Department of Computer & Information Sciences
Pakistan Institute of Engineering and Applied Sciences

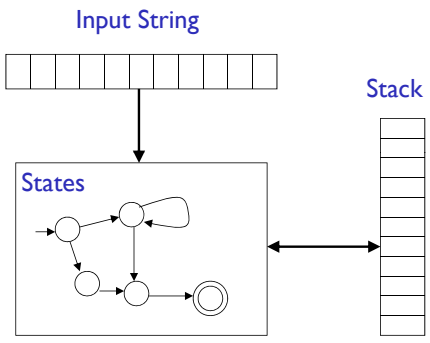
Pushdown Automata
PDAs

Umar Faiz
http://www.pieas.edu.pk/umarfaiz/cis317

Pushdown Automata (PDA)

- A Pushdown Automaton (PDA) is an abstract computing device similar to the FSA. It has a finite set of states.
- PDA may be pictured as a finite automaton with the addition of a stack onto which symbols may be 'pushed' or from which they may be 'popped'. A PDA is an NFA-ε with a stack.

Pushdown Automata (PDA)

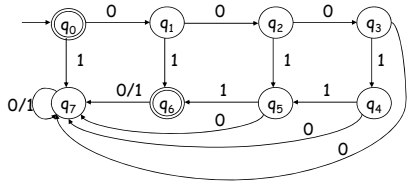


Pushdown Automata (PDA)

A DFA can "remember" only a finite amount of information, whereas a PDA can "remember" an infinite amount of (certain types of) information.

$\{0^n1^n \mid 0=<n\}$	Is *not* regular
$\{0^n1^n \mid 0\leq n\leq k$, for some fixed $k\}$	Is regular, for any fixed k.

For k=3: L = {ε, 01, 0011, 000111}



Pushdown Automata (PDA)

- In a DFA, each state remembers a finite amount of information.
- To get $\{0^n1^n \mid 0\leq n\}$ with a DFA would require an infinite number of states using the preceding technique.
- An infinite stack solves the problem for $\{0^n1^n \mid 0\leq n\}$ as follows:
  - Read all 0's and place them on a stack
  - Read all 1's and match with the corresponding 0's on the stack

Only need two states to do this in a PDA

Formal Definition of a PDA

A pushdown automaton (PDA) is a seven-tuple:

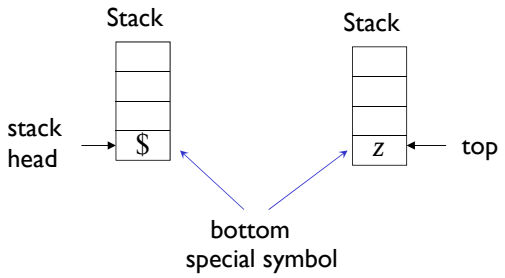
$M = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$

- Q A finite set of states
- Σ A finite input alphabet
- Γ A finite stack alphabet
- q_0 The initial/starting state, q_0 is in Q
- z_0 A starting stack symbol, is in Γ
- F A set of final/accepting states, which is a subset of Q
- δ A transition function, where

  $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow$ finite subsets of $Q \times \Gamma^*$

