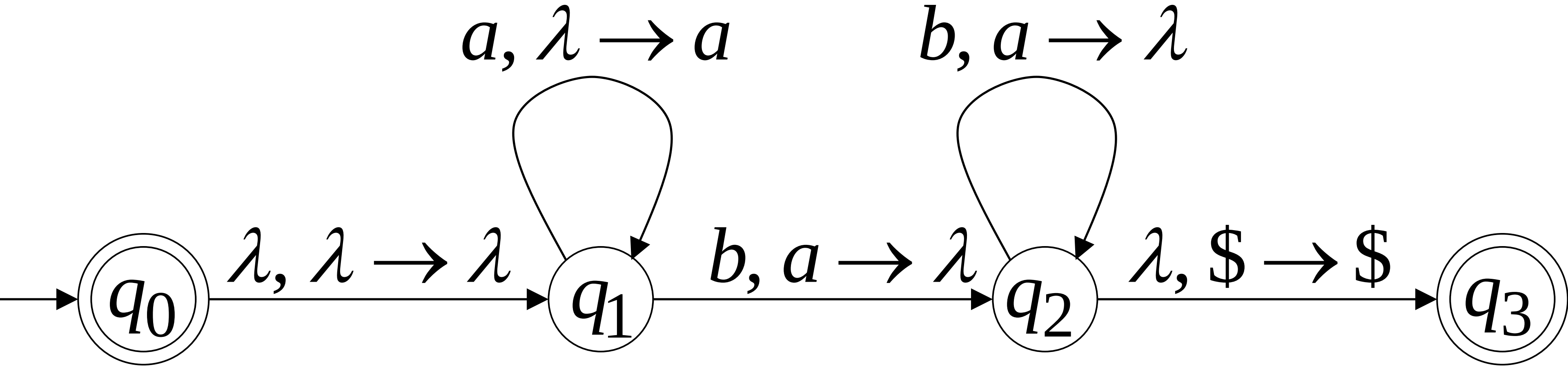
Example:

$$(q_0, aaabbb, \$) \overset{*}{\succ} (q_3, \lambda, \$)$$

$$\Downarrow$$

$$aaabbb \in L(M)$$

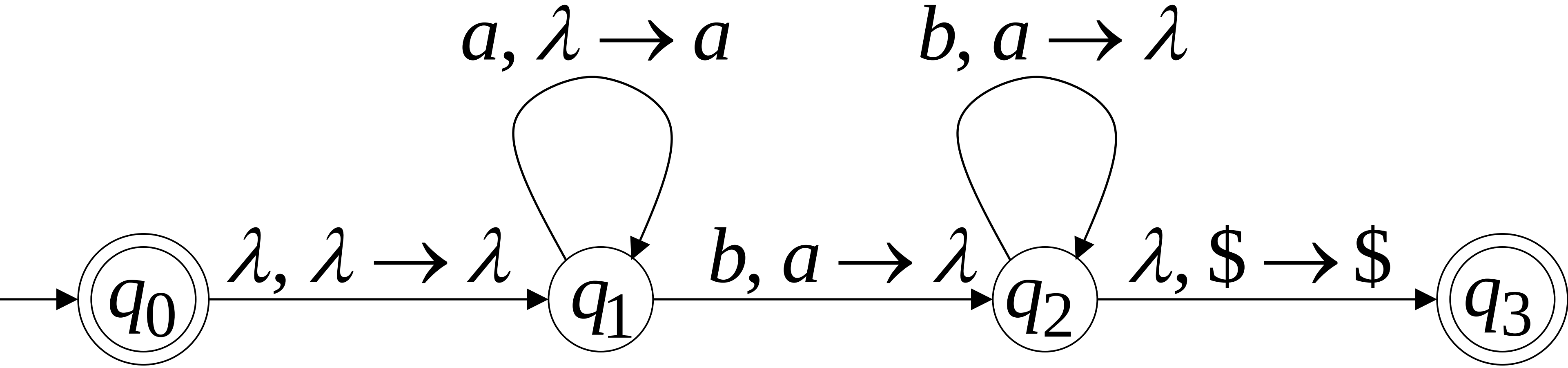
PDA M:



$$(q_0, a^n b^n, \$) \overset{*}{\succ} (q_3, \lambda, \$)$$

$$a^n b^n \in L(M)$$

PDA M :



Therefore: $L(M) = \{a^n b^n : n \geq 0\}$

PDA M :