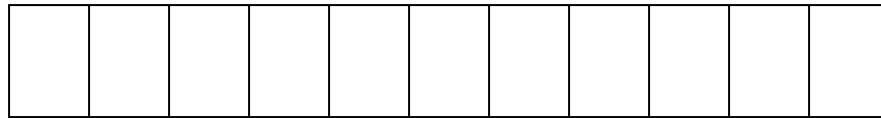


Pushdown Automata

PDA's

Pushdown Automaton -- PDA

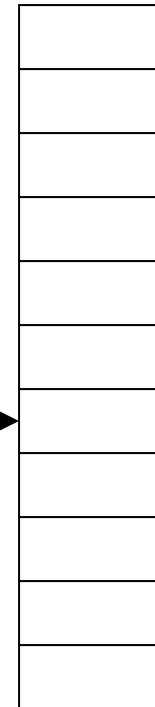
Input String



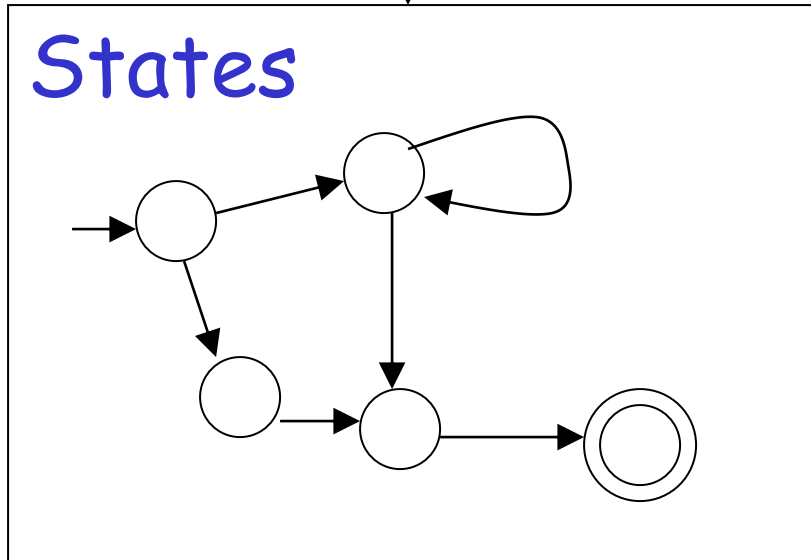
temporary memory

หน่วยความจำชั่วคราว

Stack

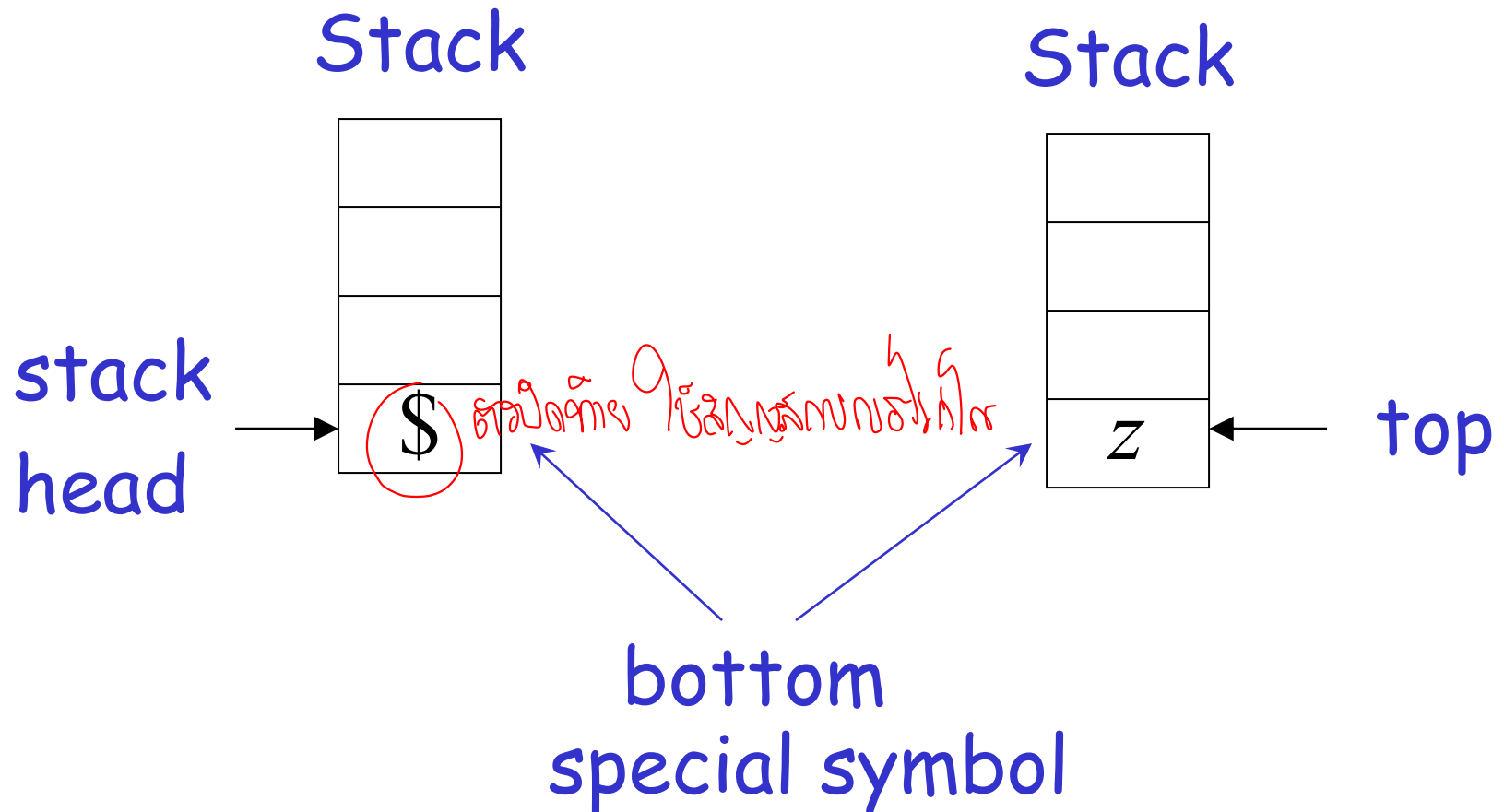


States



ไม่ว่าจะ string บน stack หรือ state ขึ้น

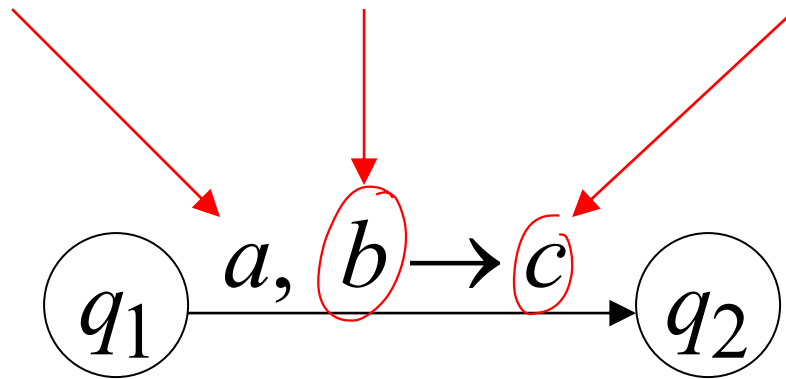
Initial Stack Symbol

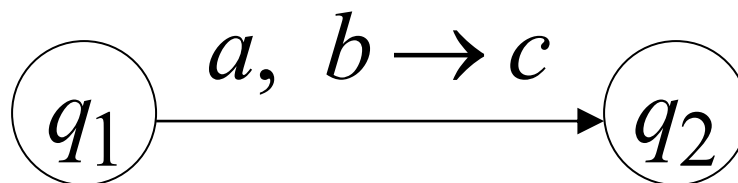


The States

transition 100 push down a b t o m a n

Input symbol Pop ^① symbol ^② Push symbol

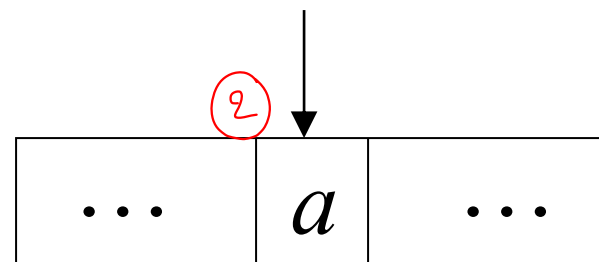
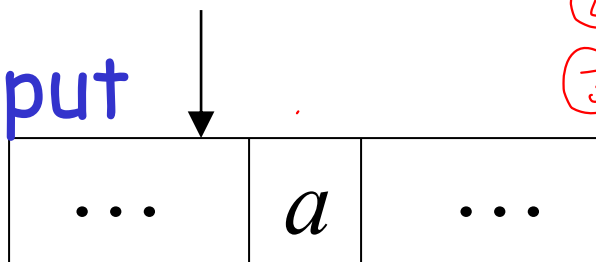




①

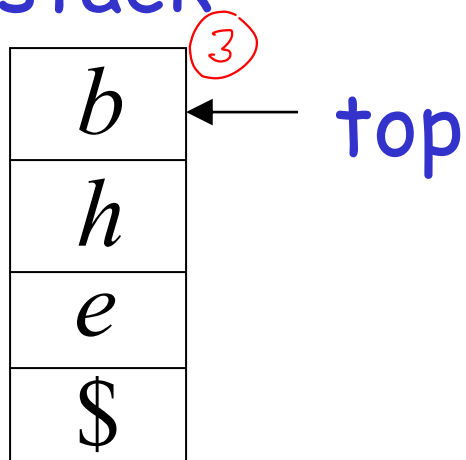
① word state oslug
② input oslug
③ top stack oslug

input

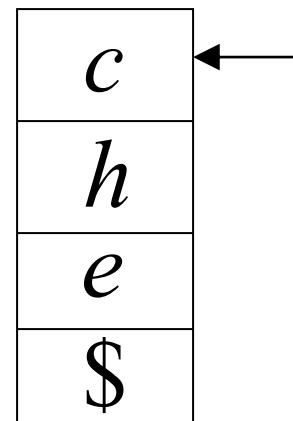
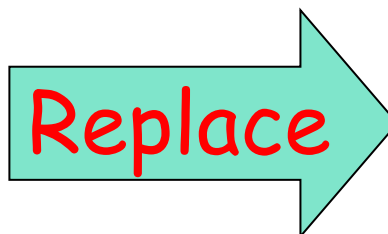


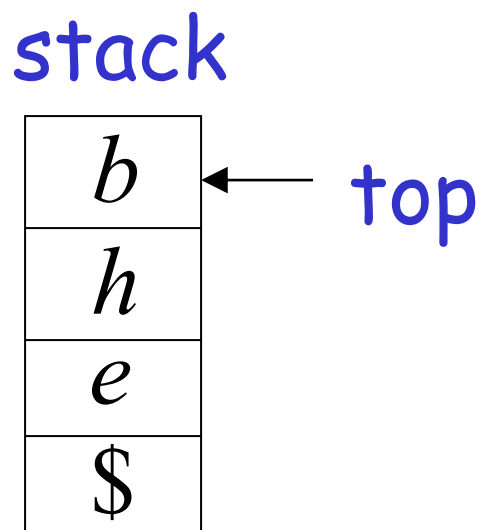
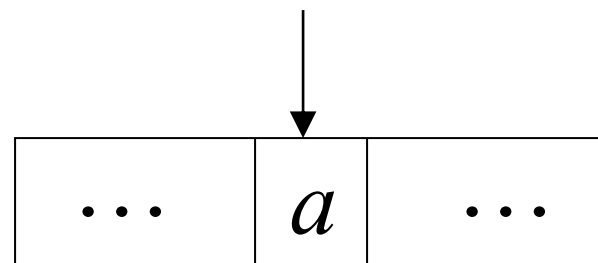
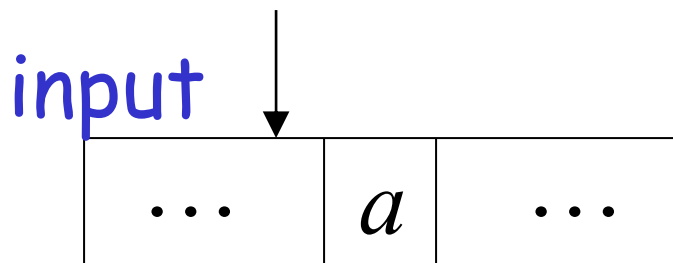
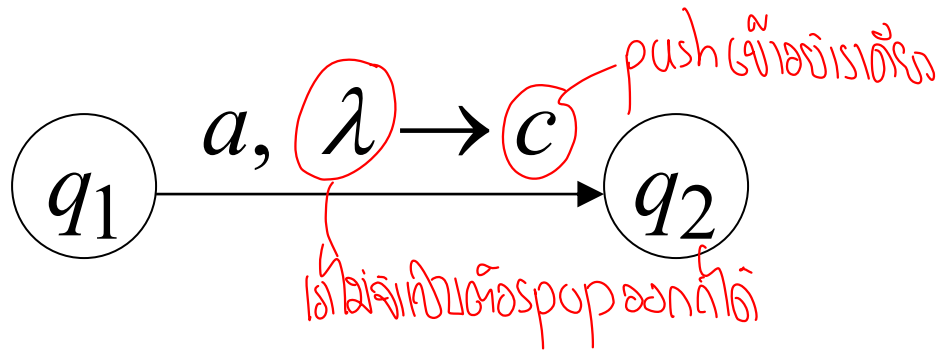
②

stack

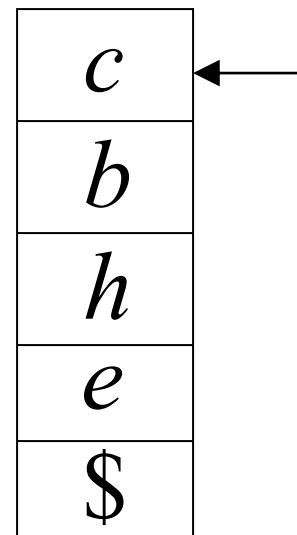


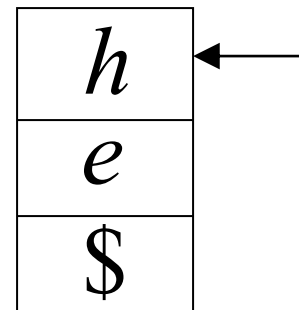
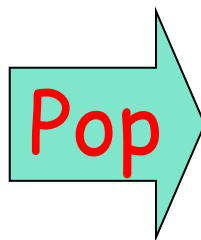
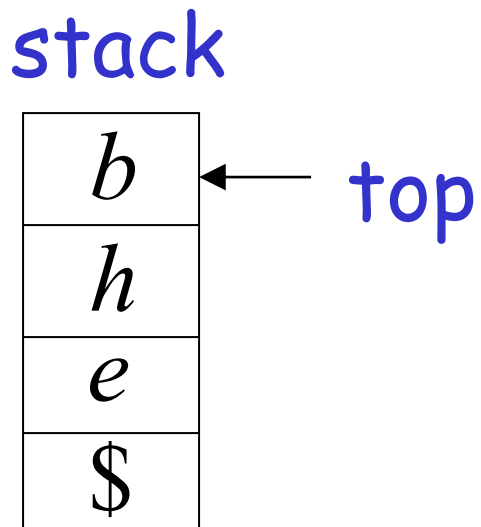
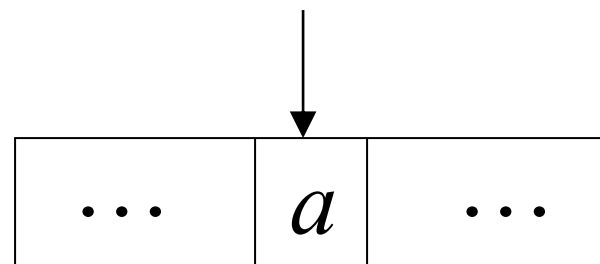
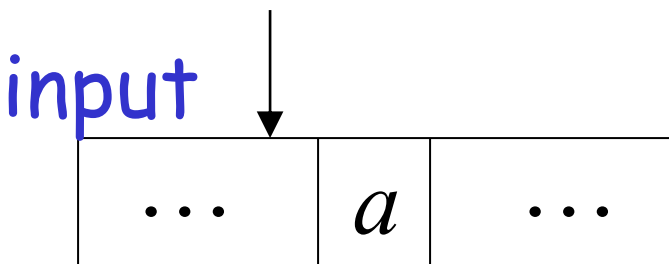
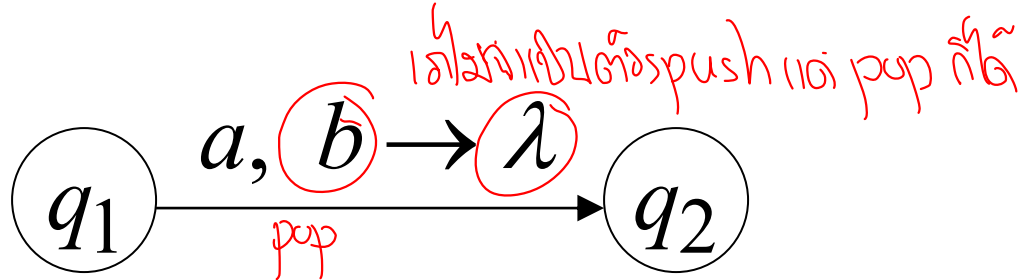
③

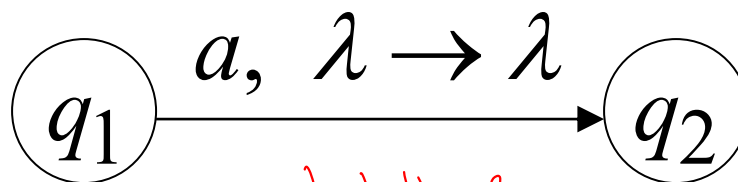




Push





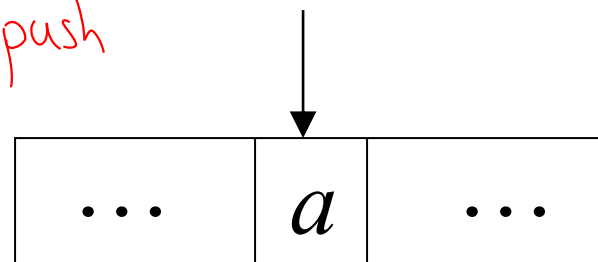
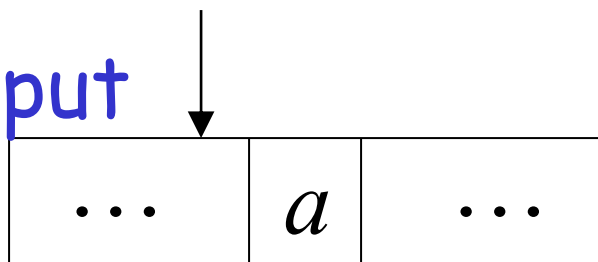


what this?