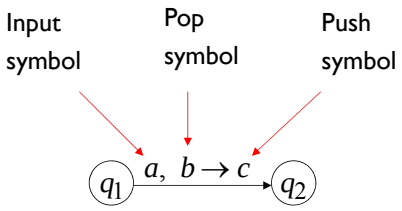
Pushdown Automaton - PDA

Initial Stack Symbol

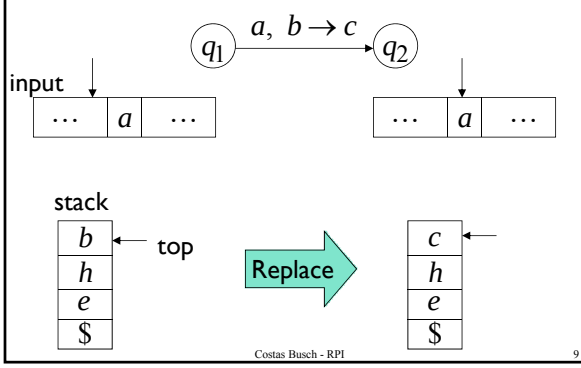


Pushdown Automaton - PDA

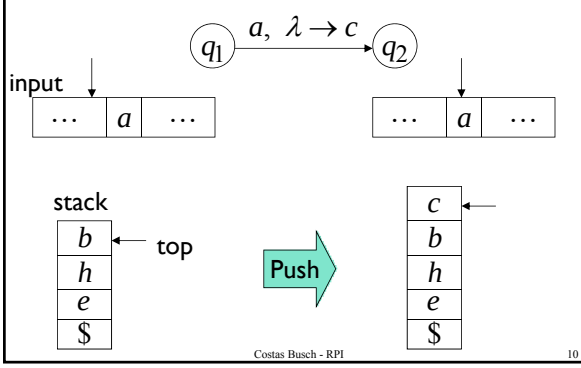
The States



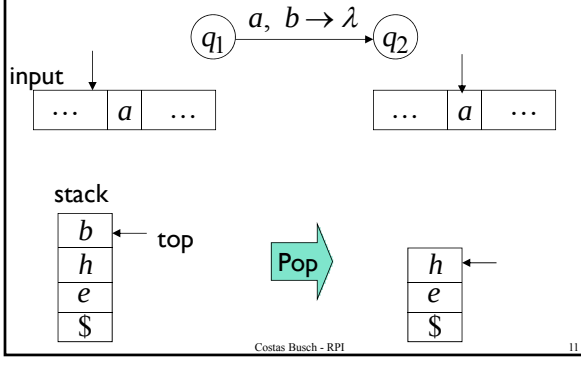
Pushdown Automaton - PDA



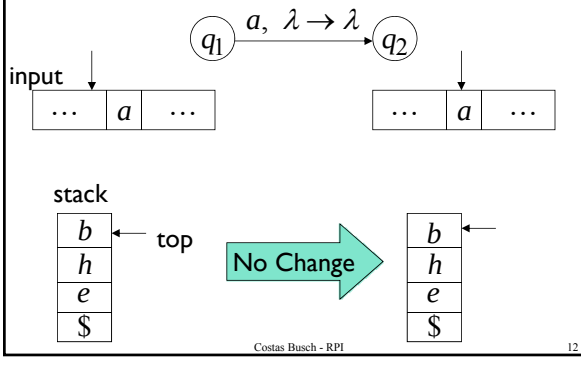
Pushdown Automaton - PDA



Pushdown Automaton - PDA

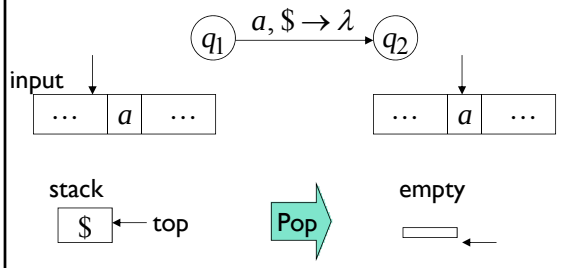


Pushdown Automaton - PDA



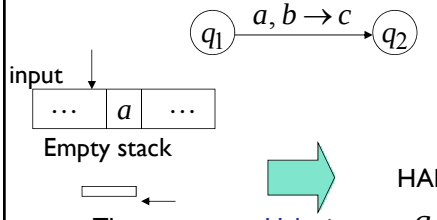
Pushdown Automaton - PDA

A Possible Transition:



Pushdown Automaton - PDA

A Bad Transition:

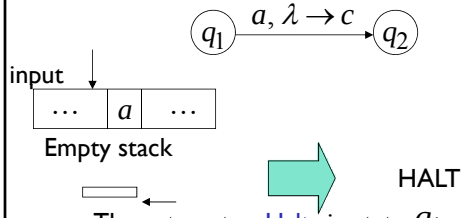


The automaton Halts in state q_1

and Rejects the input string

Pushdown Automaton - PDA

A Bad Transition:

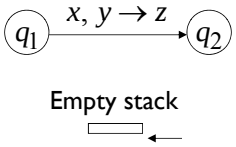


The automaton Halts in state q_1

and Rejects the input string

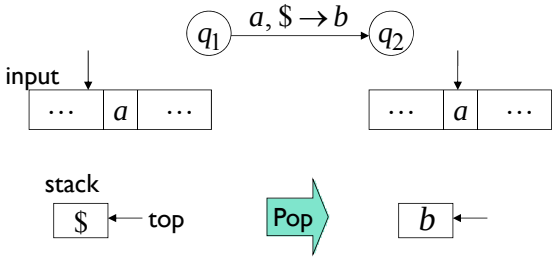
Pushdown Automaton - PDA

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

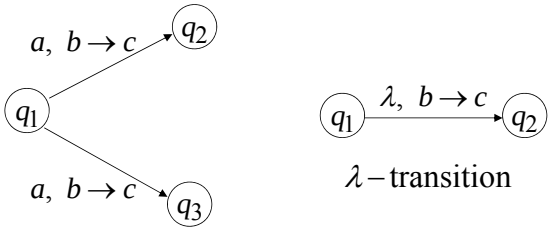


Pushdown Automaton - PDA

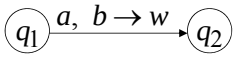
A Good Transition:



Non-Determinism

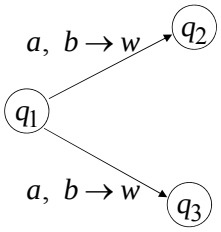


These are allowed transitions in a Non-deterministic PDA (NPDA)



Transition function:

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



Transition function:

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

Non-Deterministic Pushdown Automaton

Definition:

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

States

Input alphabet

Stack alphabet

Transition function

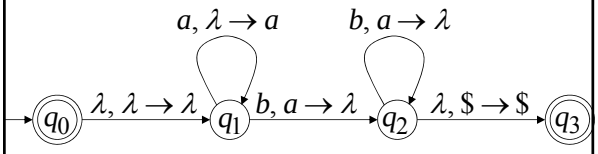
Initial state

Stack start symbol

Final states

NPDA: Non-Deterministic PDA

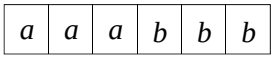
Example:

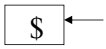


NPDA: Non-Deterministic PDA

Execution Example: Time 0

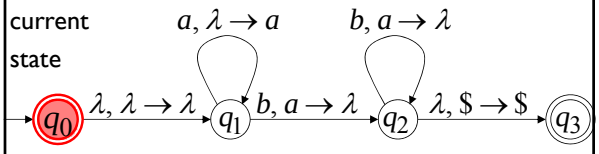
Input





Stack

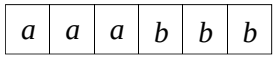
current state

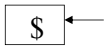


NPDA: Non-Deterministic PDA

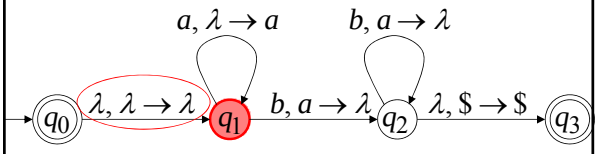
Execution Example: Time 1

Input



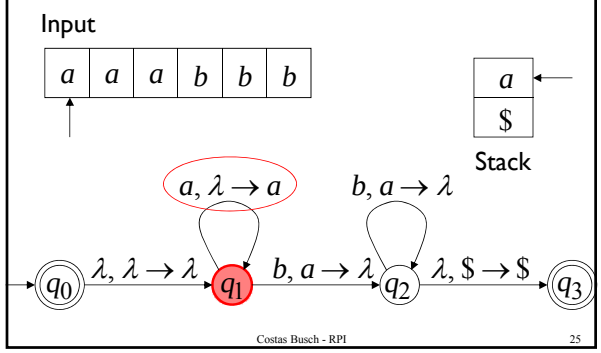


Stack



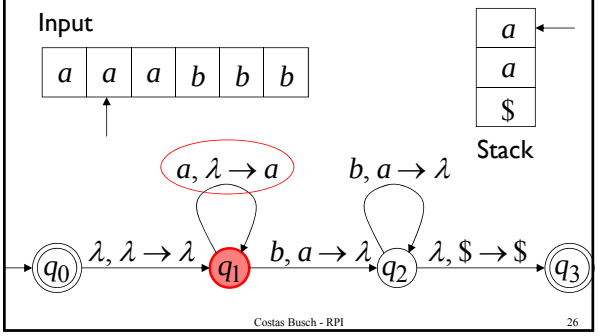
NPDA: Non-Deterministic PDA