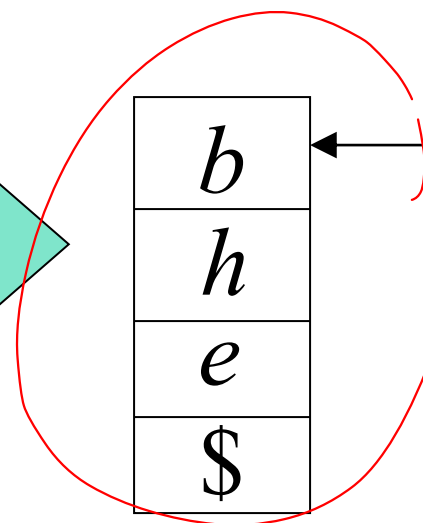
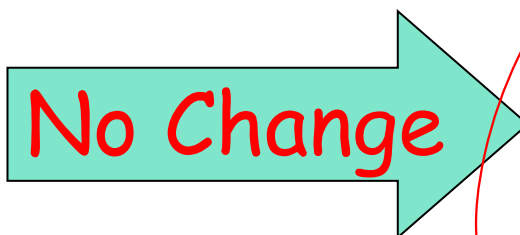
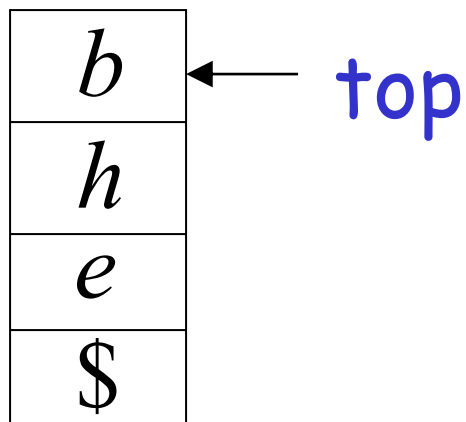
I
no pop, no push

↓
stack never change

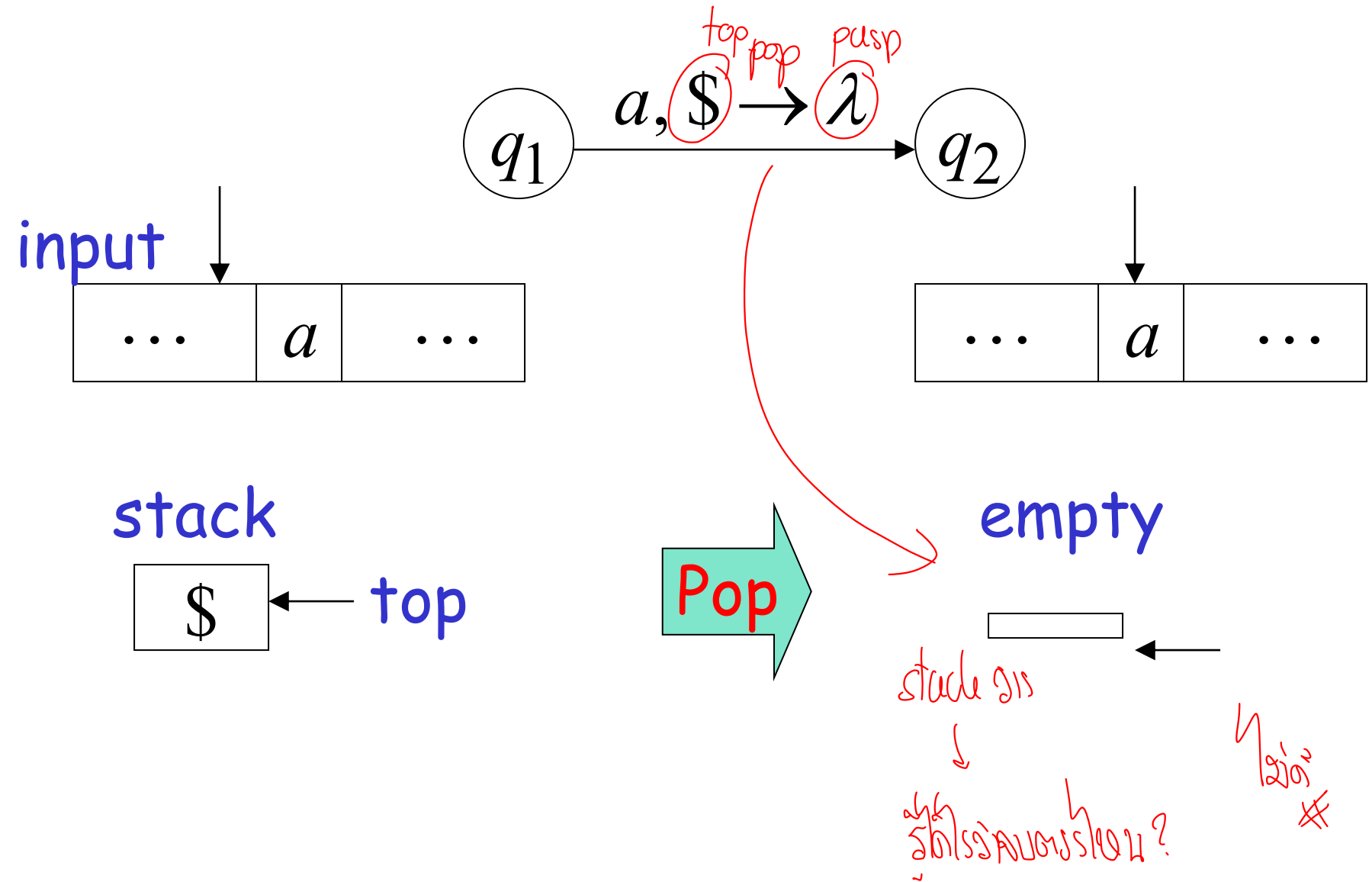
input



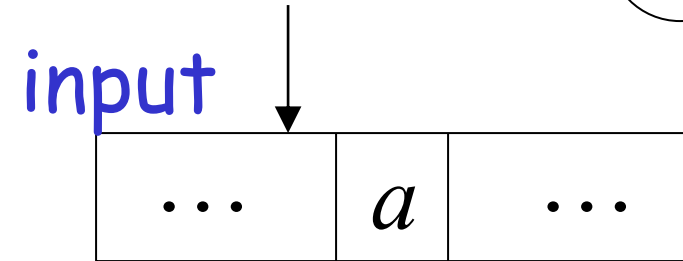
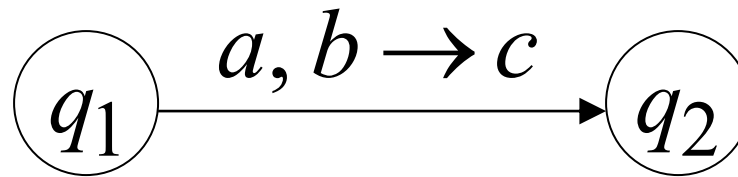
stack



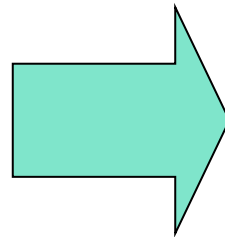
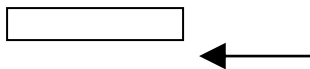
A Possible Transition



A Bad Transition



Empty stack

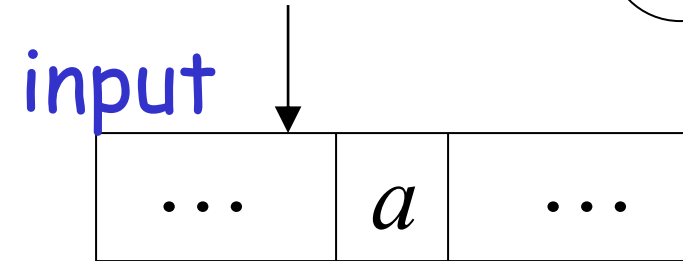
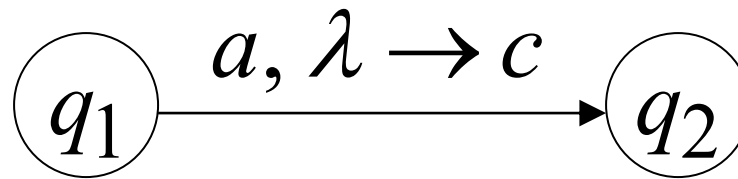


HALT

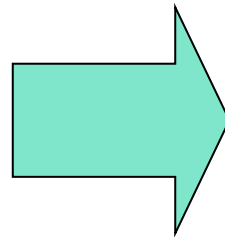
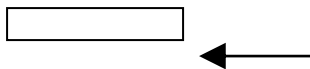
The automaton **Halts** in state q_1 and **Rejects** the input string

reject \rightarrow reject
input string
ปฏิเสธ
สถานะ final state แล้ว

A Bad Transition



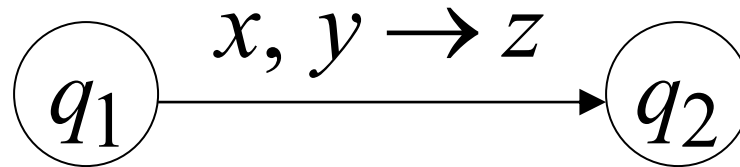
Empty stack



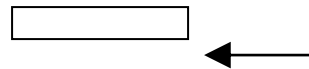
HALT

The automaton **Halts** in state q_1
and **Rejects** the input string

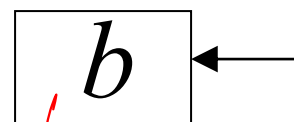
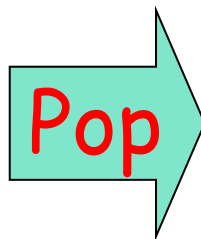
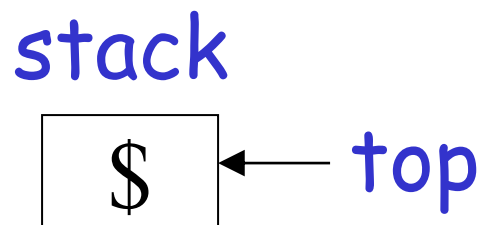
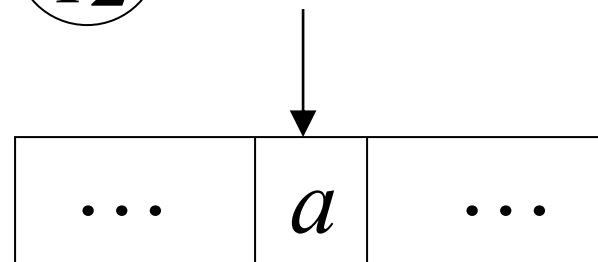
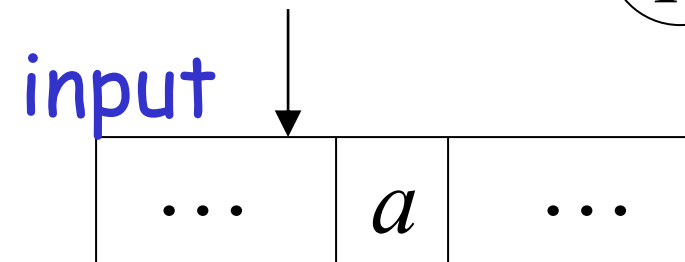
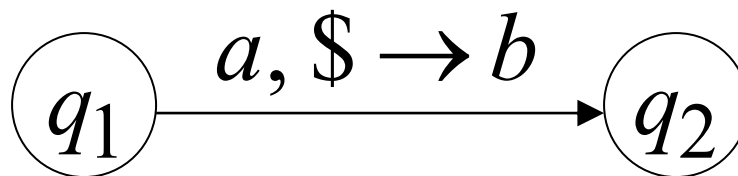
No transition is allowed to be followed
When the stack is empty



Empty stack



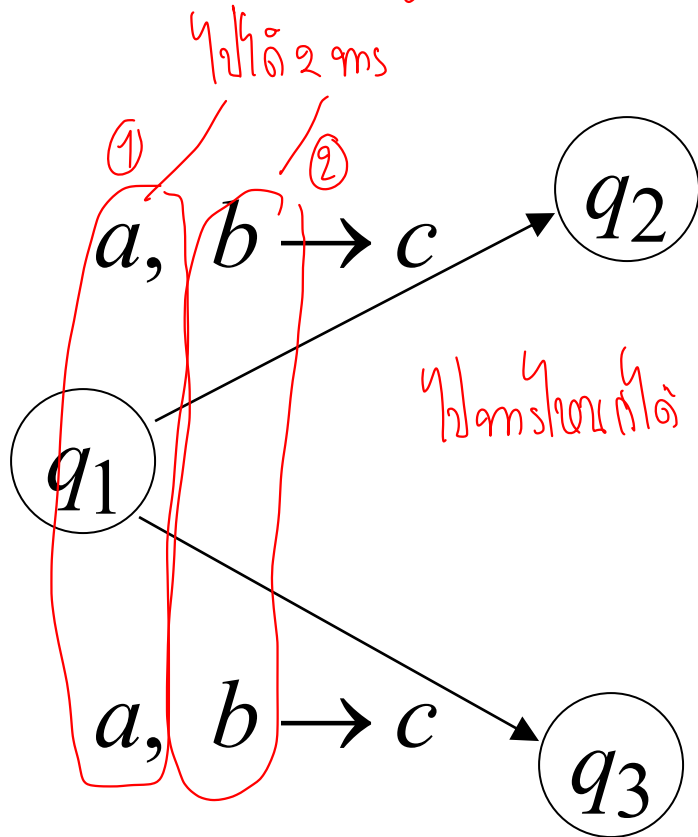
A Good Transition



dollar sign បាត់បង់

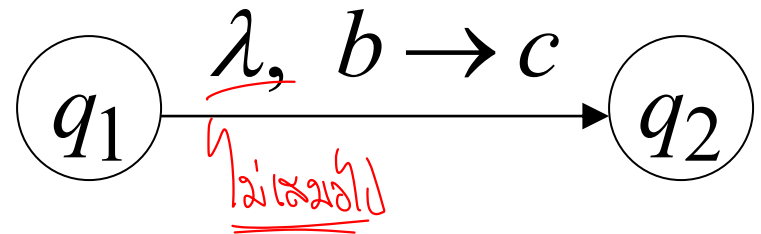
ឥឡូវ អ្នក កំពុង ប្រើប្រាស់ ប្រព័ន្ធ

Non-Determinism



Handwritten notes in Thai:

- non FA (non FA)
- non deterministic (non deterministic)
- NPDA (NPDA) - pointing to the text below.
- NPDA (NPDA) - pointing to the text below.
- NPDA (NPDA) - pointing to the text below.

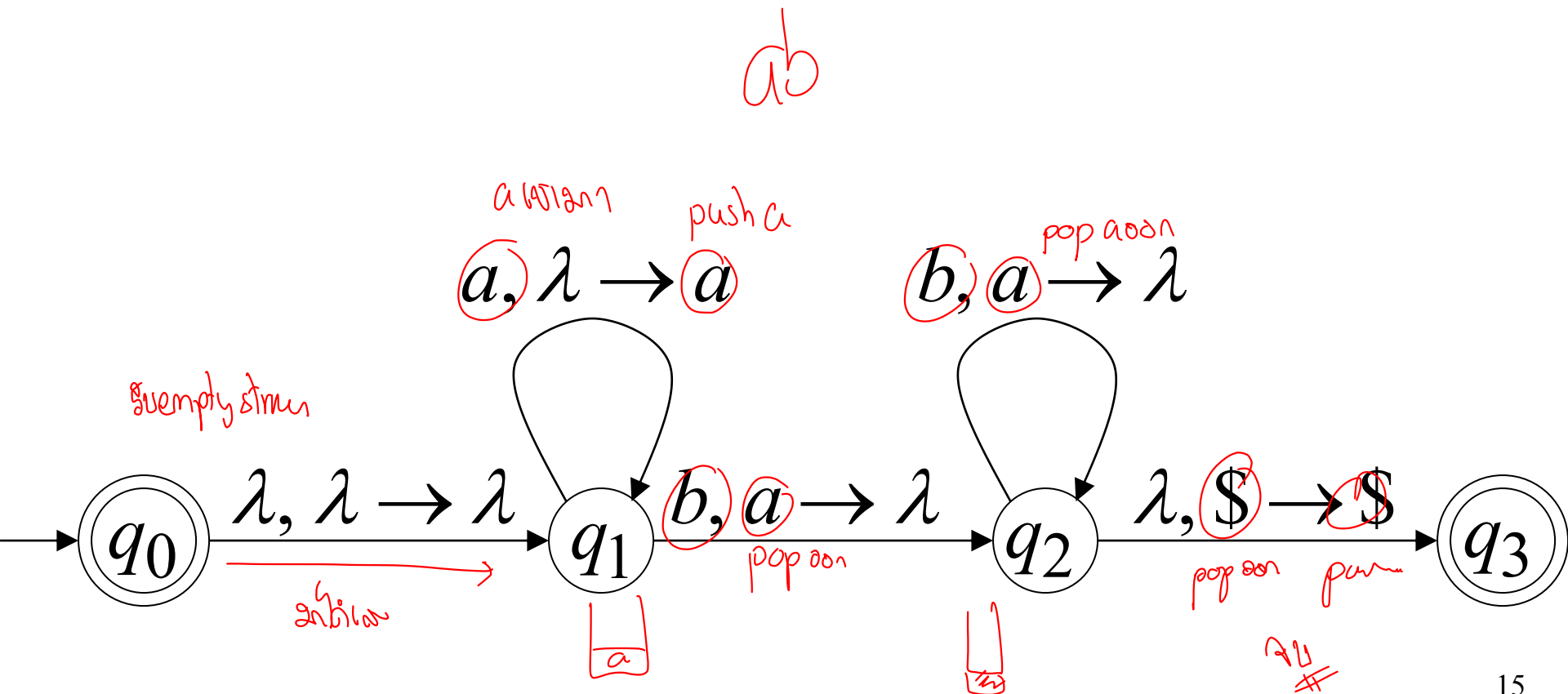


λ - transition

These are allowed transitions in a
Non-deterministic PDA (NPDA)

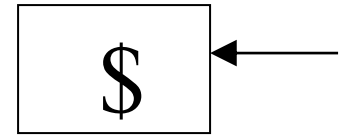
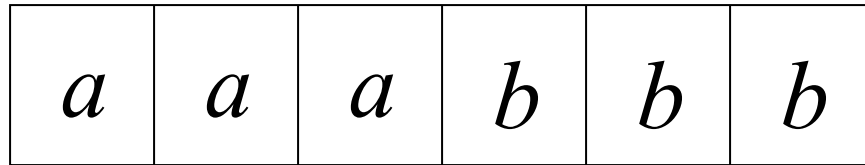
NPDA: Non-Deterministic PDA

Example:

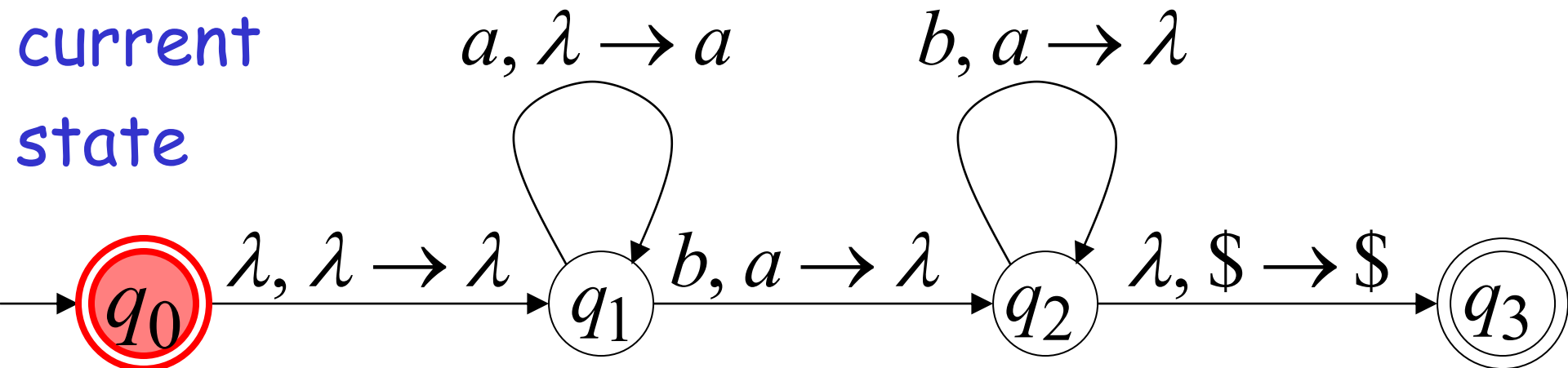


Execution Example: Time 0

Input

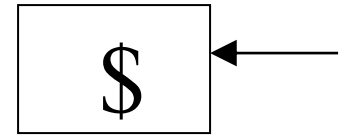
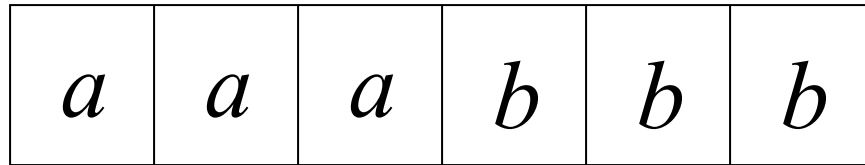


Stack

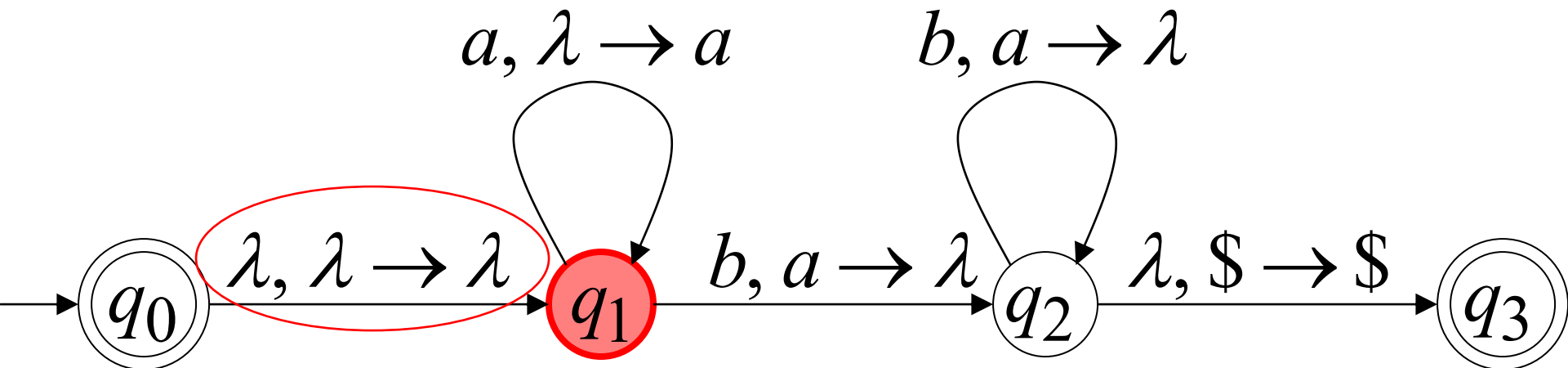


Time 1

Input

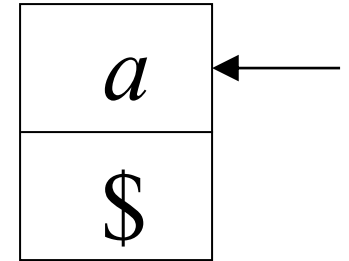
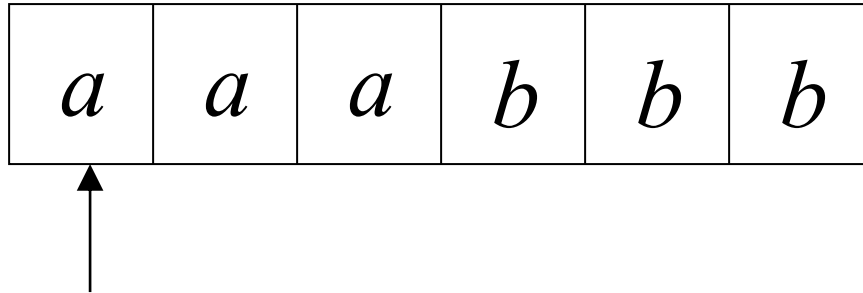