Execution Example: Time 2



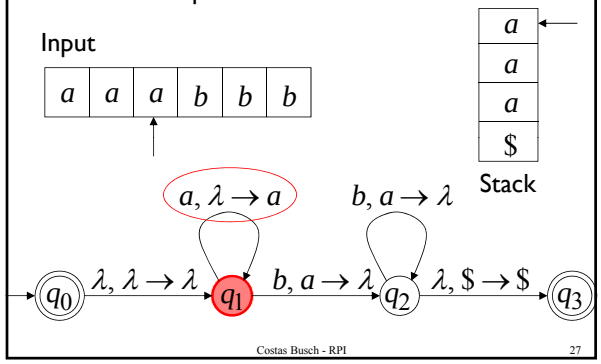
NPDA: Non-Deterministic PDA

Execution Example: Time 3



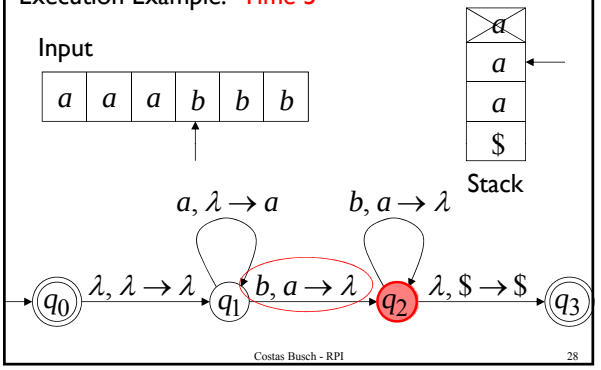
NPDA: Non-Deterministic PDA

Execution Example: Time 4



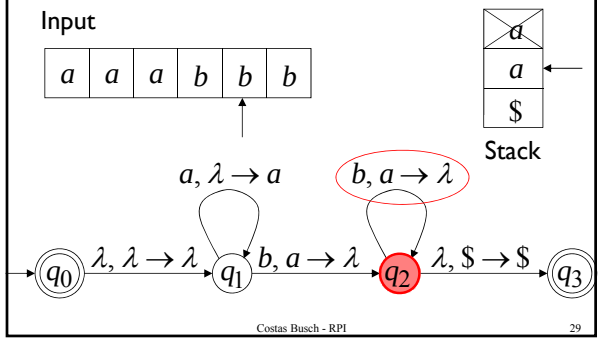
NPDA: Non-Deterministic PDA

Execution Example: Time 5



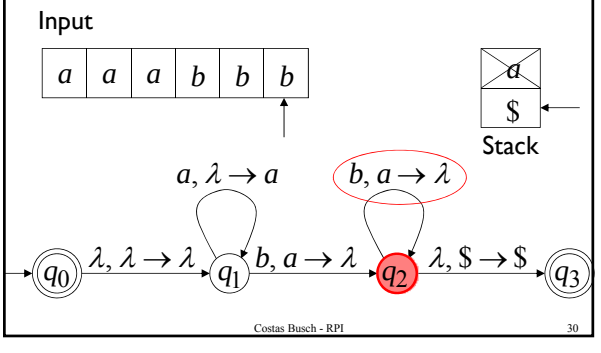
NPDA: Non-Deterministic PDA

Execution Example: Time 6



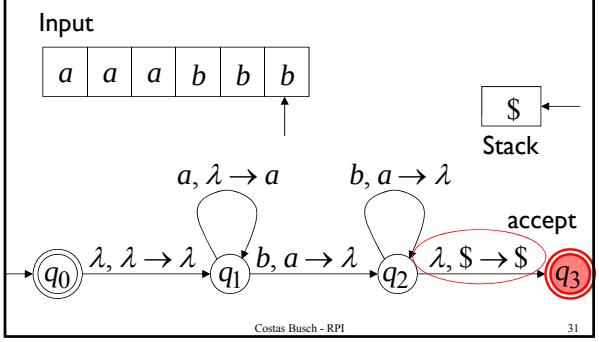
NPDA: Non-Deterministic PDA

Execution Example: Time 7



NPDA: Non-Deterministic PDA

Execution Example: Time 8

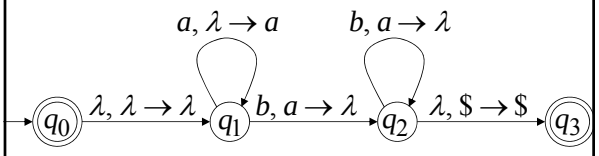


NPDA: Non-Deterministic PDA

- A string is accepted if there is a 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

NPDA: Non-Deterministic PDA

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

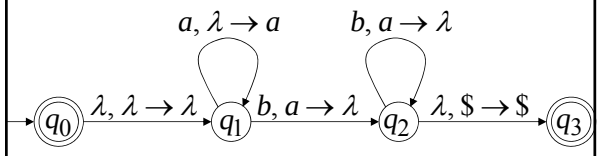


NPDA: Non-Deterministic PDA

In general,

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

is the language accepted by the NPDA:

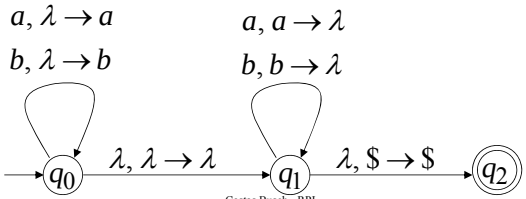


NPDA: Non-Deterministic PDA

Another Example:

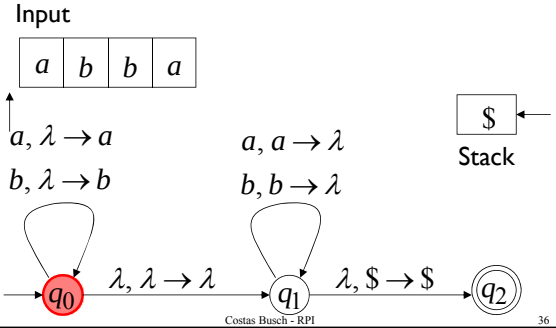
NPDA M

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



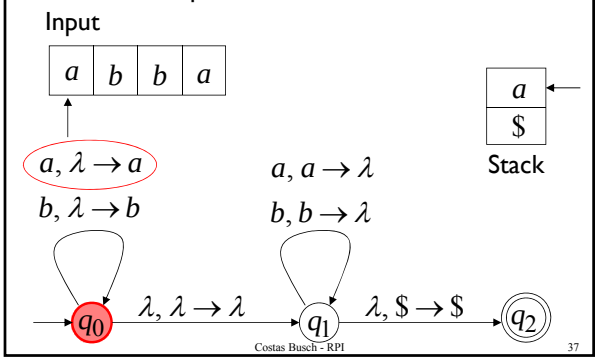
NPDA: Non-Deterministic PDA

Execution Example: Time 0



NPDA: Non-Deterministic PDA

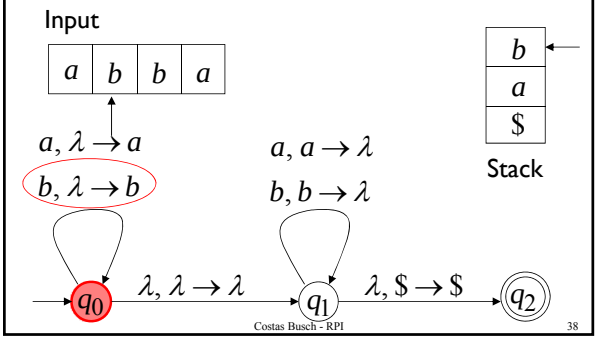
Execution Example: Time 1



Costas Busch - RPI 37

NPDA: Non-Deterministic PDA

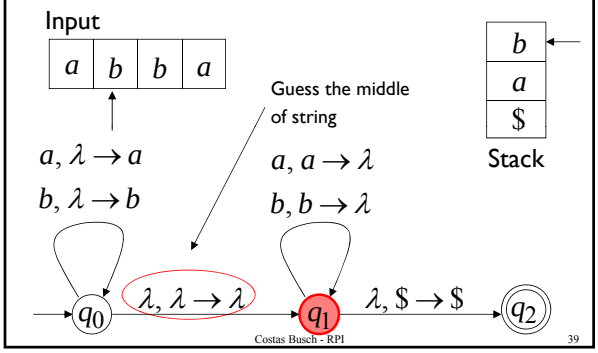
Execution Example: Time 2



Costas Busch - RPI 38

NPDA: Non-Deterministic PDA

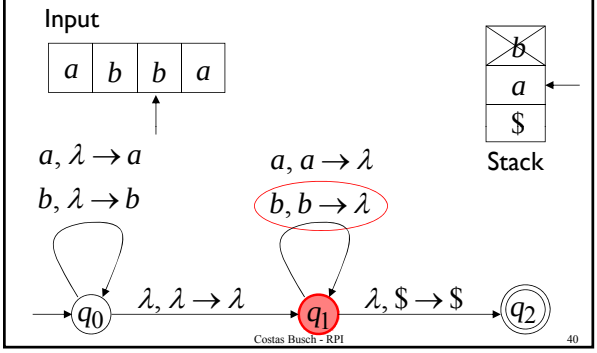
Execution Example: Time 3



Costas Busch - RPI 39

NPDA: Non-Deterministic PDA

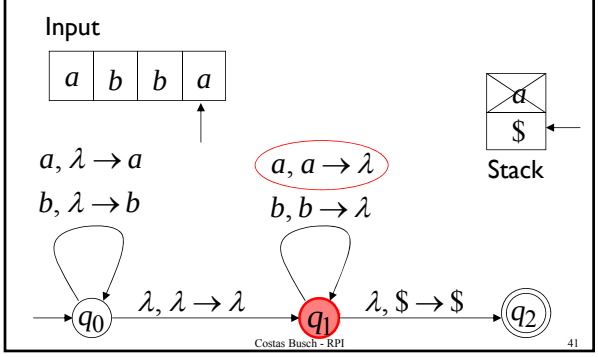
Execution Example: Time 4



Costas Busch - RPI 40

NPDA: Non-Deterministic PDA

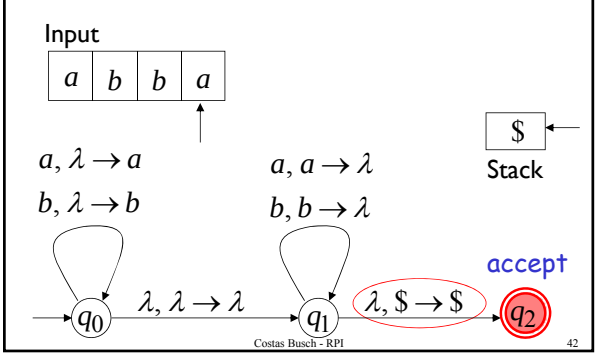
Execution Example: Time 5