Stack

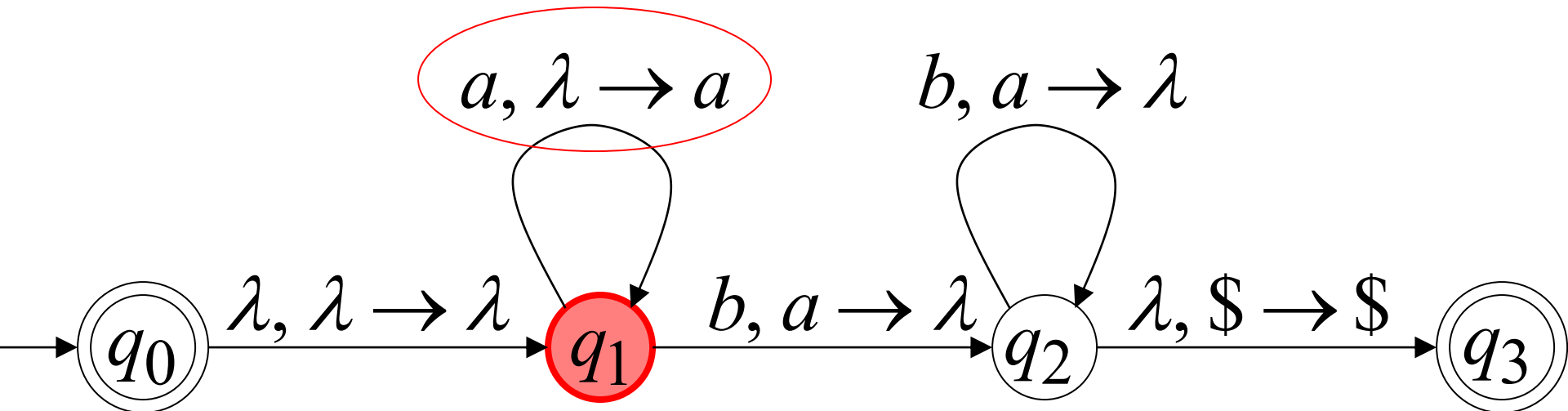


Time 2

Input

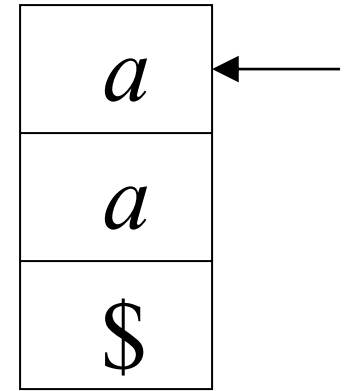
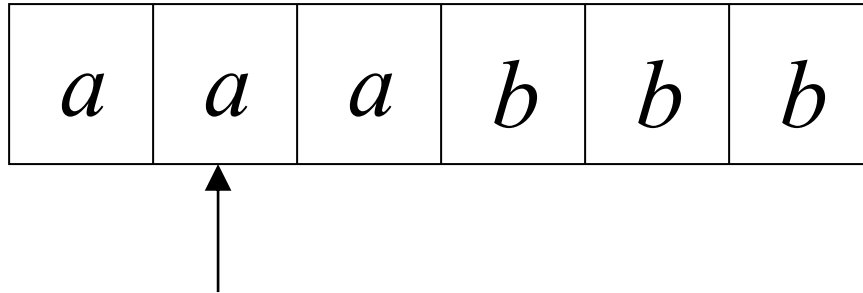


Stack

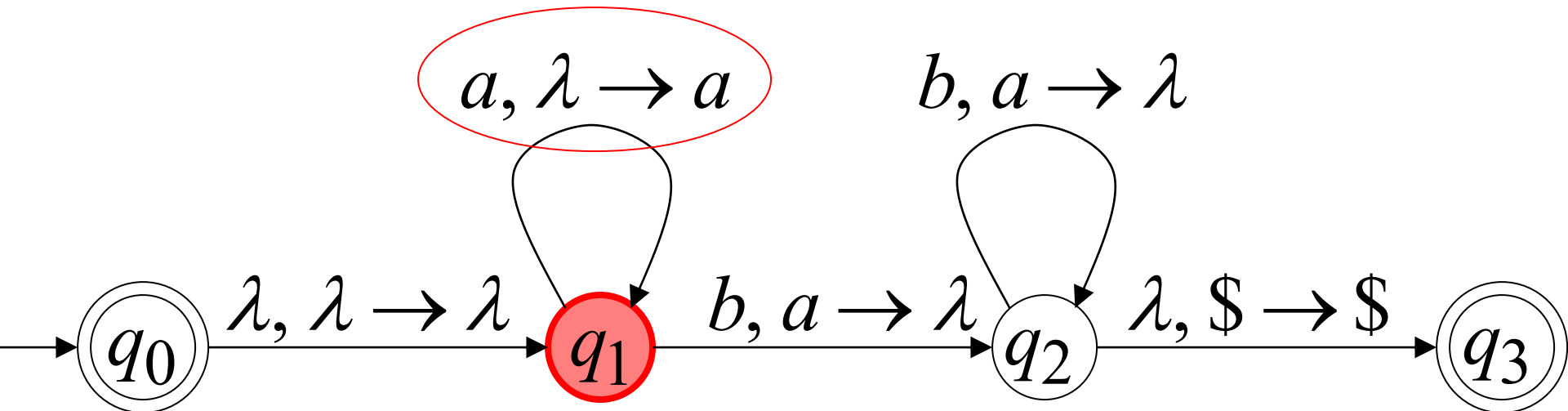


Time 3

Input

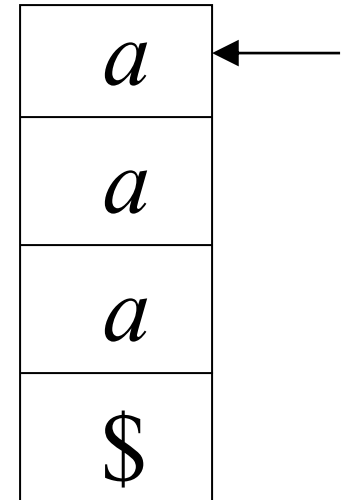
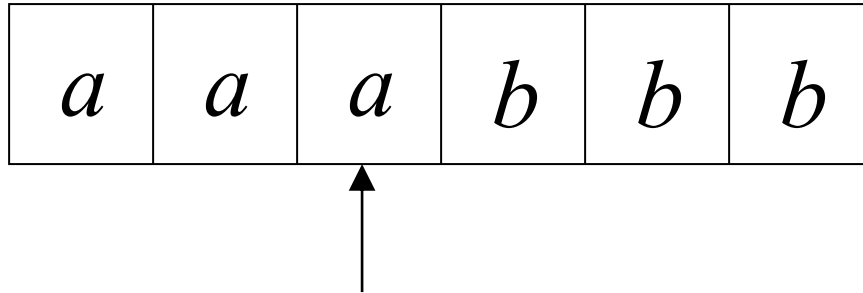


Stack

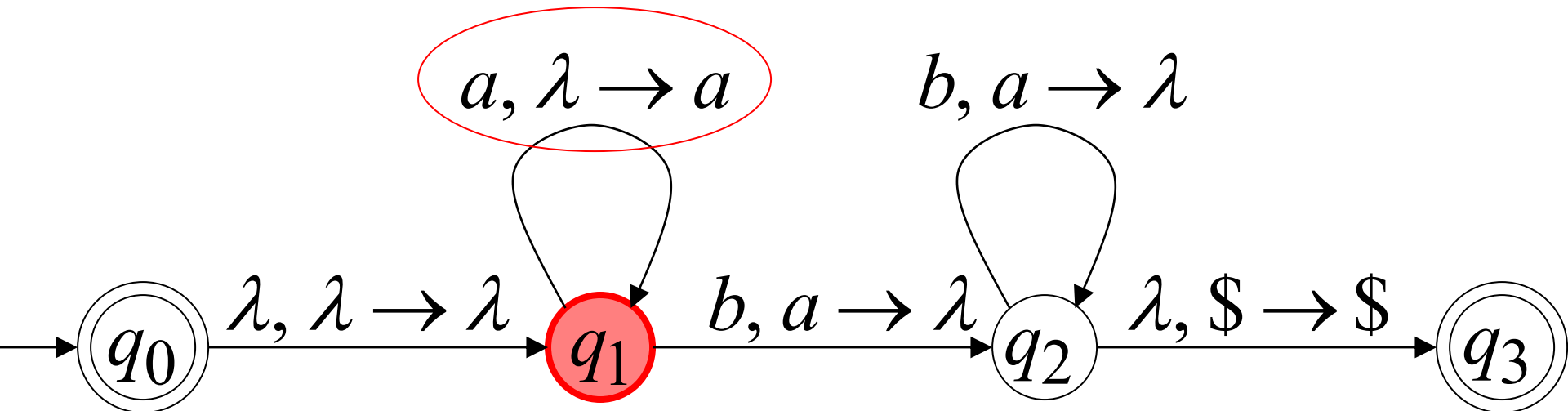


Time 4

Input

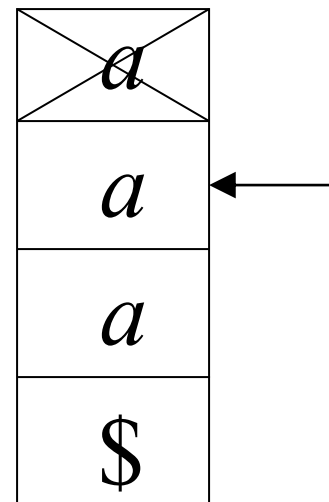
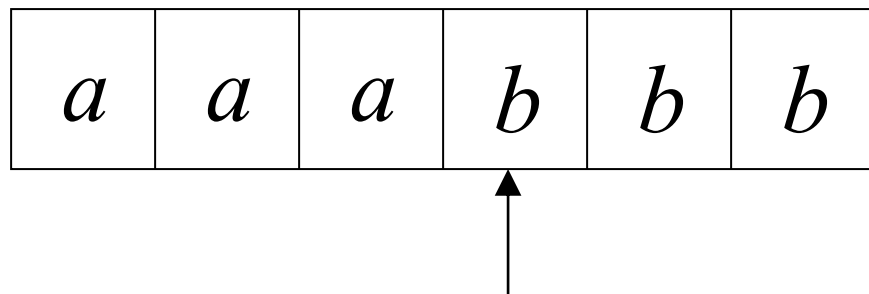


Stack

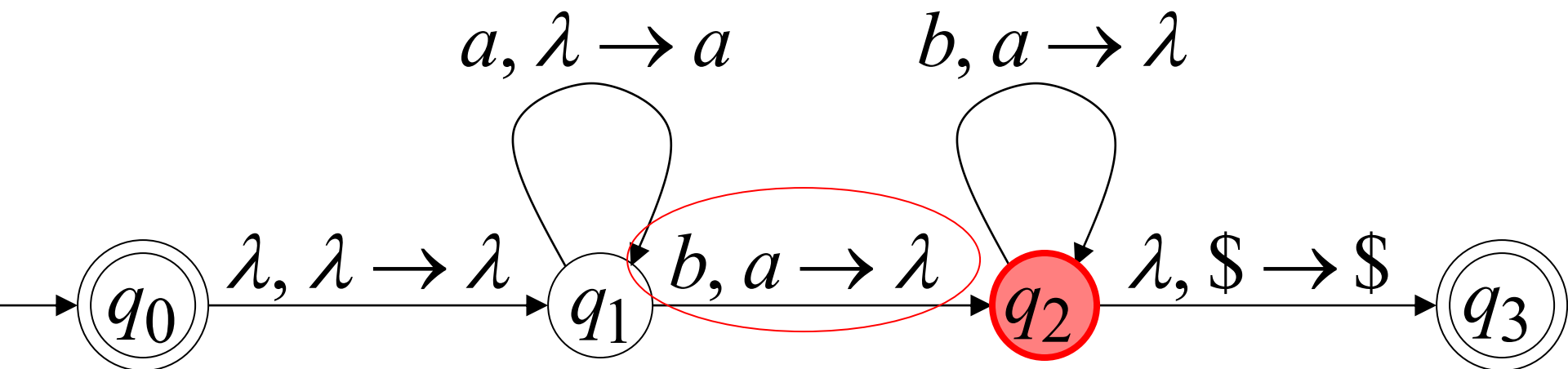


Time 5

Input

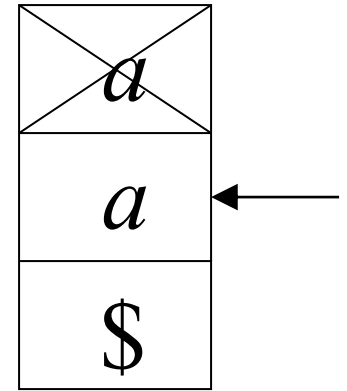
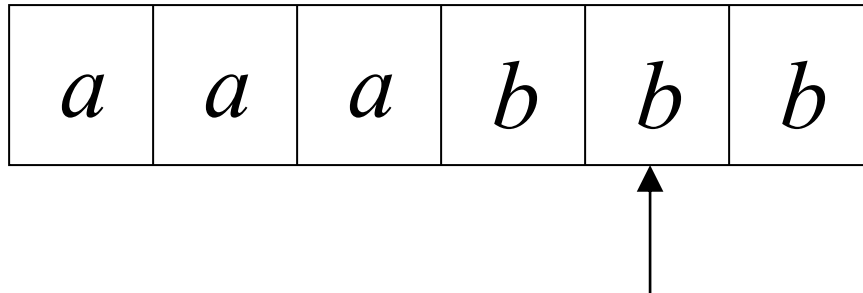


Stack

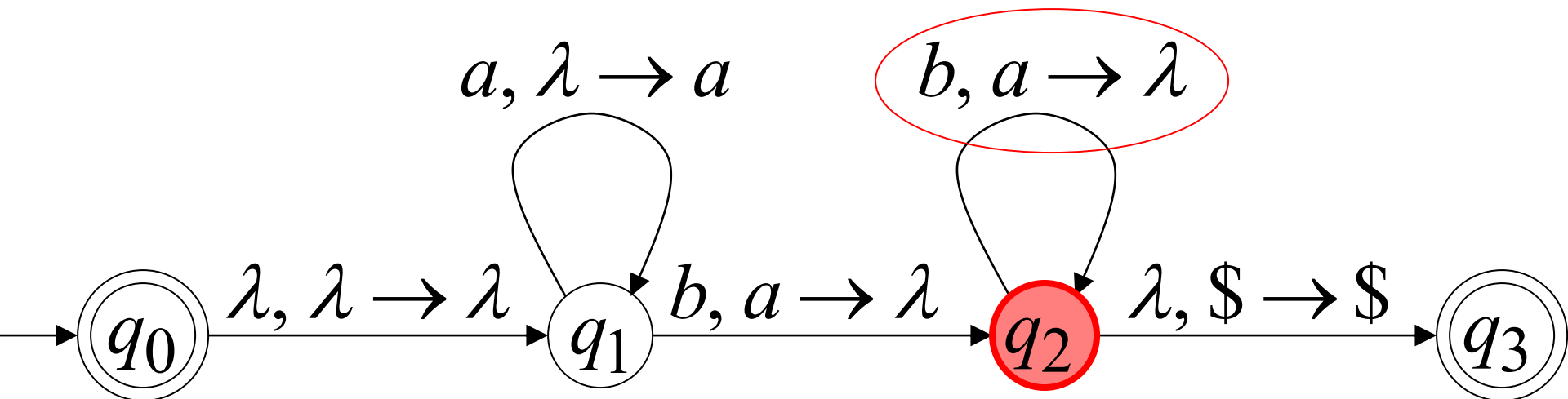


Time 6

Input

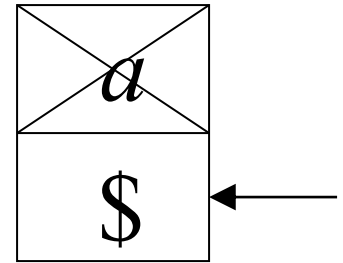
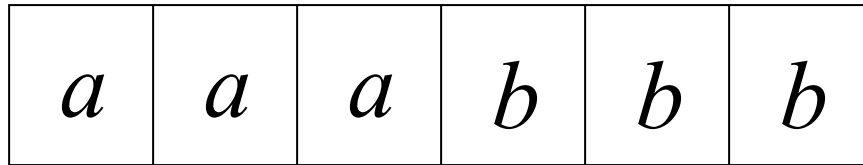


Stack

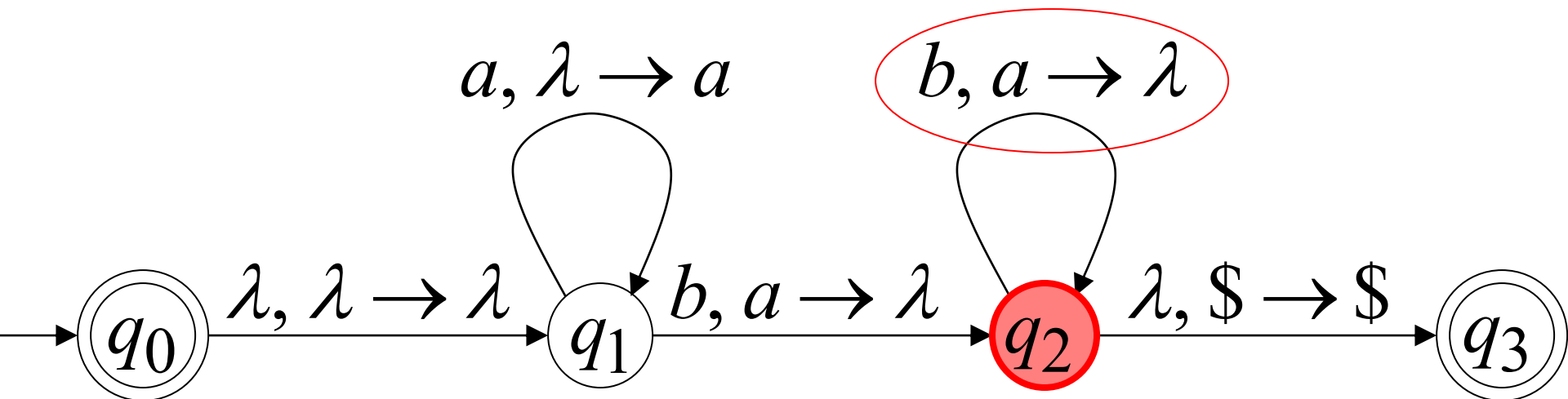


Time 7

Input

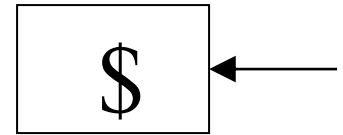
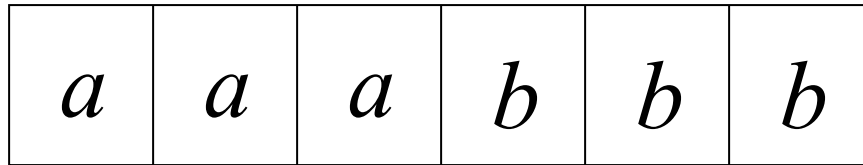


Stack

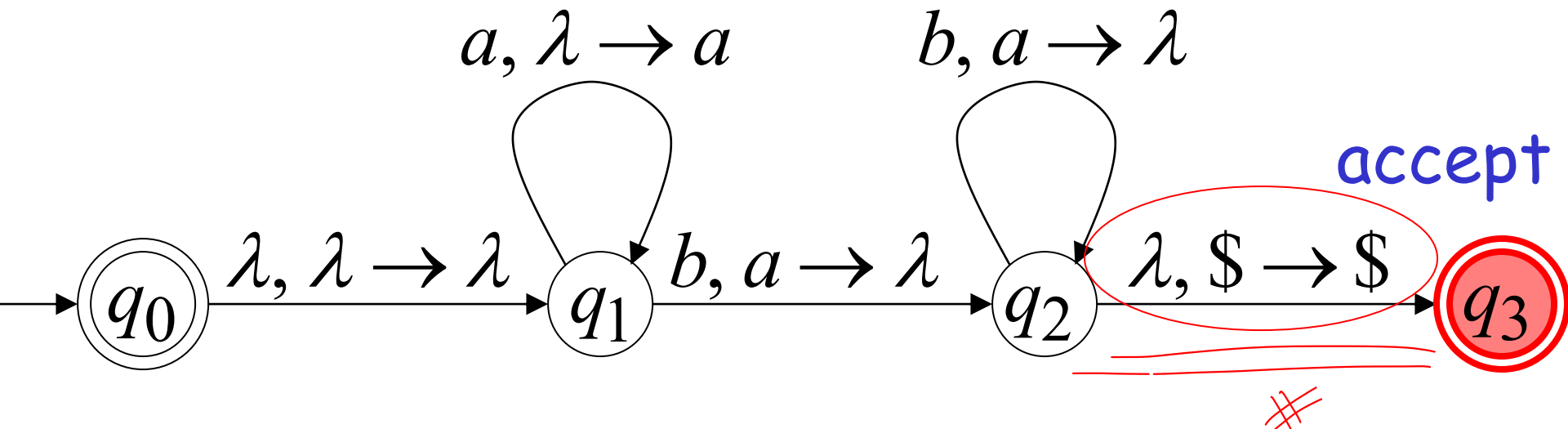


Time 8

Input



Stack



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

↳ all input is consumed & state is final

↳ No symbols left in stack \rightarrow (empty stack)

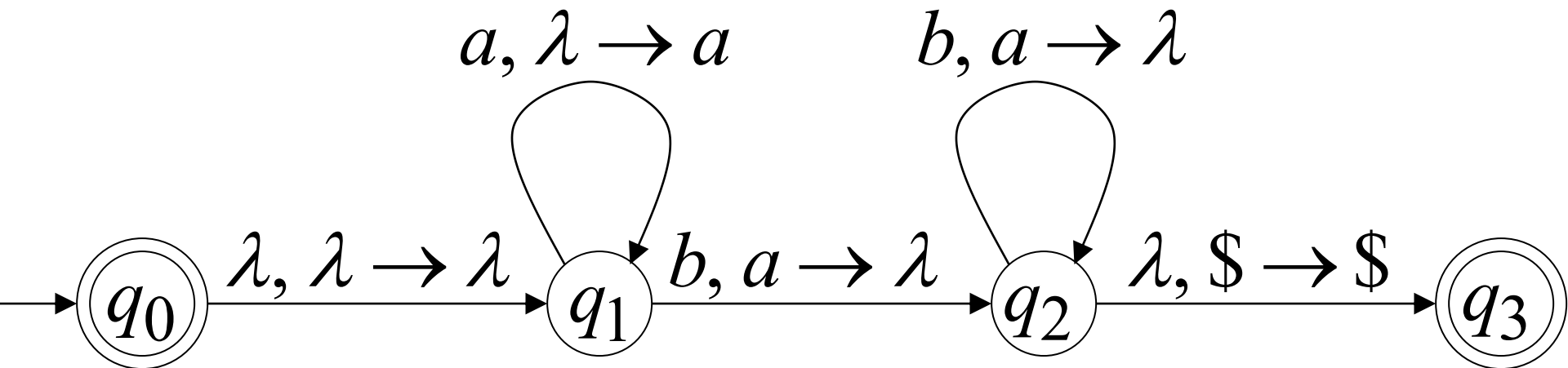
All the input is consumed

AND

The last state is a final state

At the end of the computation,
we do not care about the stack contents

The input string *aaabbb*
is accepted by the NPDA:

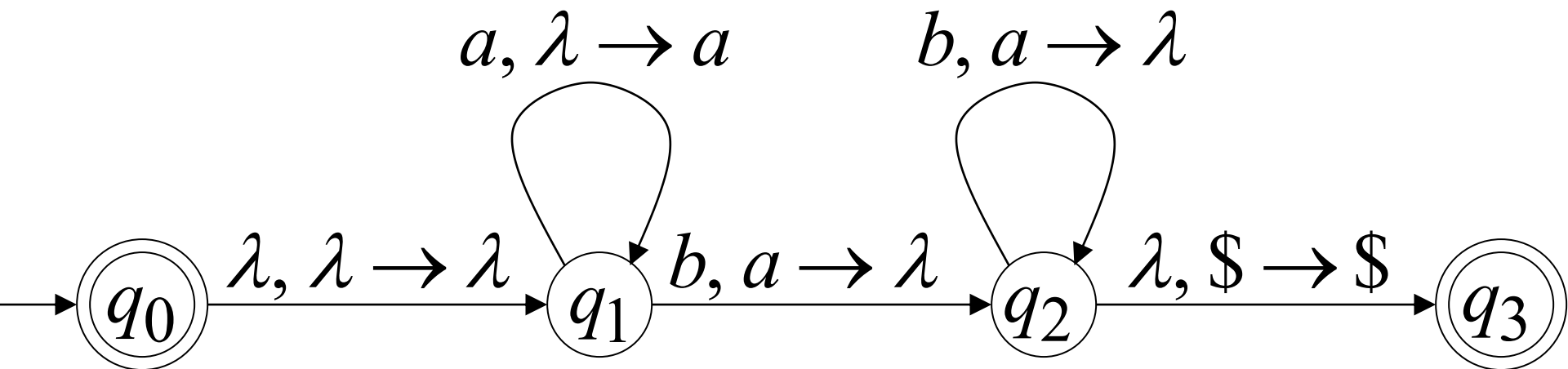


In general,

ଉଦାହରଣ

$$L = \{a^n b^n : \underline{n \geq 0}\}$$

is the language accepted by the NPDA:

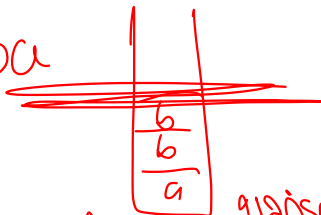


Another NPDA example

NPDA M

ตัวอย่างของ NPDA ซึ่งใช้อัลกอริทึมที่ recognize strings ww^R เป็น regular language

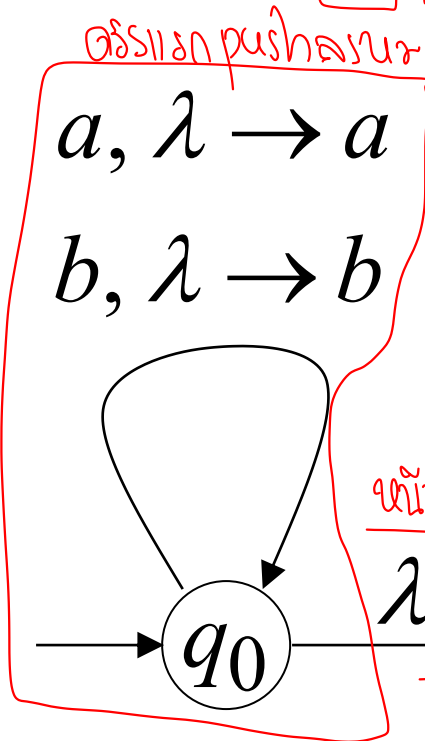
abb bba



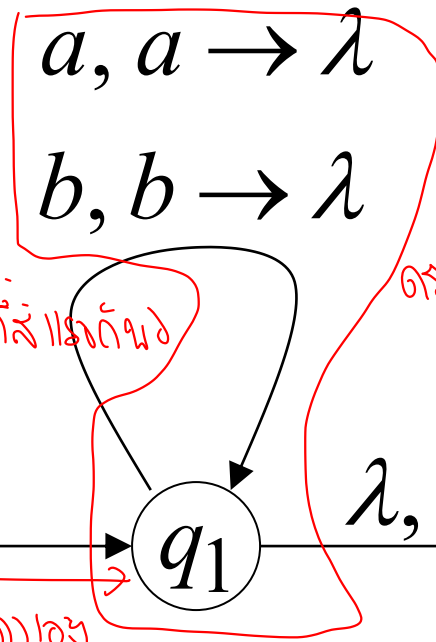
$$L(M) = \{ww^R\}$$

ทั้งนี้ \rightarrow ① language ใด
ใน NPDA

② ① NPDA ใด
ใน language



NPDA จะใช้วิธี
ใส่ทั้ง w และ w^R
ลงบน stack



การลบ pop ออกจากราวตัว

↓
การที่ w และ w^R
เข้ากันพอดี (match)

$\lambda, \lambda \rightarrow \lambda$

NPDA เป็น automaton

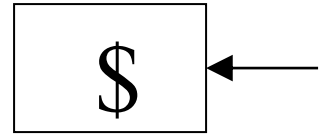
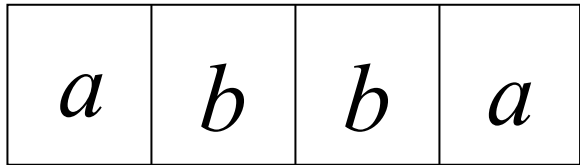
$\lambda, \$ \rightarrow \$$

การ input ว่างลงแล้ว $\$$



Execution Example: Time 0

Input



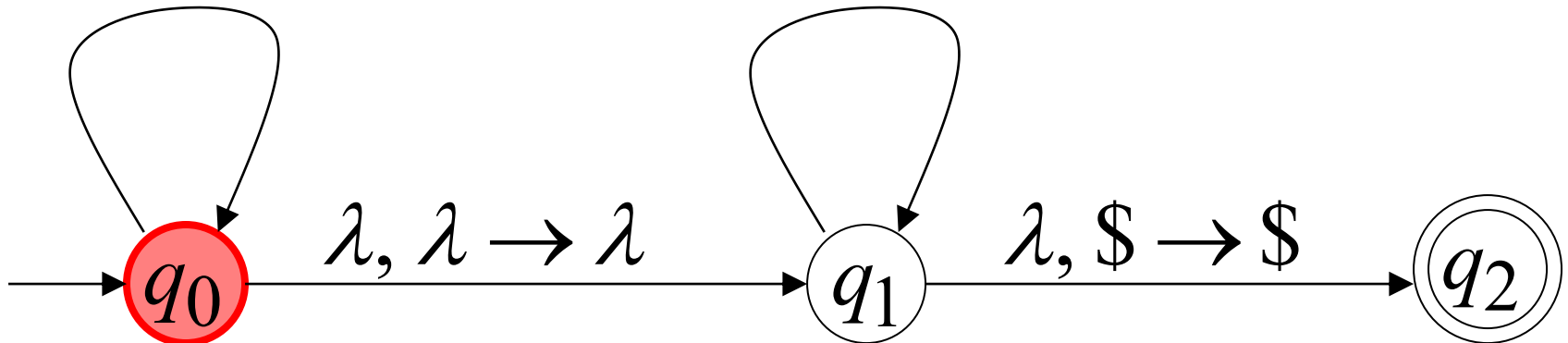
Stack

$a, \lambda \rightarrow a$

$a, a \rightarrow \lambda$

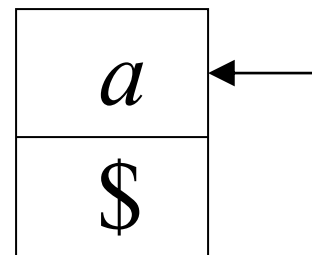
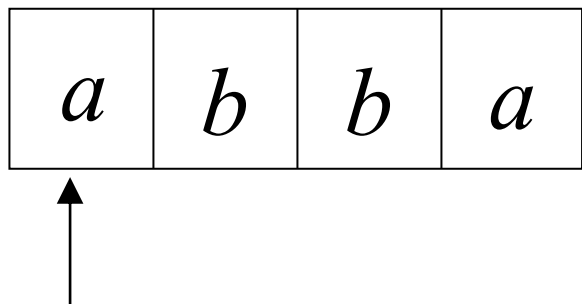
$b, \lambda \rightarrow b$

$b, b \rightarrow \lambda$



Time 1

Input



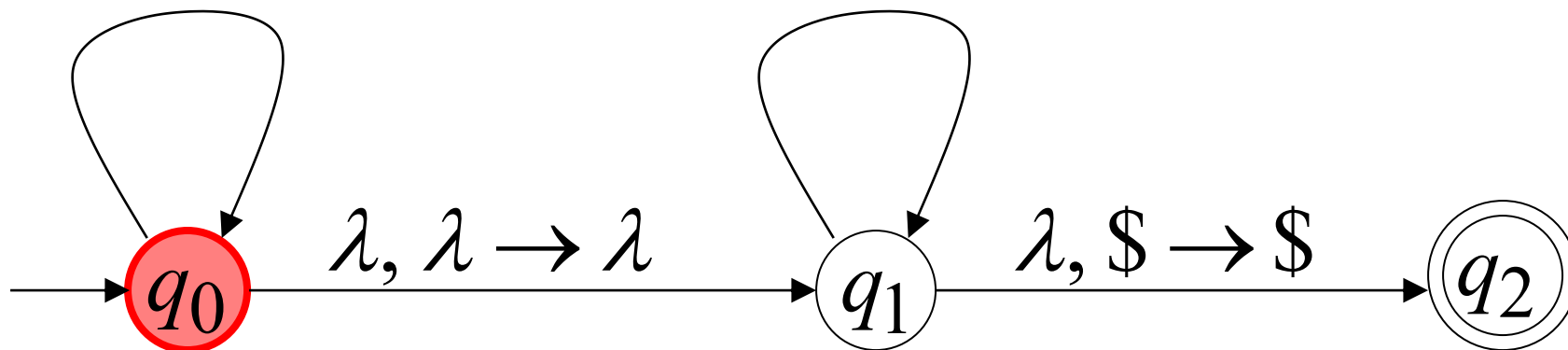
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

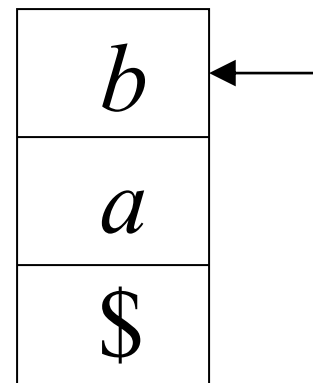
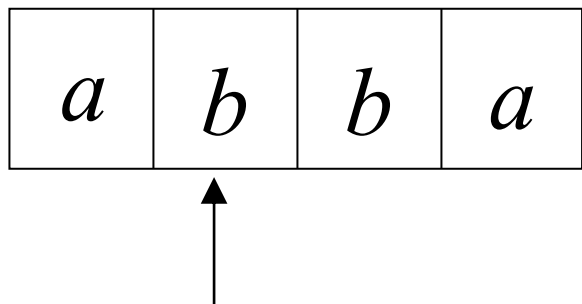
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$



Time 2

Input



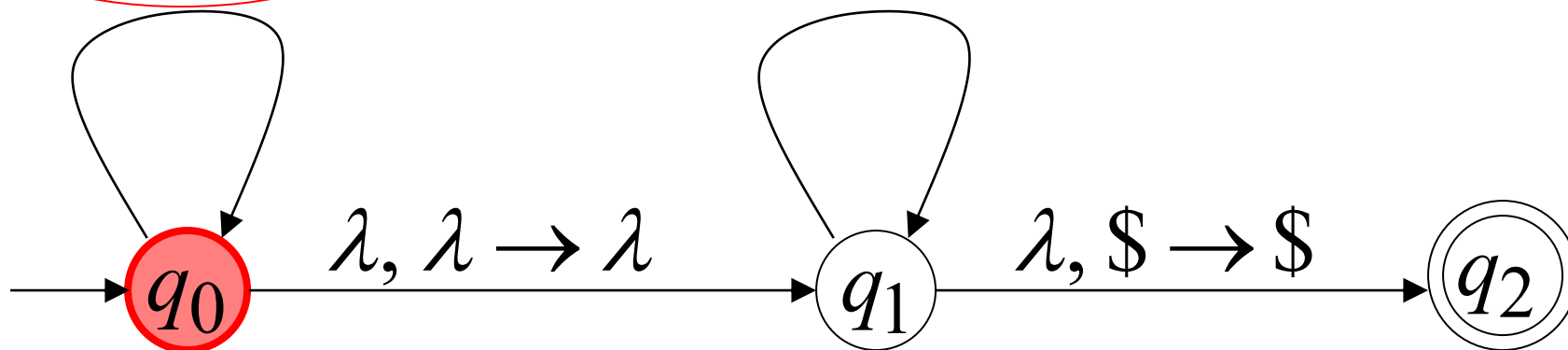
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

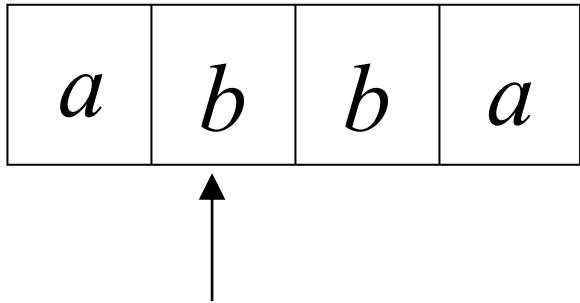
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$

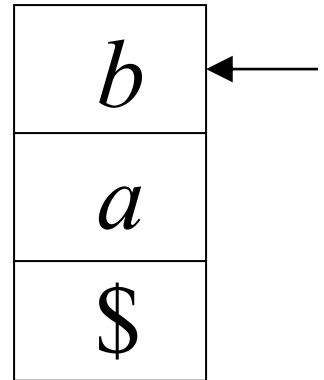


Time 3

Input



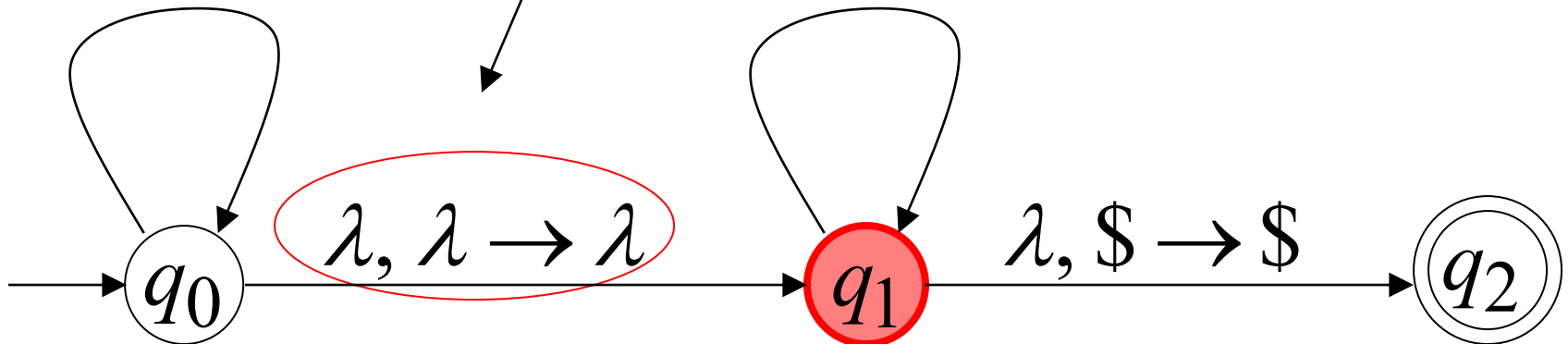
Guess the middle
of string



Stack

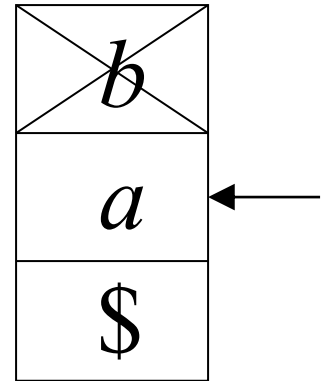
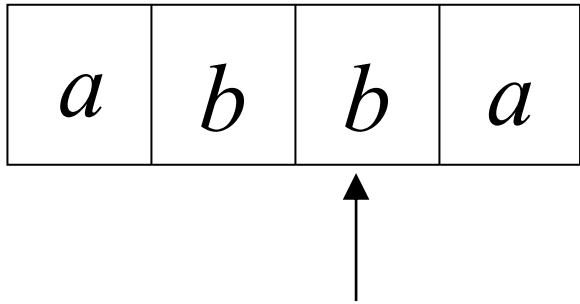
$a, \lambda \rightarrow a$
 $b, \lambda \rightarrow b$

$a, a \rightarrow \lambda$
 $b, b \rightarrow \lambda$



Time 4

Input



Stack

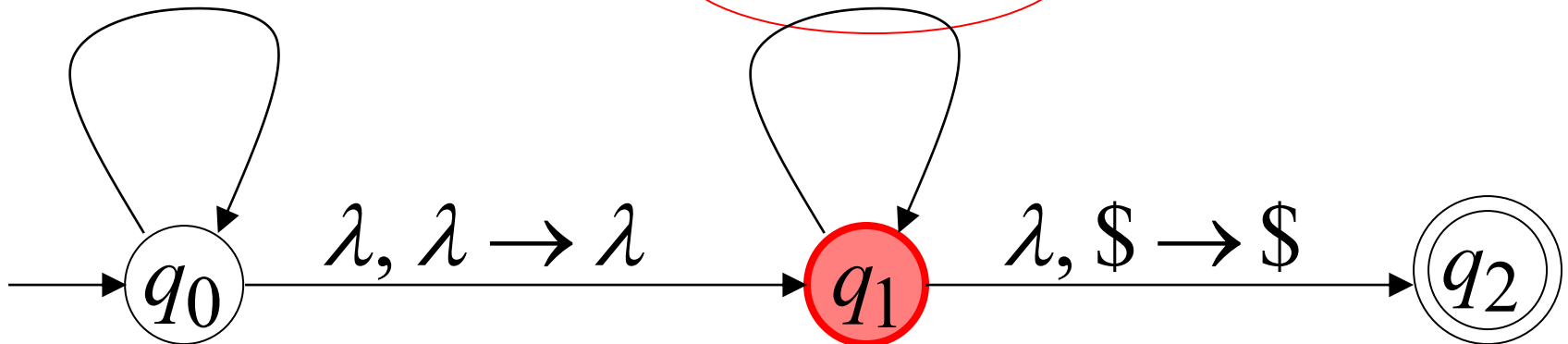
$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

$a, a \rightarrow \lambda$

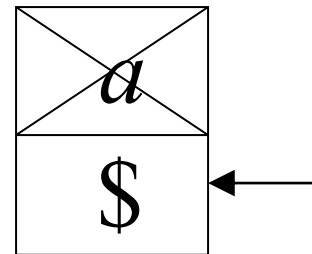
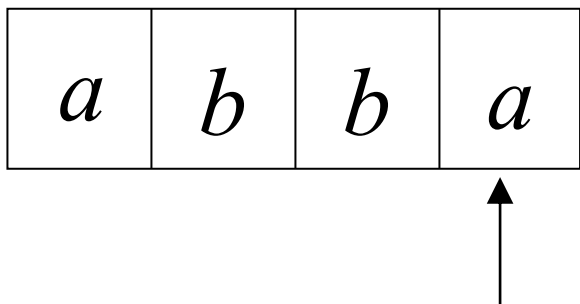
$b, b \rightarrow \lambda$

Handwritten notes in red:
- $b, b \rightarrow \lambda$
- $b, b \rightarrow \lambda$ (with a tilde over the first b)
- \downarrow top $b \rightarrow \lambda$
- $b \rightarrow \lambda$ (with a tilde over the b)
- \downarrow string cleared



Time 5

Input



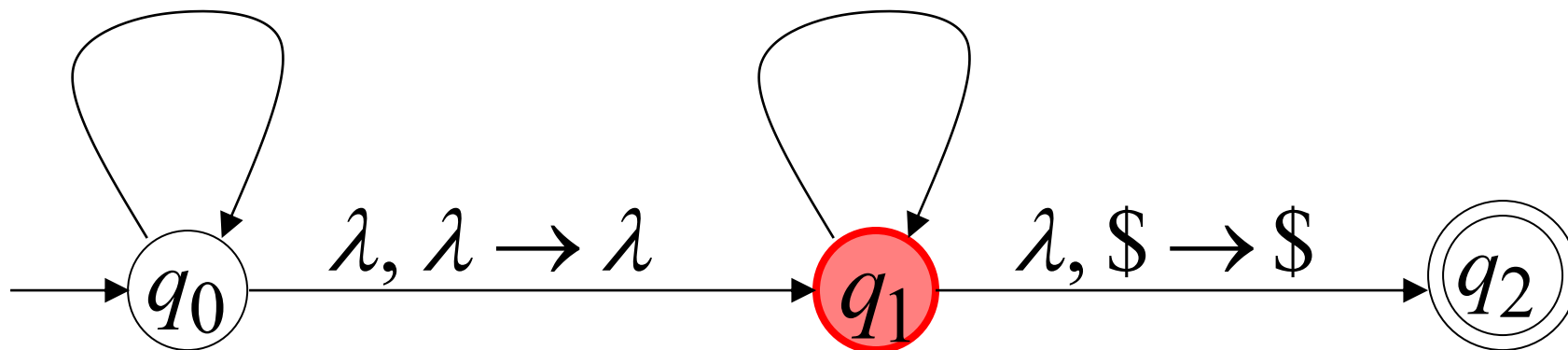
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