Costas Busch - RPI 41

NPDA: Non-Deterministic PDA

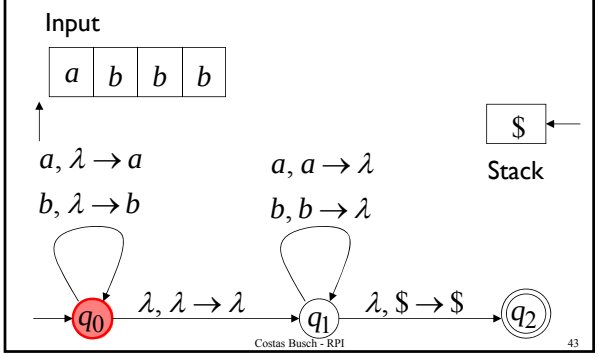
Execution Example: Time 6



Costas Busch - RPI 42

NPDA: Non-Deterministic PDA

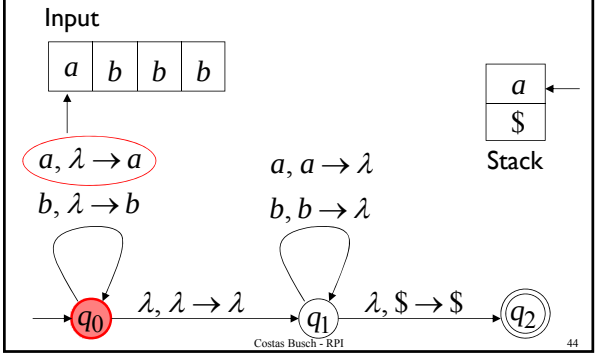
Rejection Example: Time 0



Costas Busch - RPI 43

NPDA: Non-Deterministic PDA

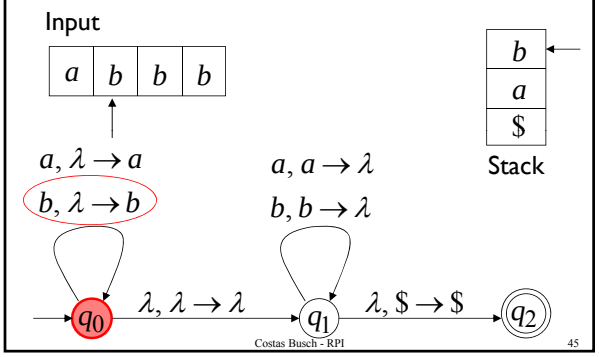
Rejection Example: Time 1



Costas Busch - RPI 44

NPDA: Non-Deterministic PDA

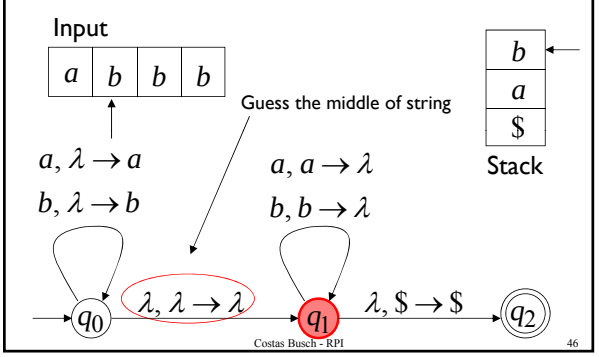
Rejection Example: Time 2



Costas Busch - RPI 45

NPDA: Non-Deterministic PDA

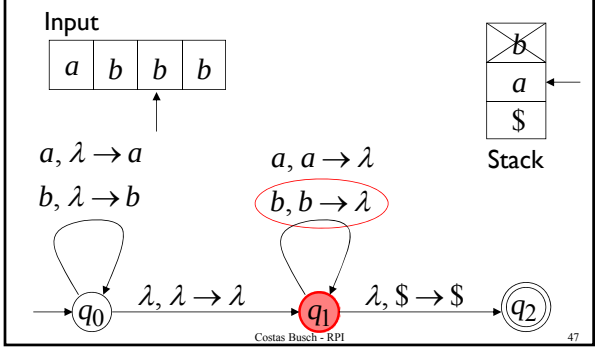
Rejection Example: Time 2



Costas Busch - RPI 46

NPDA: Non-Deterministic PDA

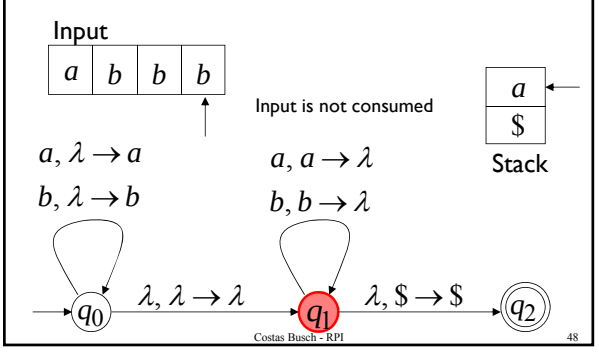
Rejection Example: Time 4



Costas Busch - RPI 47

NPDA: Non-Deterministic PDA

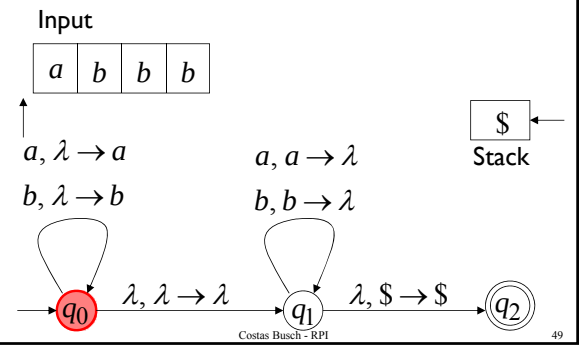
There is no possible transition



Costas Busch - RPI 48

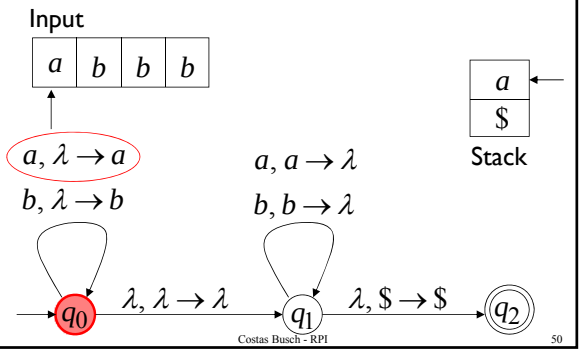
NPDA: Non-Deterministic PDA

Another computation on same string: Time 0



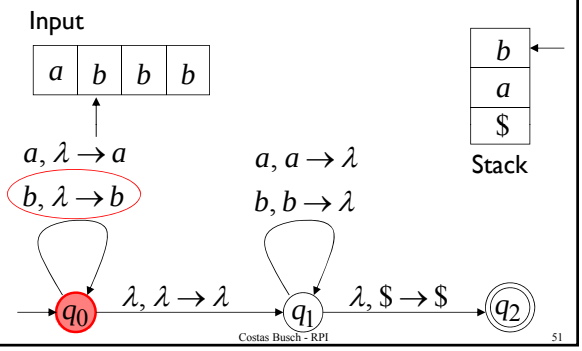
NPDA: Non-Deterministic PDA

Another computation on same string: Time 1



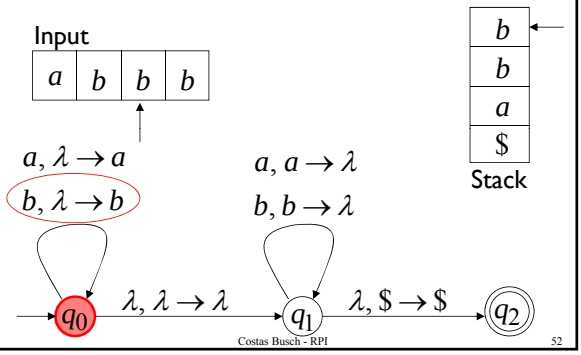
NPDA: Non-Deterministic PDA

Another computation on same string: Time 2



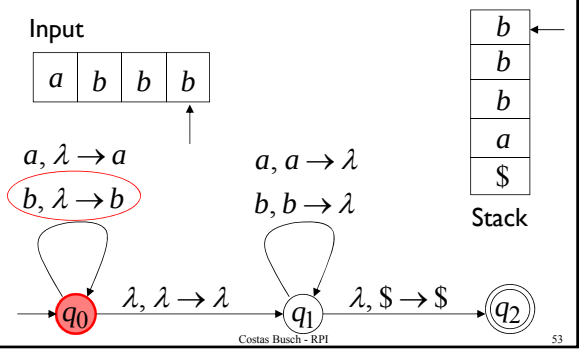
NPDA: Non-Deterministic PDA

Another computation on same string: Time 3



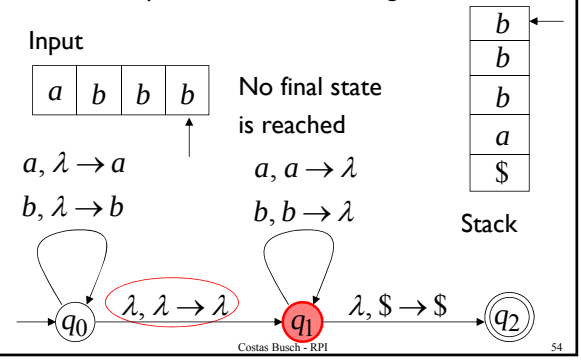
NPDA: Non-Deterministic PDA

Another computation on same string: Time 4



NPDA: Non-Deterministic PDA

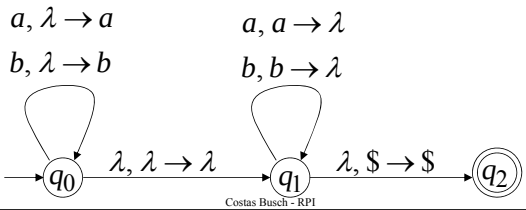
Another computation on same string: Time 5



NPDA: Non-Deterministic PDA