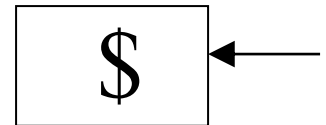
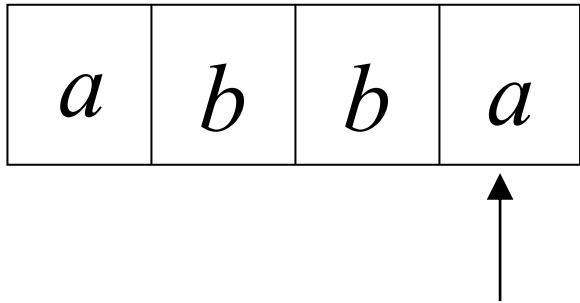
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$



Time 6

Input



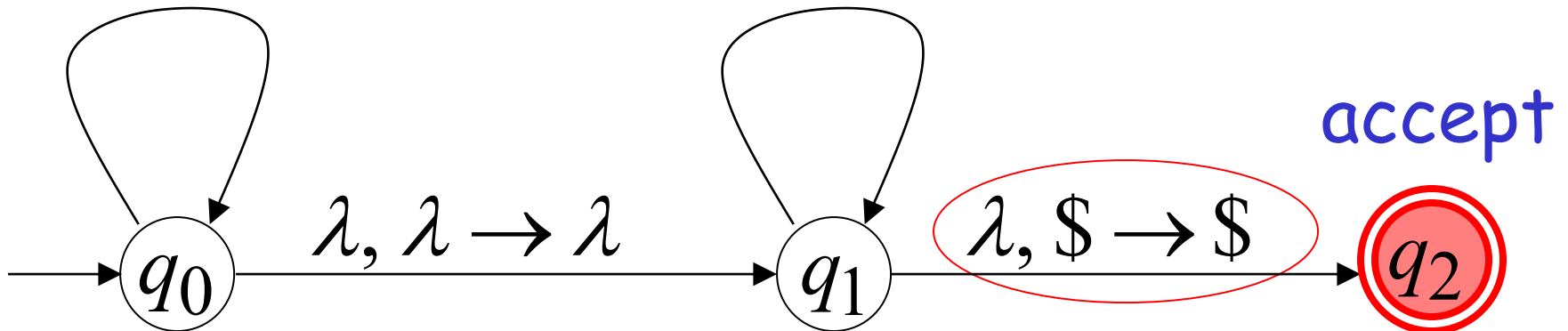
Stack

$a, \lambda \rightarrow a$

$a, a \rightarrow \lambda$

$b, \lambda \rightarrow b$

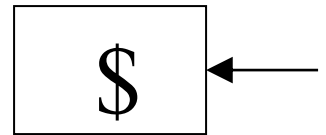
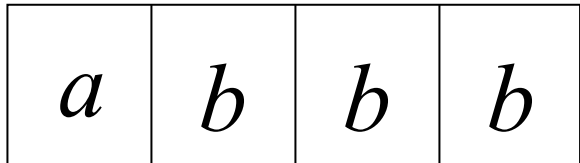
$b, b \rightarrow \lambda$



accept

Rejection Example: Time 0

Input



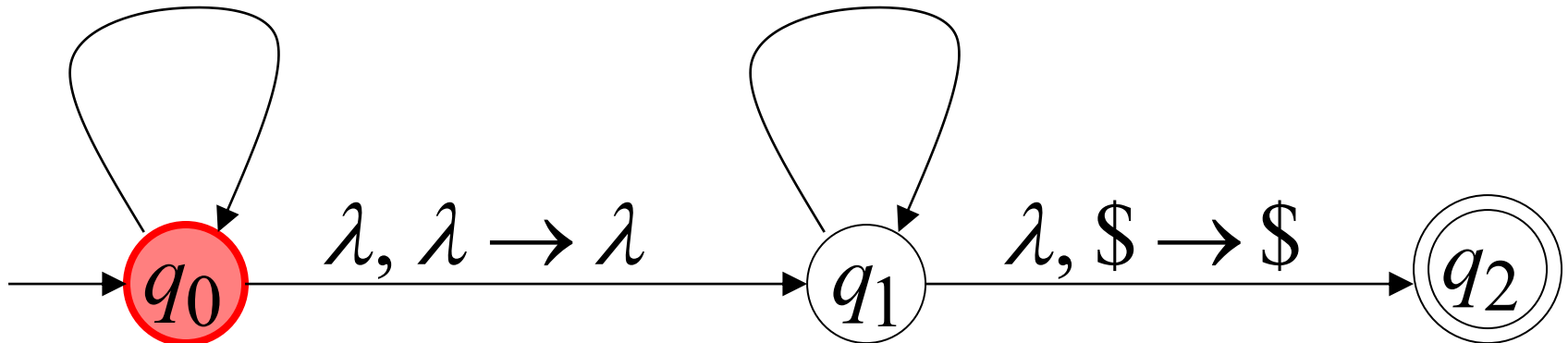
Stack

$a, \lambda \rightarrow a$

$a, a \rightarrow \lambda$

$b, \lambda \rightarrow b$

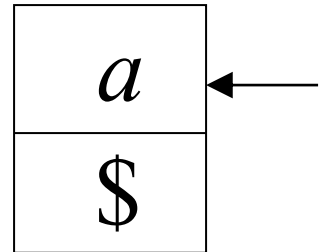
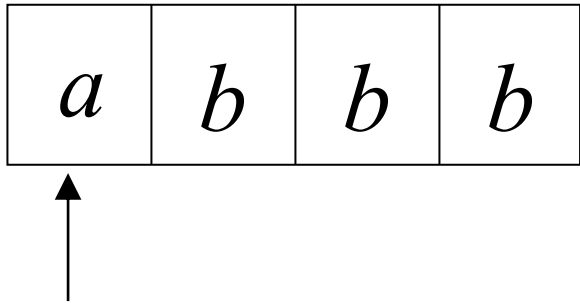
$b, b \rightarrow \lambda$



Time 1

not reject

Input



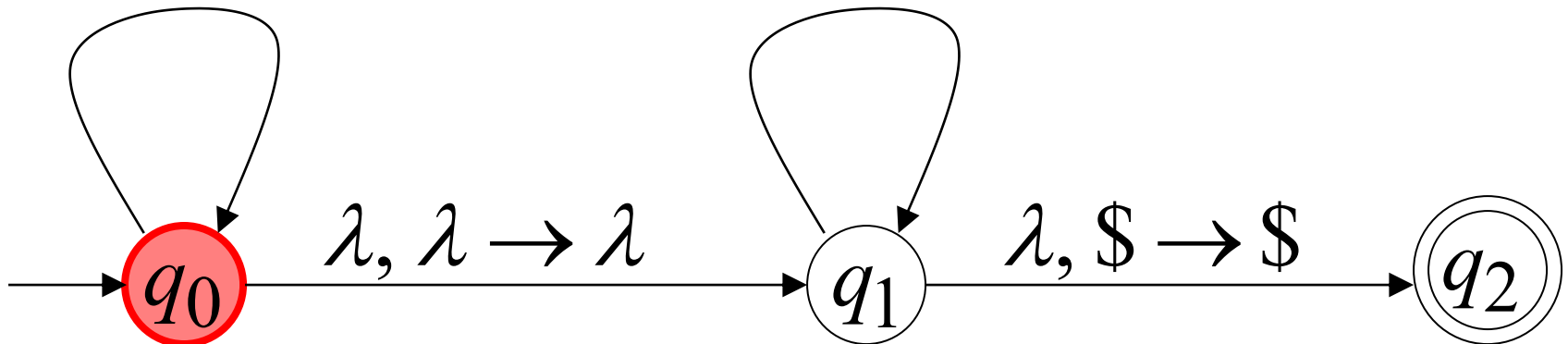
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

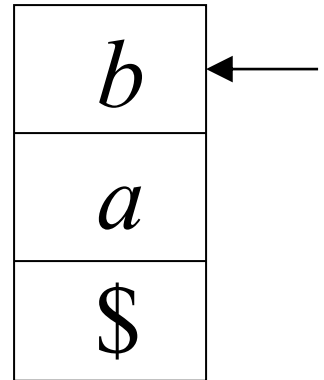
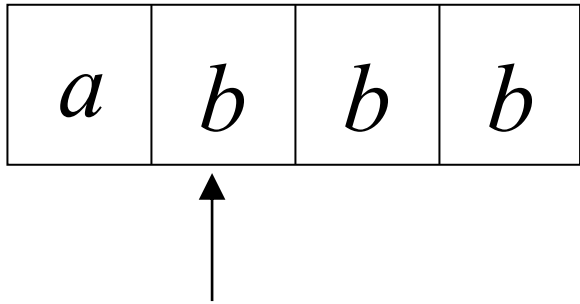
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$



Time 2

Input



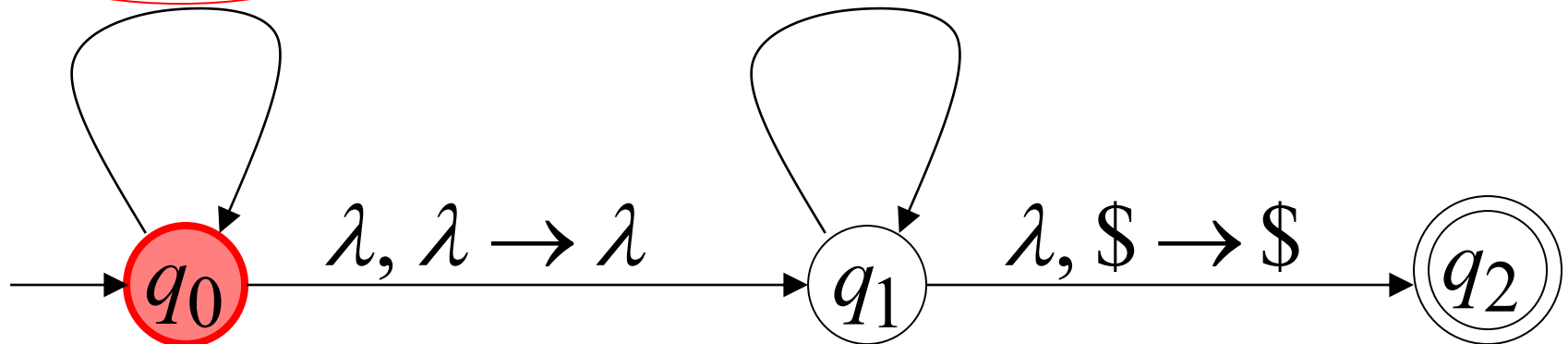
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

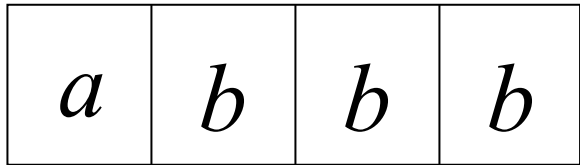
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$

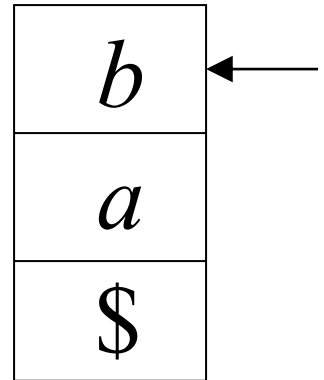


Time 3

Input



Guess the middle
of string



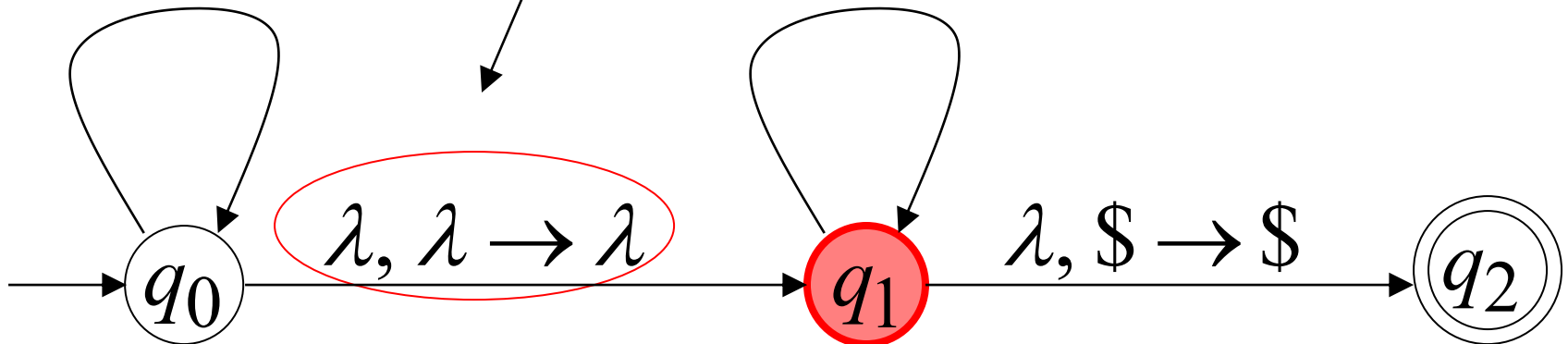
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

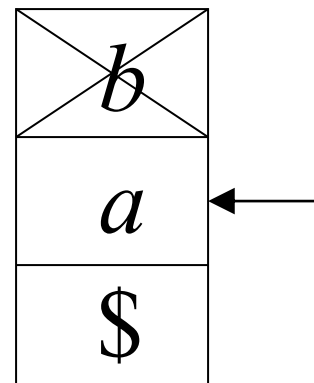
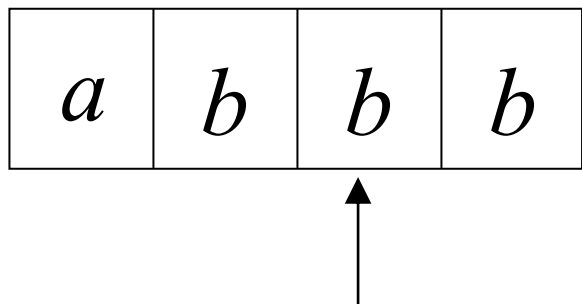
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$



Time 4

Input



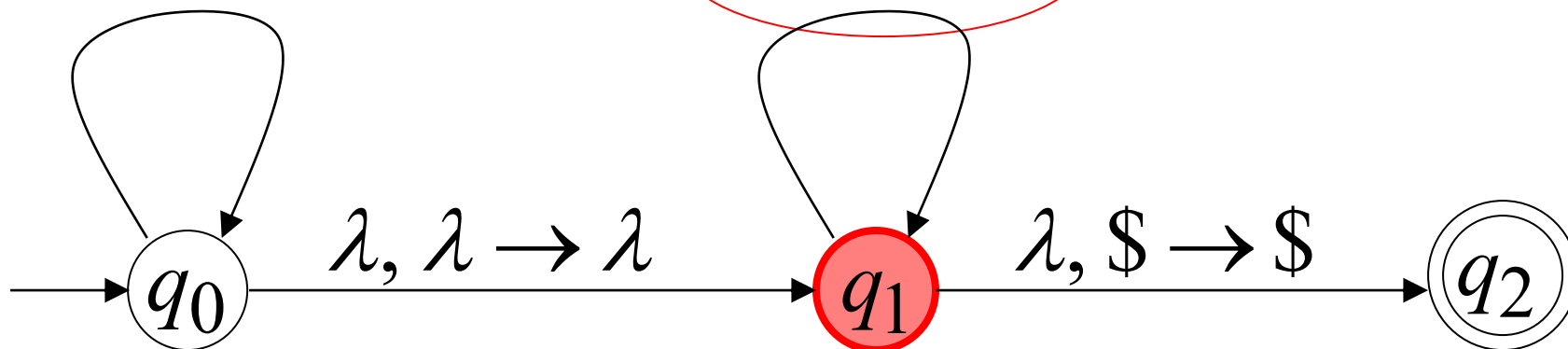
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

$a, a \rightarrow \lambda$

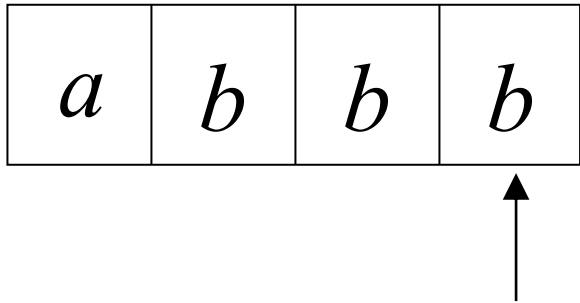
$b, b \rightarrow \lambda$



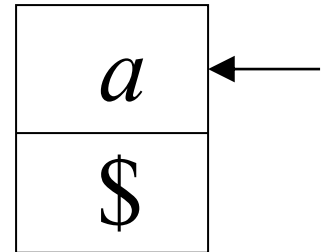
Time 5

Input

There is no possible transition.



Input is not consumed



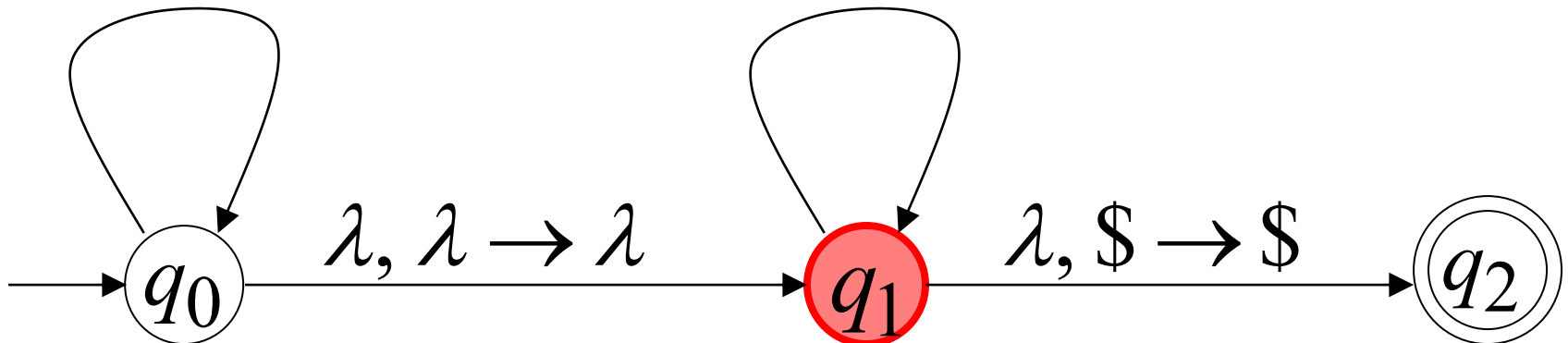
Stack

$a, \lambda \rightarrow a$

$a, a \rightarrow \lambda$

$b, \lambda \rightarrow b$

$b, b \rightarrow \lambda$

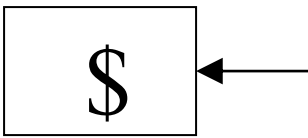
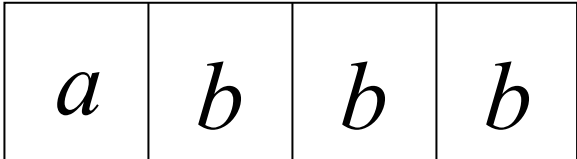


Another computation on same string:

လက်ရှိအချိန်

Input

Time 0



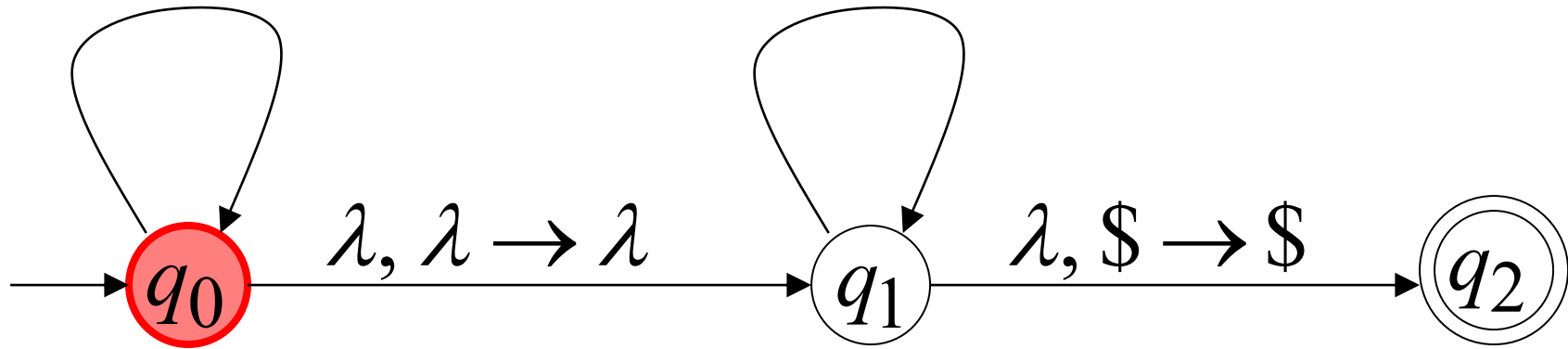
Stack

$a, \lambda \rightarrow a$

$a, a \rightarrow \lambda$

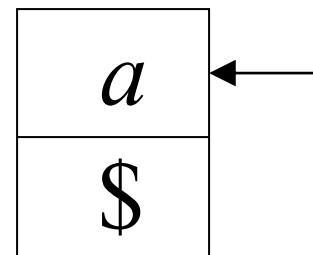
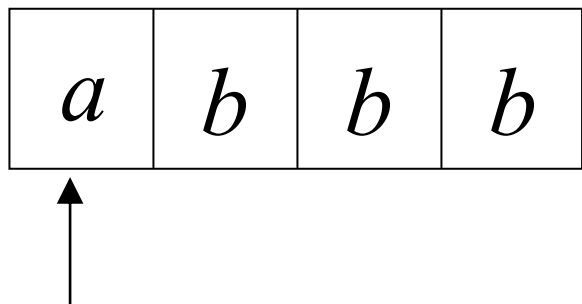
$b, \lambda \rightarrow b$

$b, b \rightarrow \lambda$



Time 1

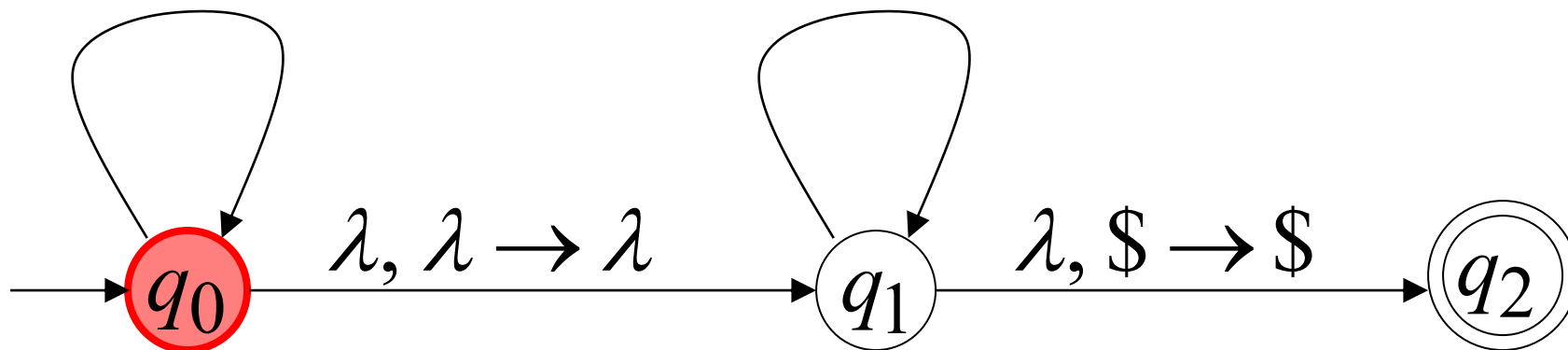
Input



Stack

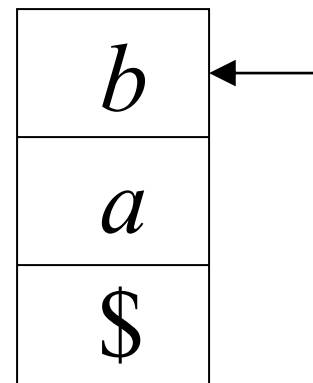
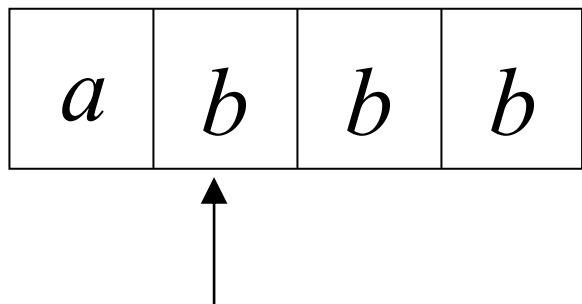
$a, \lambda \rightarrow a$
 $b, \lambda \rightarrow b$

$a, a \rightarrow \lambda$
 $b, b \rightarrow \lambda$



Time 2

Input



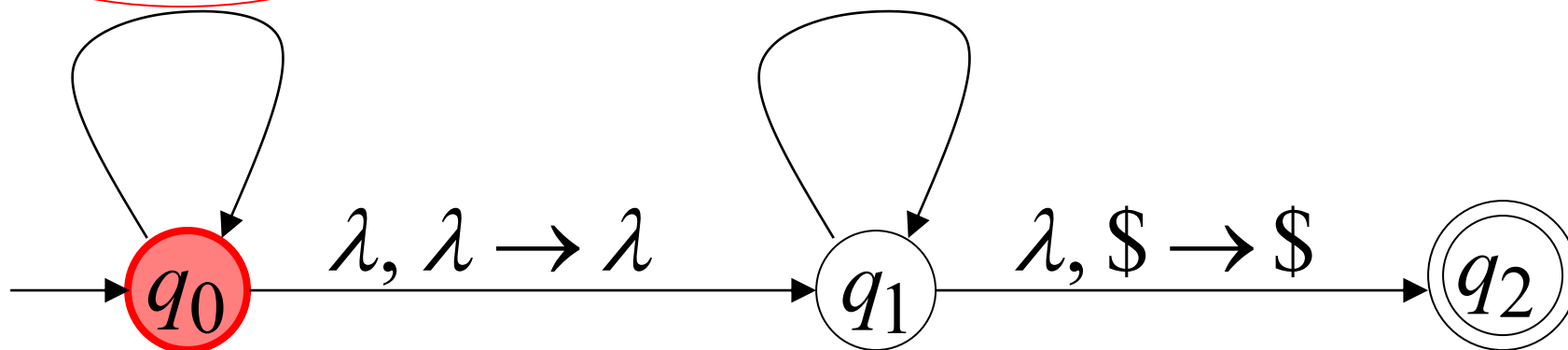
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

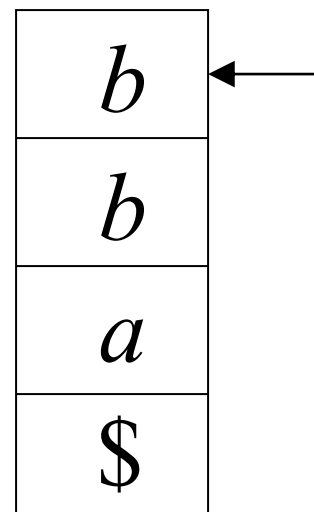
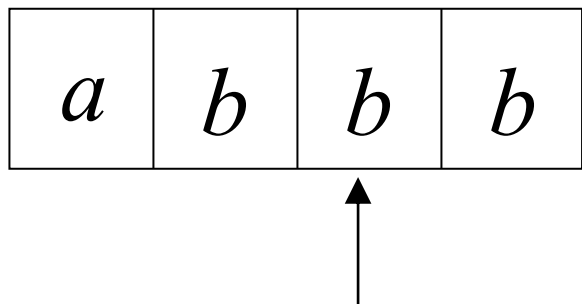
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$



Time 3

Input



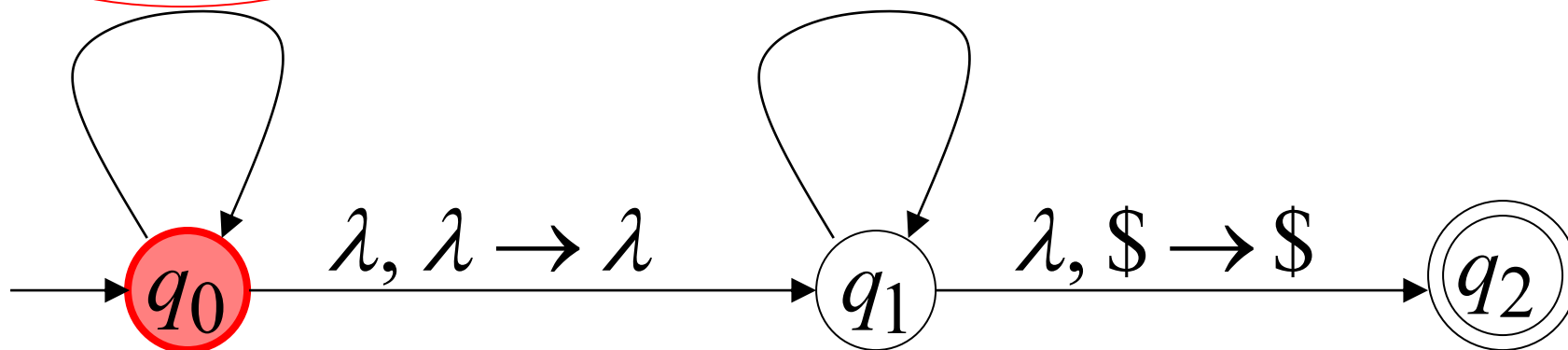
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

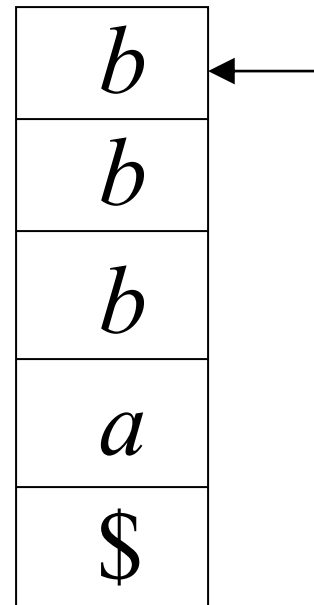
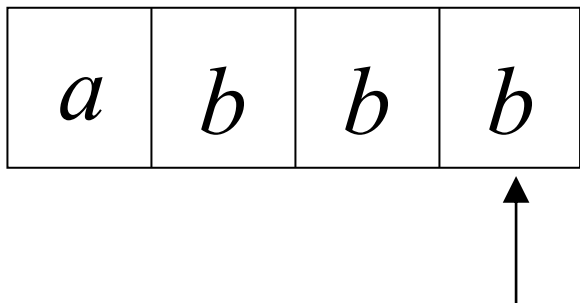
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$



Time 4

Input



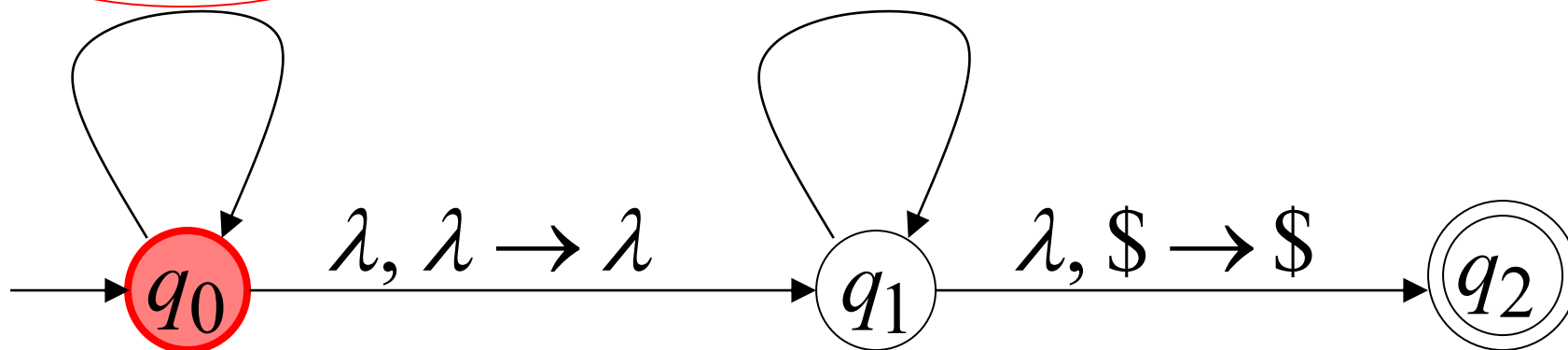
Stack

$a, \lambda \rightarrow a$

$b, \lambda \rightarrow b$

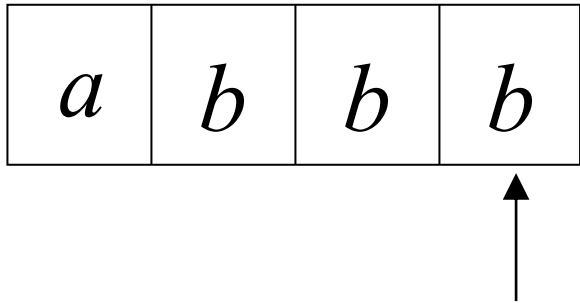
$a, a \rightarrow \lambda$

$b, b \rightarrow \lambda$

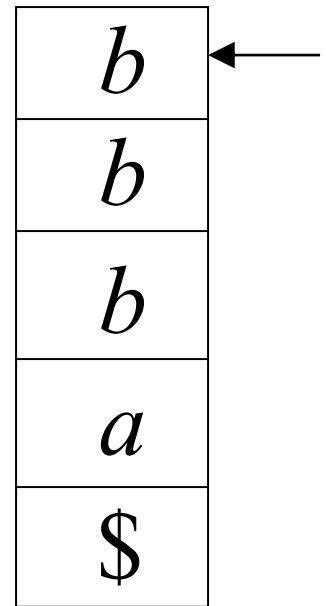


Time 5

Input



No final state
is reached



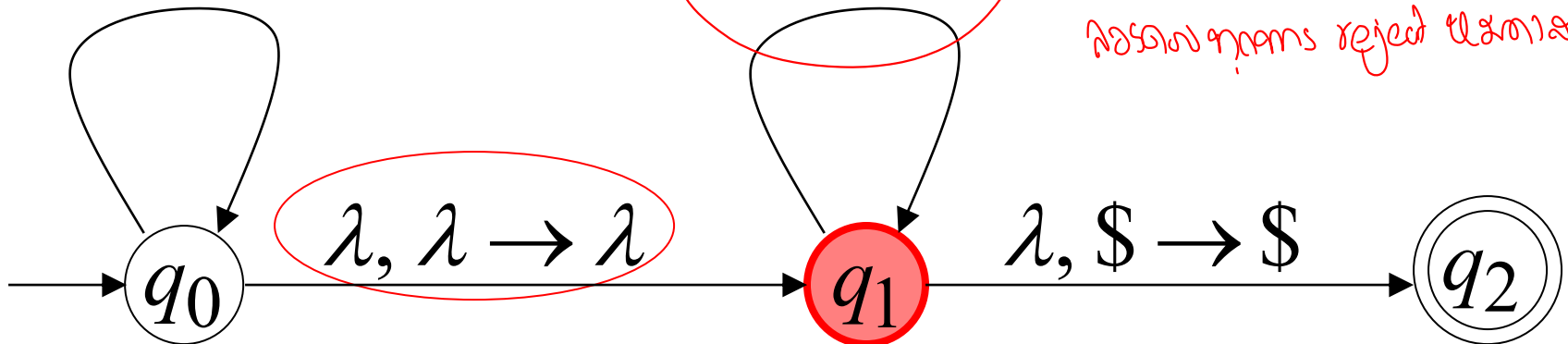
Stack

$a, \lambda \rightarrow a$
 $b, \lambda \rightarrow b$

$a, a \rightarrow \lambda$
 $b, b \rightarrow \lambda$

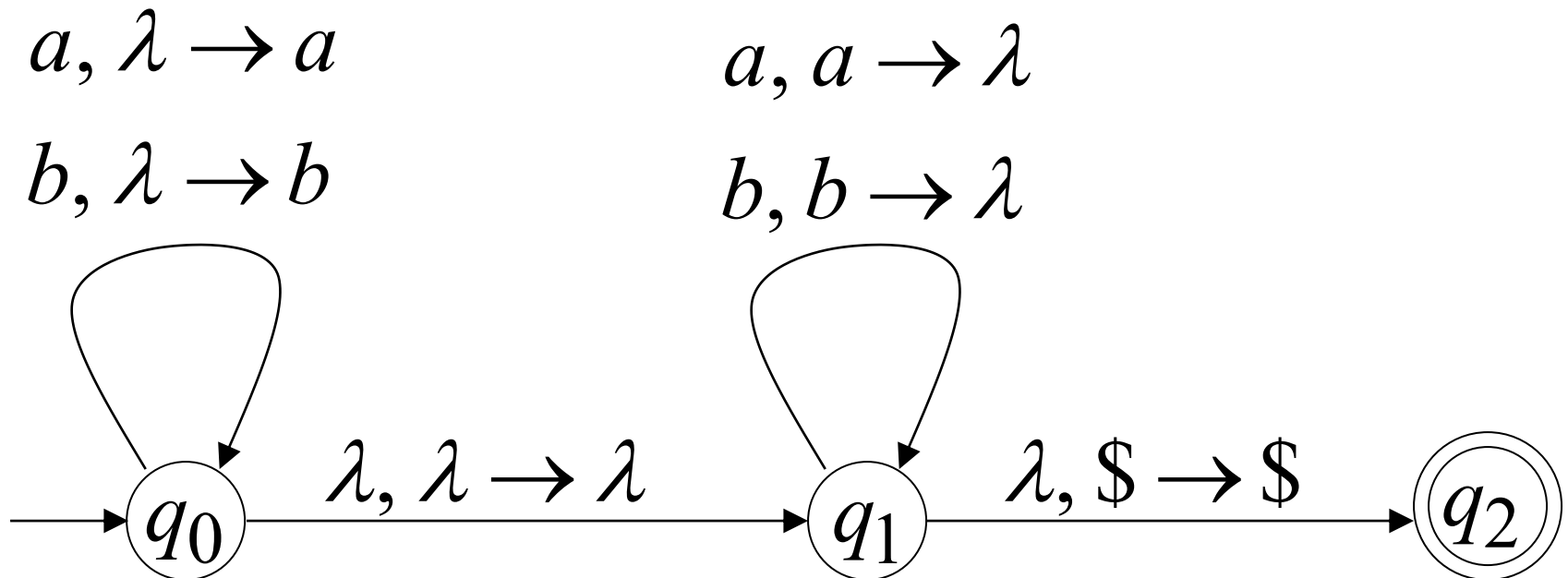
Handwritten notes:
"No final state is reached" (circled)
"reject string" (with arrow pointing to the transition)

Handwritten note:
"No final state is reached" (circled)



There is no computation
that accepts string $abbb$

$$abbb \notin L(M)$$



A string is rejected if there is
no computation such that:

All the input is consumed

AND

The last state is a final state

At the end of the computation,
we do not care about the stack contents

In other words, a string is rejected
if in every computation with this string:

The input cannot be consumed

OR

input *not* consumed

The input is consumed but the last
state is not a final state

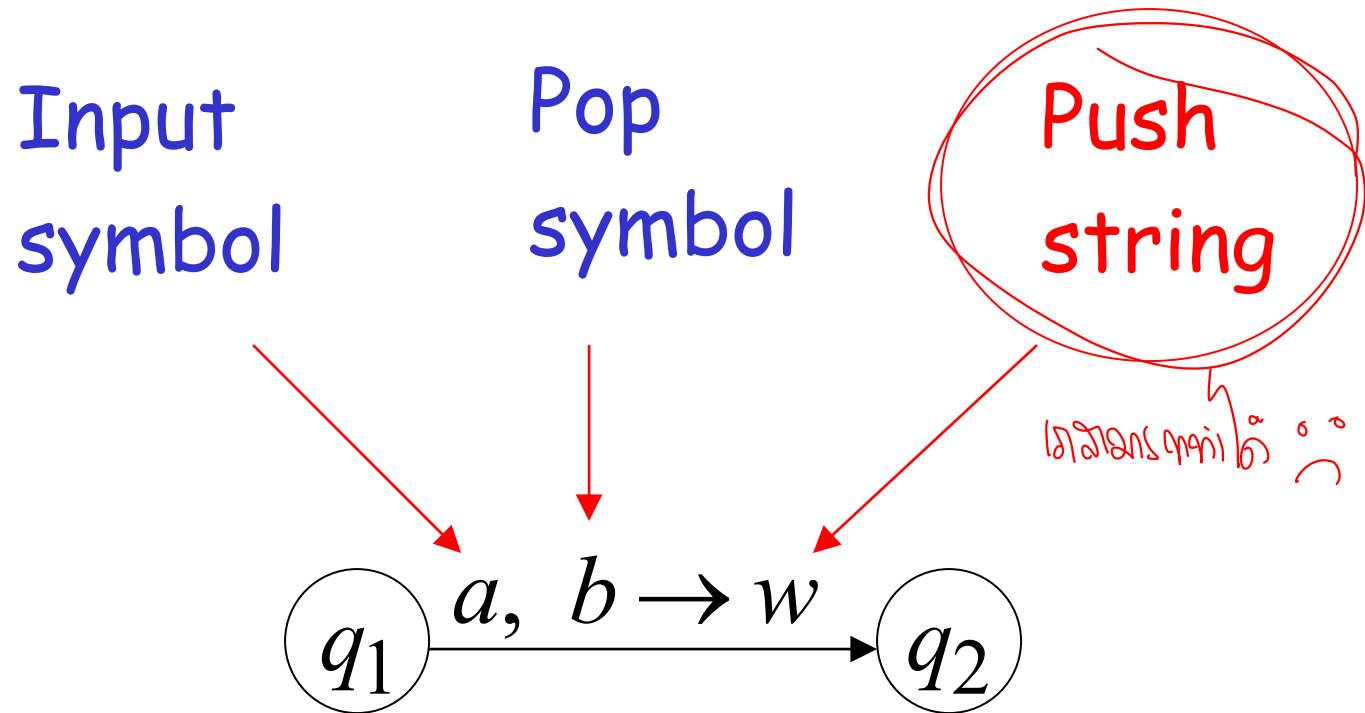
OR

input *consumed* but *not* in final state

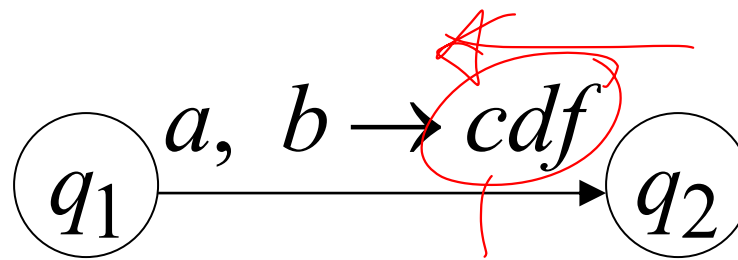
The stack head moves below the
bottom of the stack

stack *underflow*

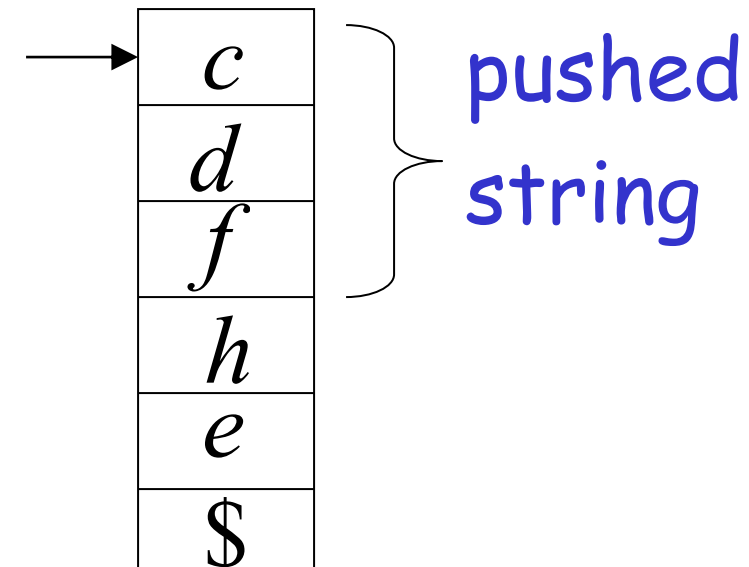
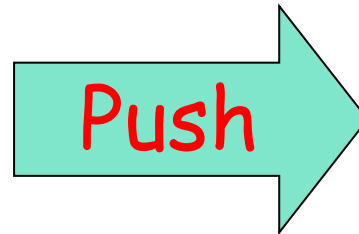
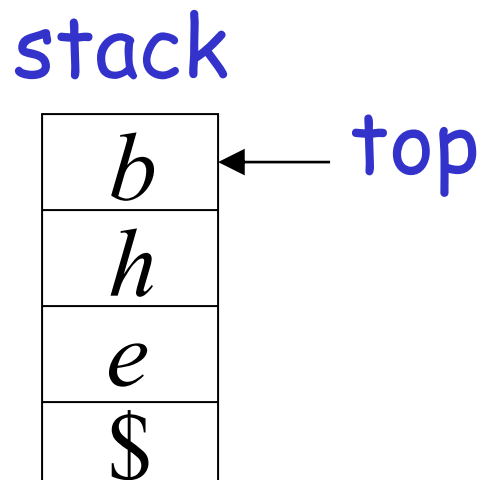
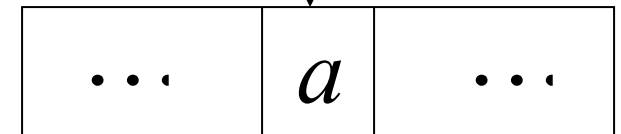
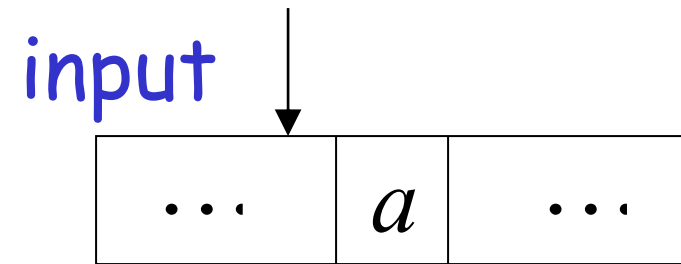
Pushing Strings



Example:



push แบบทอยบอลใส่กล่อง pop แล้วได้บอลเดิม



Another NPDA example

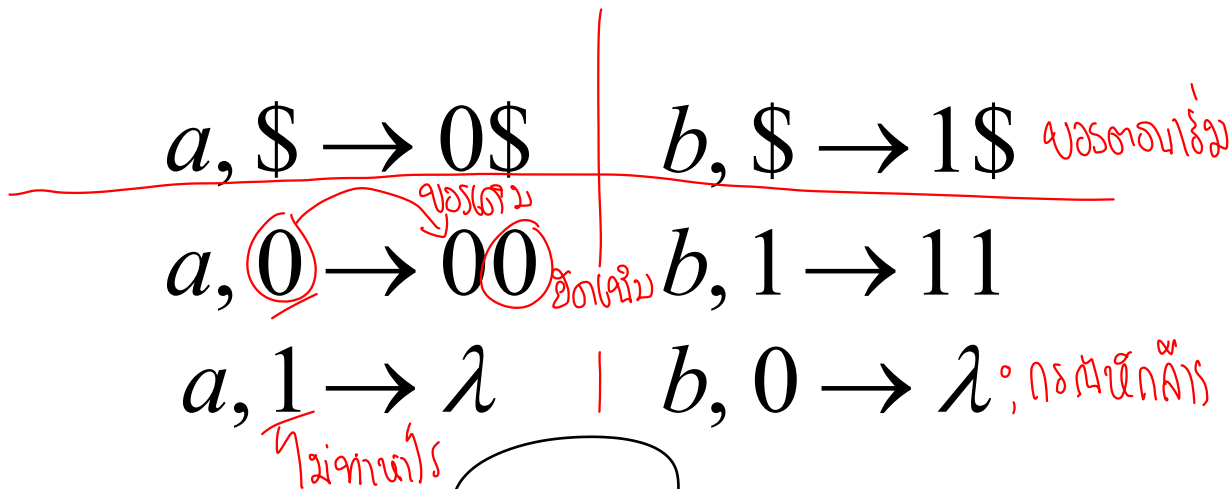
NPDA M

① ប្រើការប្រើប្រាស់ algorithm

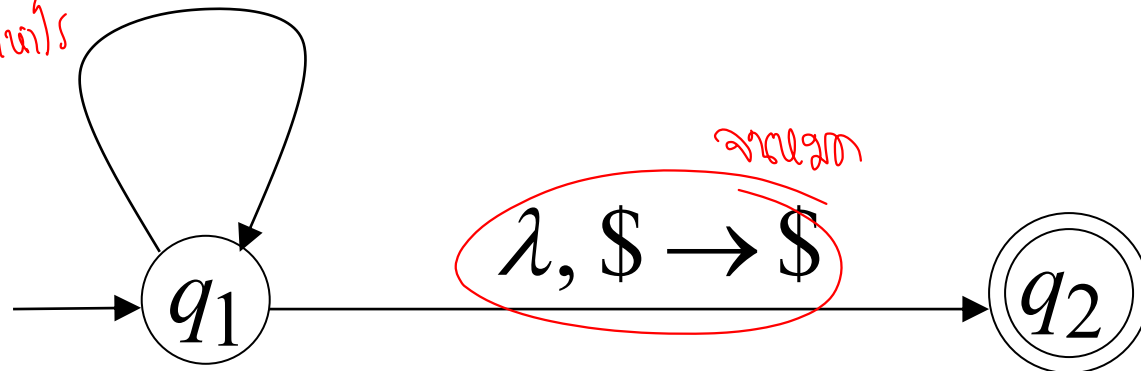
a:0
b:1

$$L(M) = \{w : n_a = n_b\}$$

abbbac

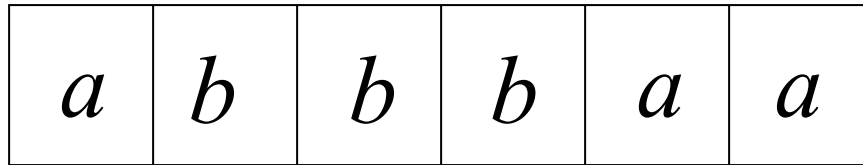


$\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$



Execution Example: Time 0

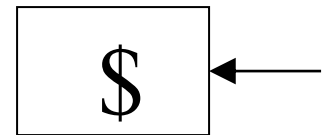
Input



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

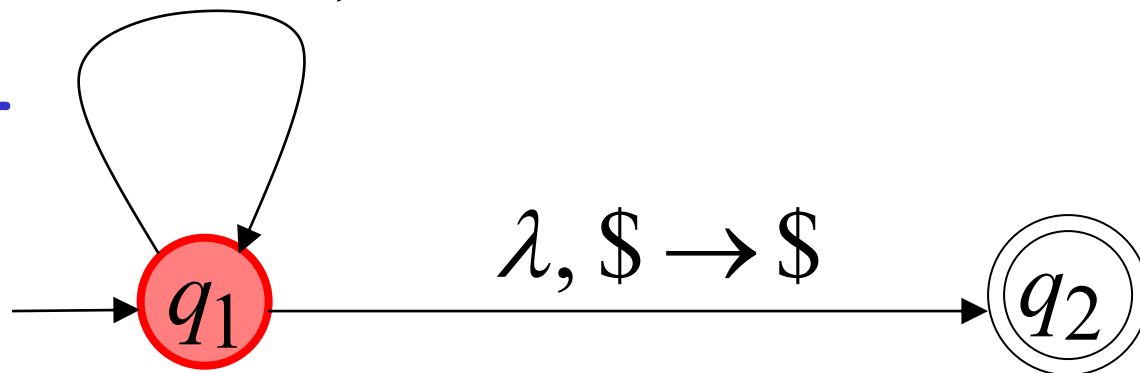
$a, 0 \rightarrow 00$ $b, 1 \rightarrow 11$

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



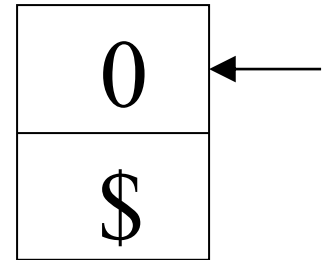
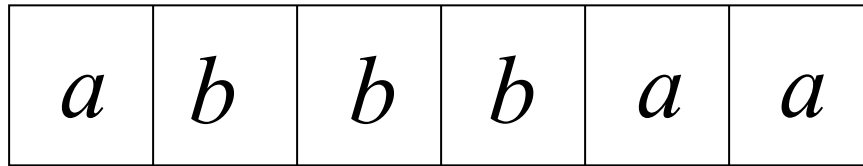
Stack

current
state



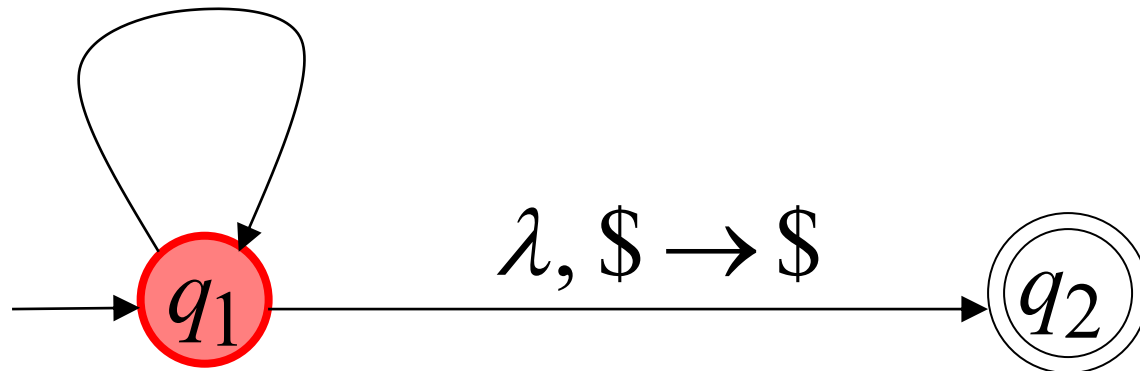
Time 1

Input



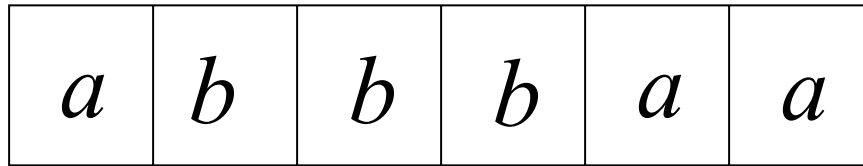
Stack

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



Time 3

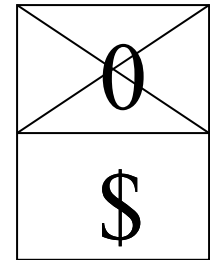
Input



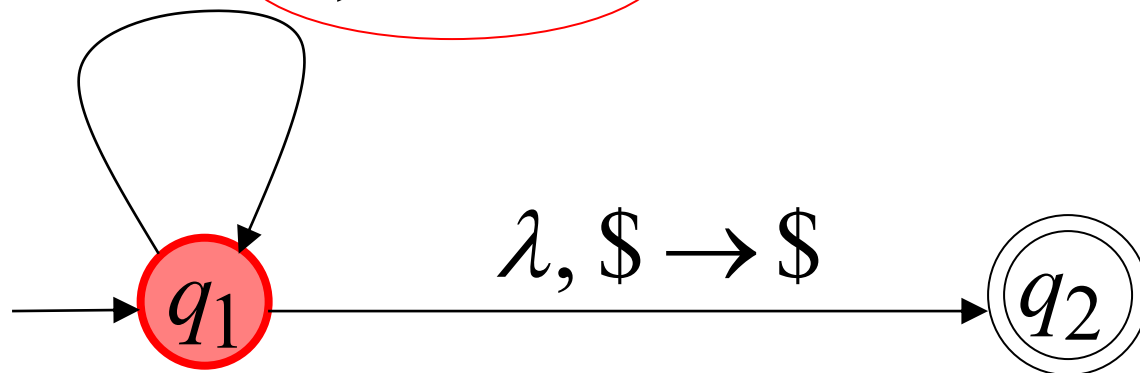
$a, \$ \rightarrow 0\$$ $b, \$ \rightarrow 1\$$

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

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

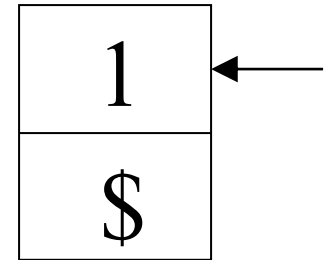
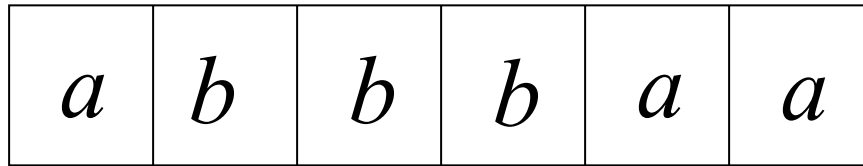


Stack



Time 4

Input

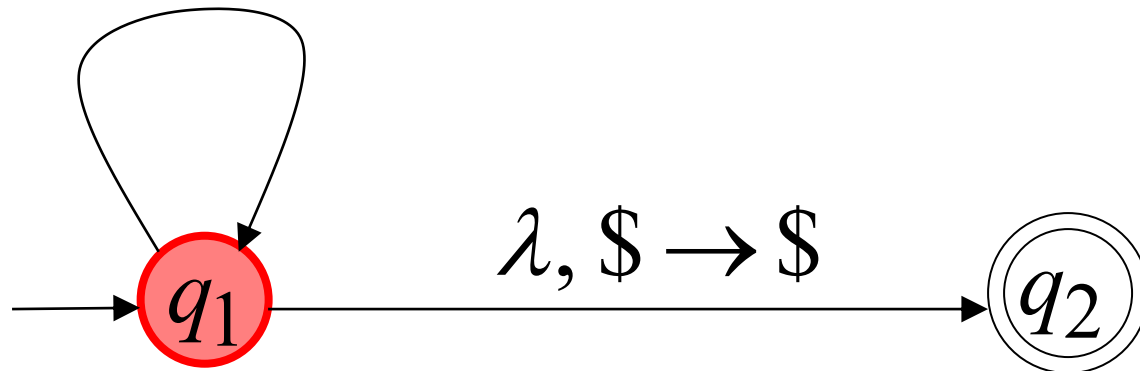


Stack

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

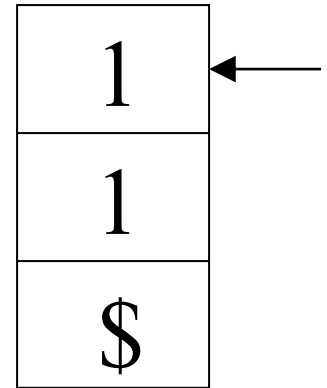
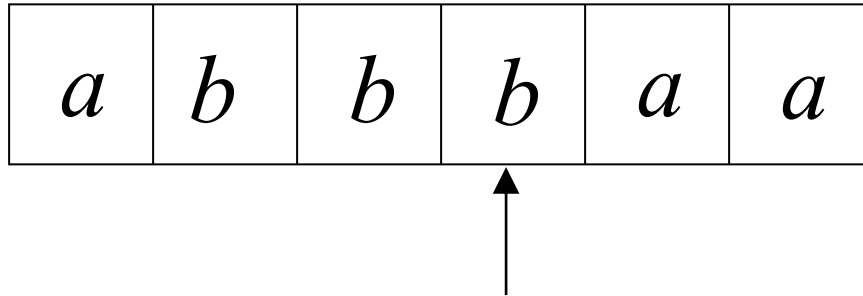
$a, 0 \rightarrow 00$ $b, 1 \rightarrow 11$

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



Time 5

Input

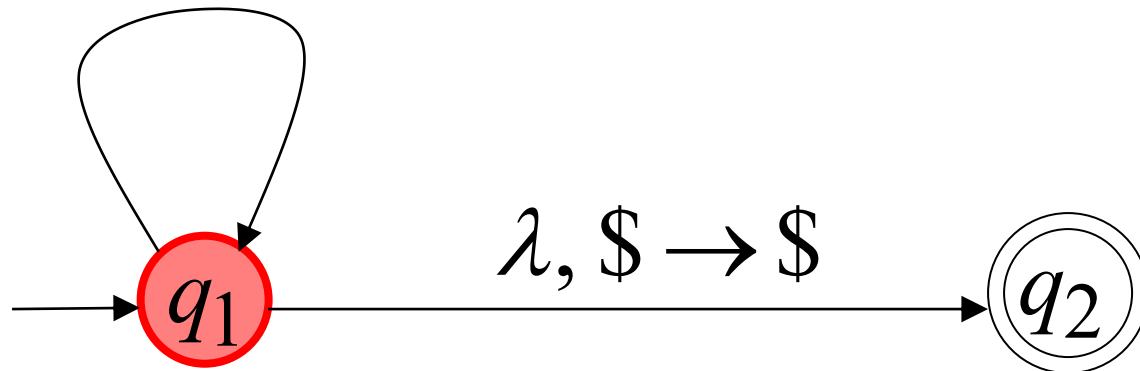


Stack

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

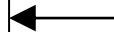
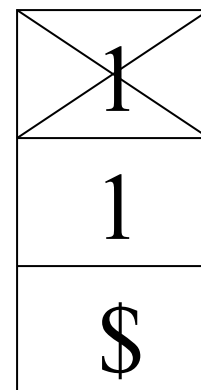
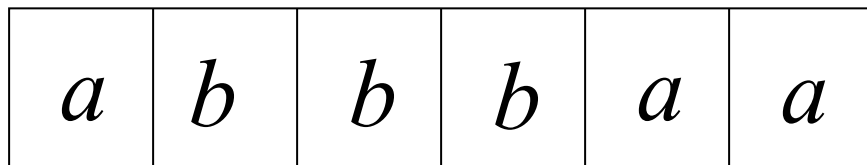
$a, 0 \rightarrow 00$ $b, 1 \rightarrow 11$

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



Time 6

Input

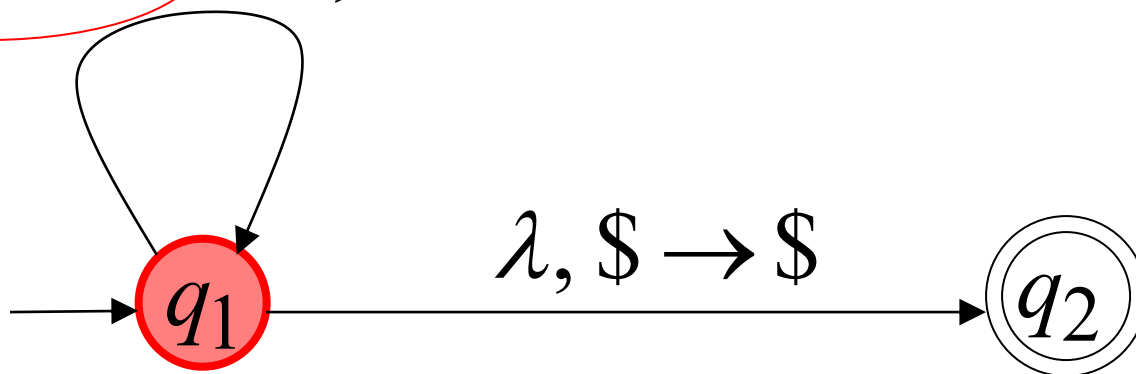


Stack

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

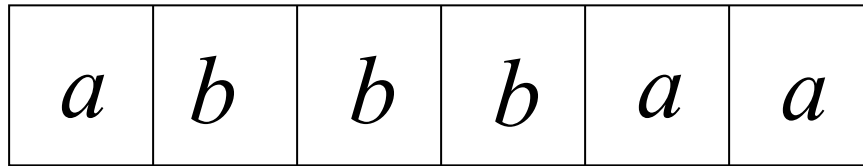
$a, 0 \rightarrow 00$ $b, 1 \rightarrow 11$

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



Time 7

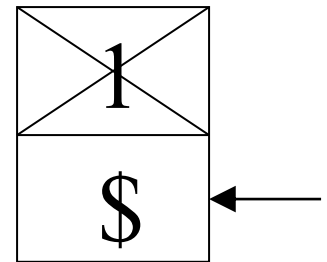
Input



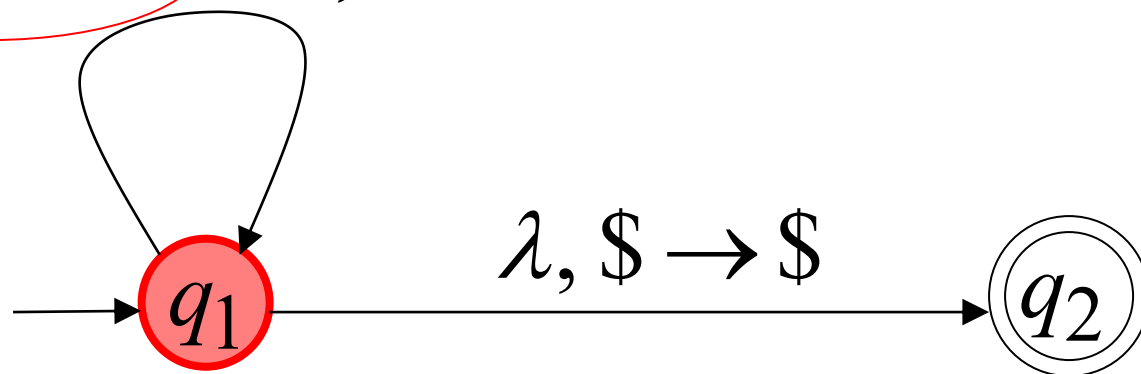
$a, \$ \rightarrow 0\$$ $b, \$ \rightarrow 1\$$

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

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

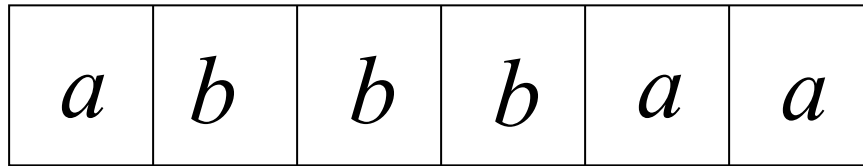


Stack



Time 8

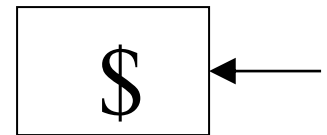
Input



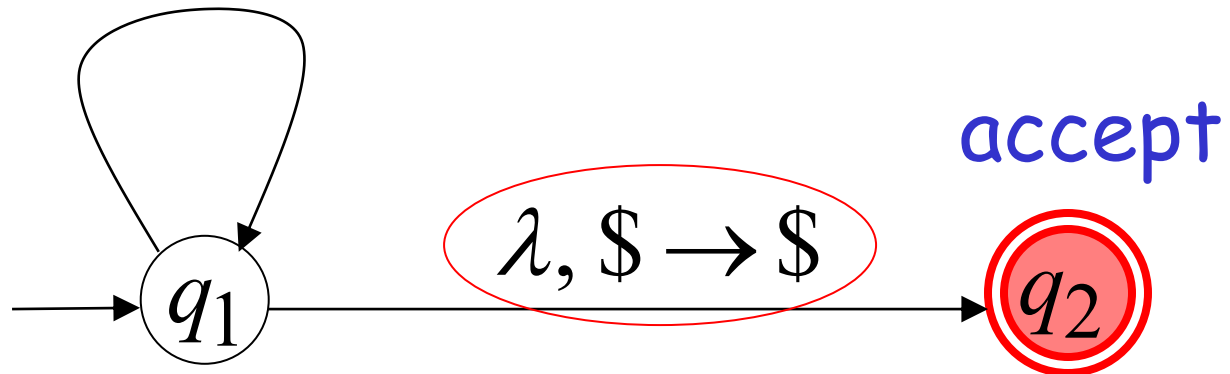
$a, \$ \rightarrow 0\$$ $b, \$ \rightarrow 1\$$

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

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

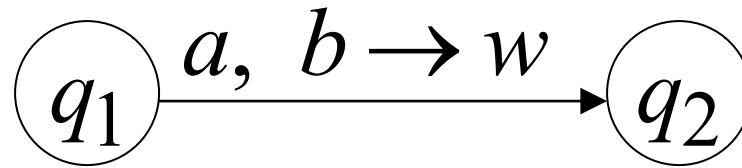


Stack



Formalities for NPDAs

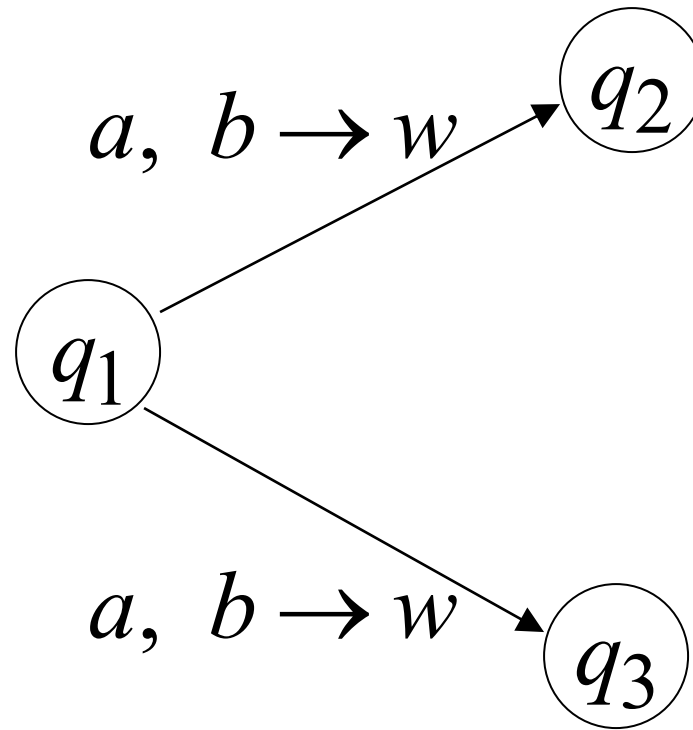
↙ বাংলা text → গ্রাফিক মোড



Transition function:

$$\delta(\underline{q_1}, \overset{\text{input}}{a}, \overset{\text{top}}{b}) = \{(q_2, \overset{\text{output}}{w})\}$$

Handwritten red annotations: "input" above a , "top" above b , "output" above w , "push" below w , and "str" above w . A red arrow points from the text "Transition function:" to the δ symbol.



Transition function:

$$\delta(q_1, a, b) = \{(q_2, w), (q_3, w)\}$$

4/19/2023

Formal Definition

Non-Deterministic Pushdown Automaton

NPDA

উদ্ভাটকীয়

গ্রামার δ

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

গ্রামার (γ)

delta (δ)

States

Input
alphabet

Stack
alphabet

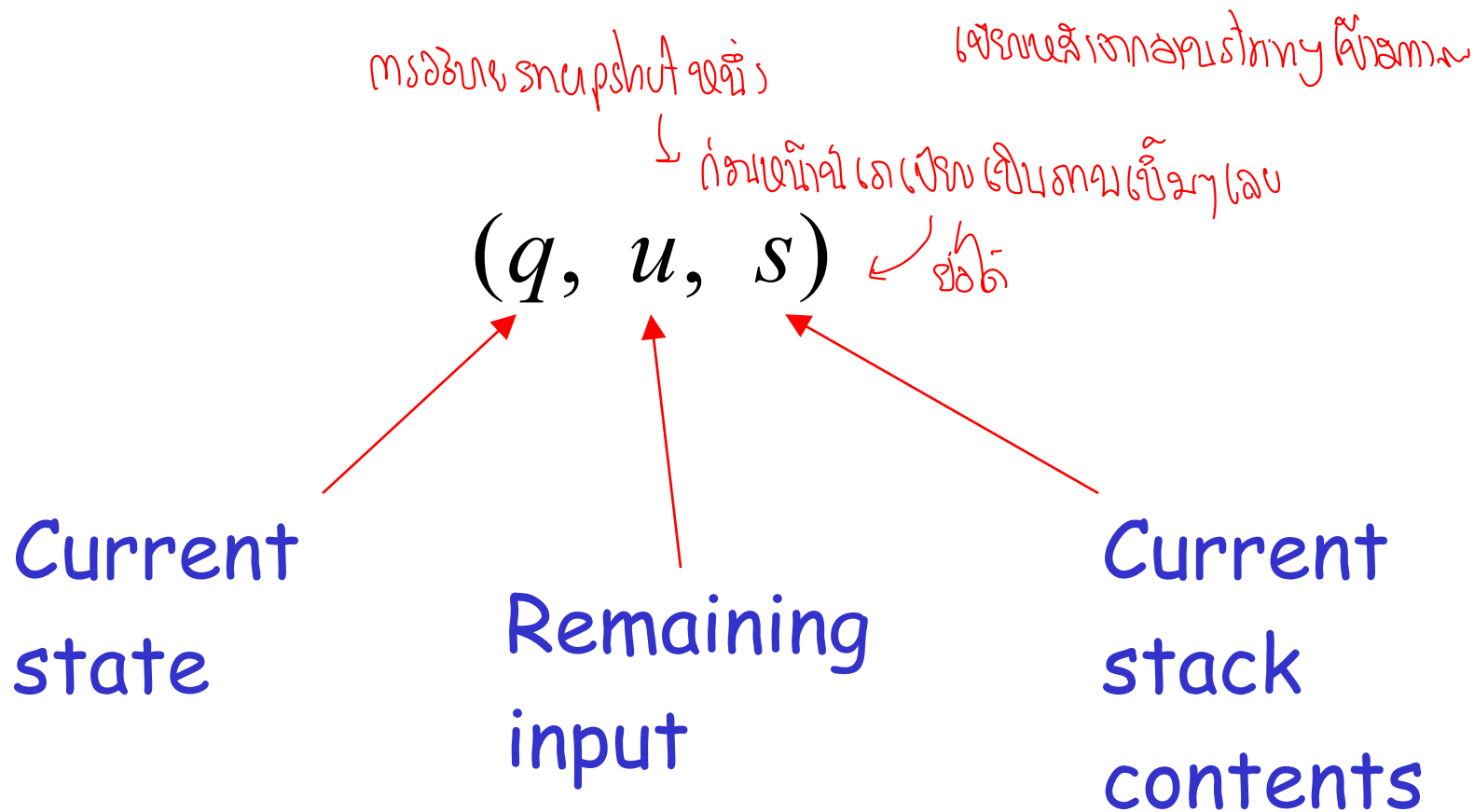
Transition
function

Initial
state

Final
states

Stack
start
symbol

Instantaneous Description



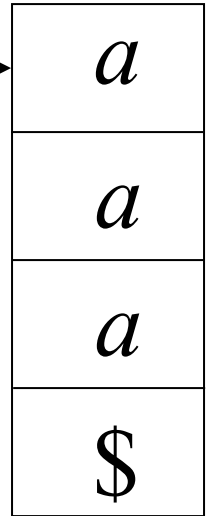
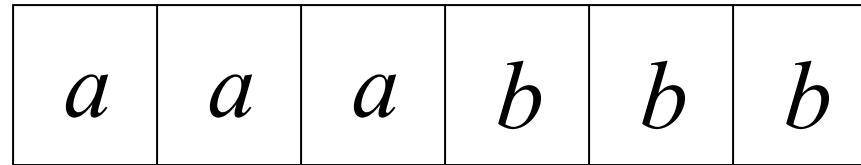
Example:

Instantaneous Description

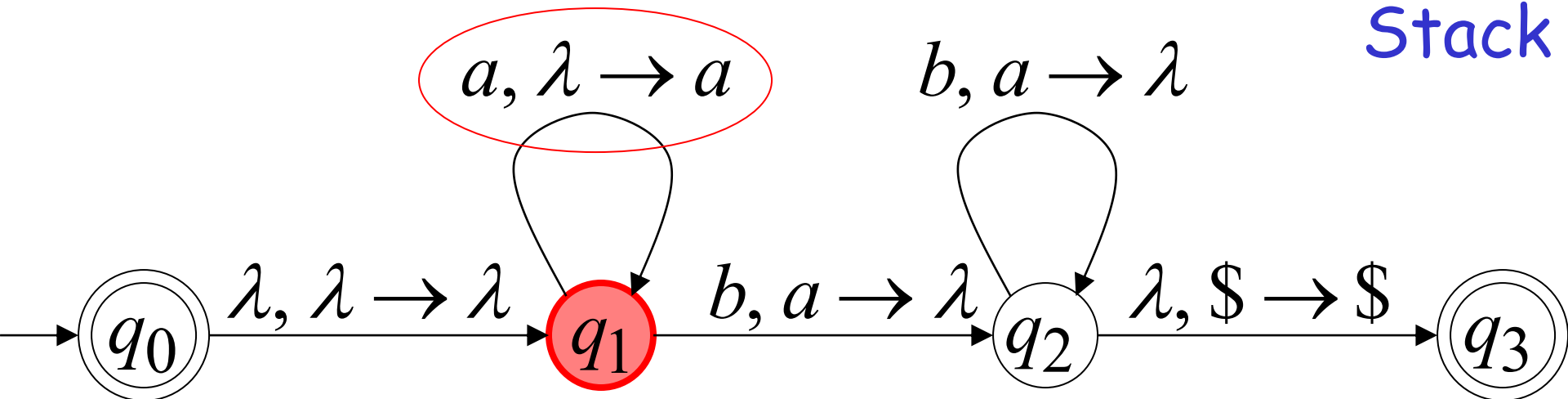
$(q_1, bbb, aaa\$)$

Handwritten notes: q_1 (with a checkmark), bbb (with an arrow), and q_1 (with an arrow). A red arrow points from the q_1 in the notes to the q_1 in the tuple above.

Input



Stack



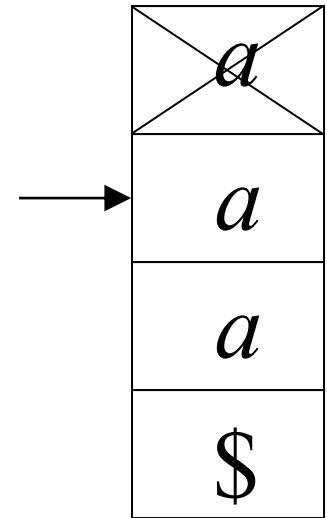
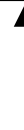
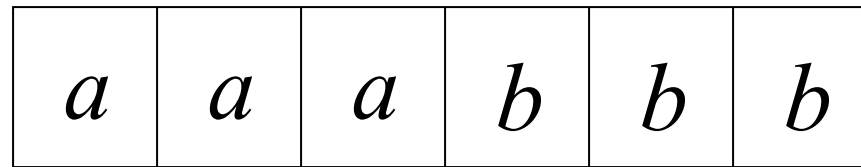
Example:

Instantaneous Description

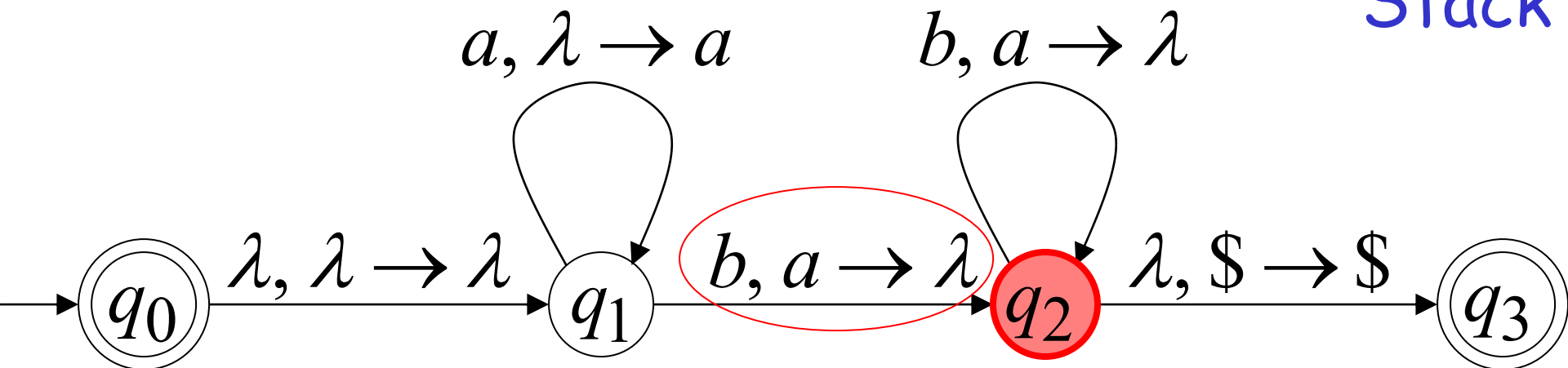
$(q_2, bb, aa\$)$

Time 5:

Input



Stack



We write:

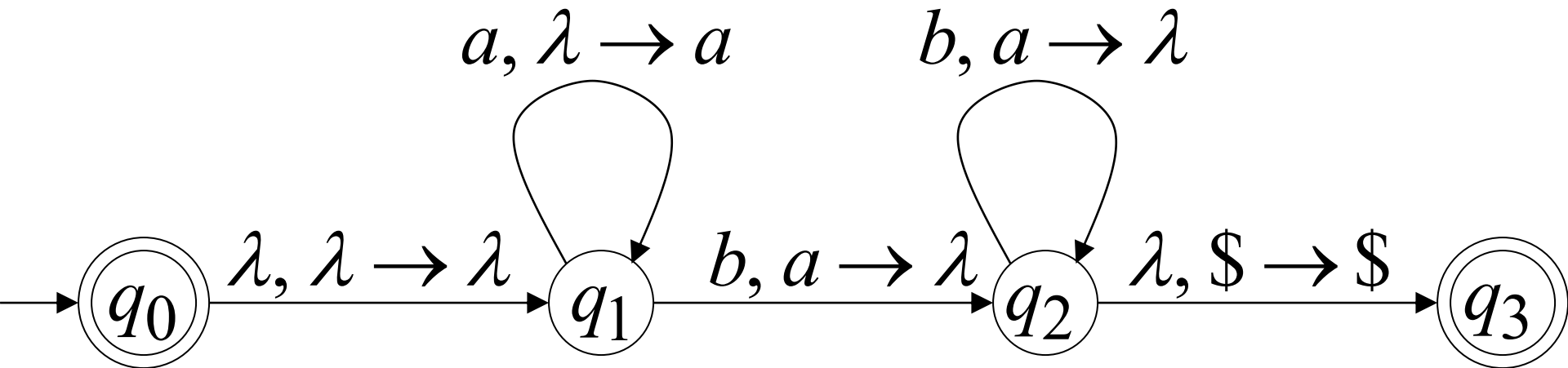
$$(q_1, bbb, aaa\$) \overset{\text{transition required}}{>} (q_2, bb, aa\$)$$

Time 4

Time 5

A computation:

$(q_0, aaabbbb, \$) \succ (q_1, aaabbbb, \$) \succ$
 $(q_1, aabbbb, a\$) \succ (q_1, abbbb, aa\$) \succ (q_1, bbb, aaa\$) \succ$
 $(q_2, bb, aa\$) \succ (q_2, b, a\$) \succ (q_2, \lambda, \$) \succ (q_3, \lambda, \$)$
අනුපිළිවෙල



$$\begin{aligned}
 &(q_0, aaabbbb, \$) \succ (q_1, aaabbbb, \$) \succ \\
 &(q_1, aabbbb, a\$) \succ (q_1, abbbb, aa\$) \succ (q_1, bbb, aaa\$) \succ \\
 &(q_2, bb, aa\$) \succ (q_2, b, a\$) \succ (q_2, \lambda, \$) \succ (q_3, \lambda, \$)
 \end{aligned}$$

For convenience we write:

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

Formal Definition

Language $L(M)$ of NPDA M :

$$L(M) = \{w: (q_0, w, s) \xrightarrow{*} (q_f, \lambda, s')\}$$

Handwritten annotations in Burmese:

- q_0 : စတင်အခြေအနေ (Initial state)
- w : စာသား (String)
- s : စတင် input stack (Initial input stack)
- $\xrightarrow{*}$: အကောင်အထည်ဖော်မှု (Computation)
- q_f : အဆုံးအခြေအနေ (Final state)
- λ : လျှပ် input stack (Empty input stack)
- s' : အဆုံးအခြေအနေ (Final state)

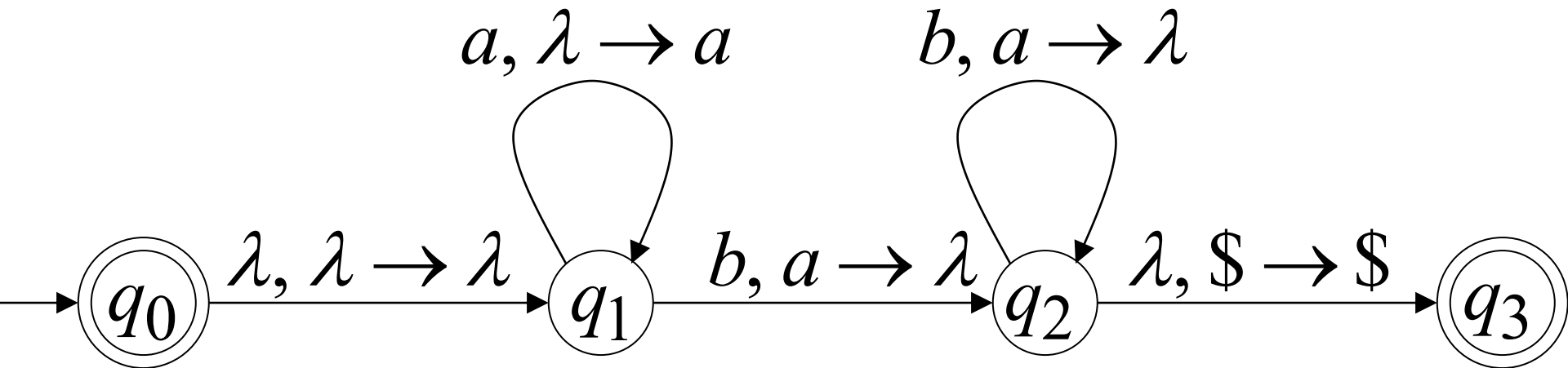
Example:

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



$$aaabbb \in L(M)$$

NPDA M :

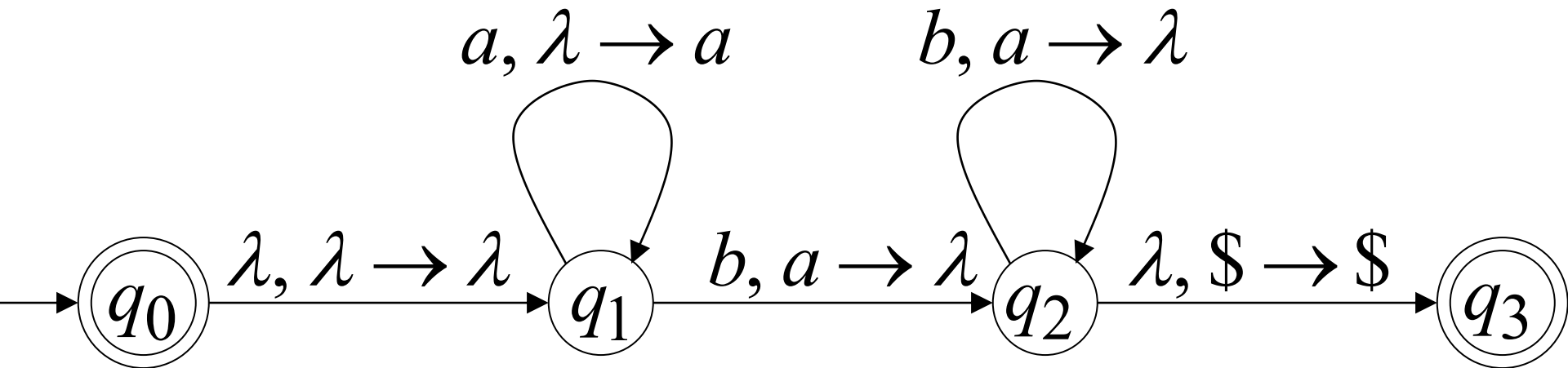


$$(q_0, a^n b^n, \$) \stackrel{*}{\succ} (q_3, \lambda, \$)$$



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

NPDA M :



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

NPDA M :