There is no computation that accepts string $abbb$

$$abbb \notin L(M)$$



NPDA: Non-Deterministic PDA

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

NPDA: Non-Deterministic PDA

A string is rejected if in every computation with this string:

- The input cannot be consumed

OR

- The input is consumed and the last state is not a final state

OR

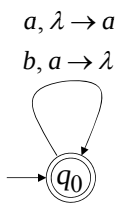
- The stack head moves below the bottom of the stack

NPDA: Non-Deterministic PDA

Another NPDA Example:

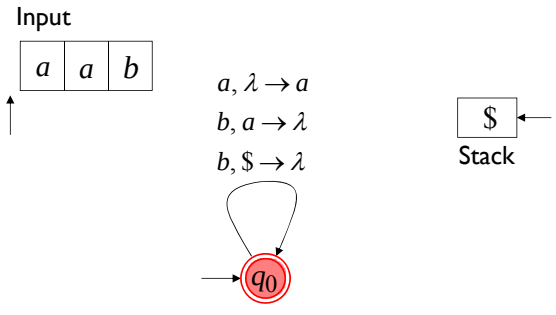
NPDA M

$$L(M) = \{a^n b^m \ : \ n \geq m-1\}$$



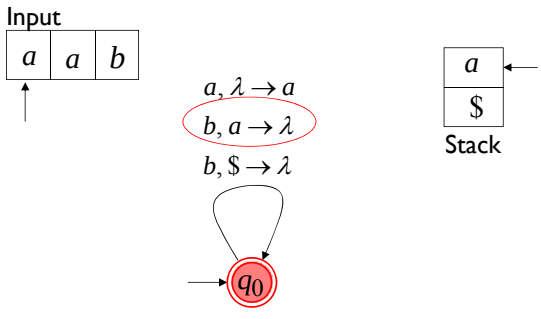
NPDA: Non-Deterministic PDA

Execution Example: Time 0



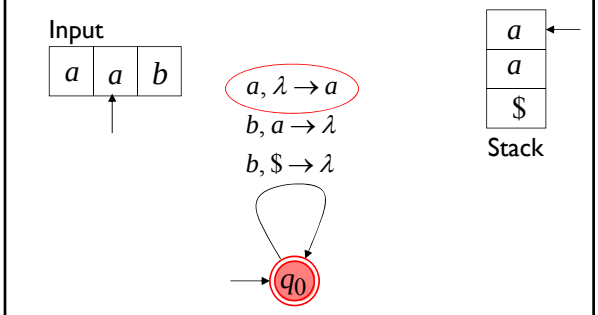
NPDA: Non-Deterministic PDA

Execution Example: Time 1



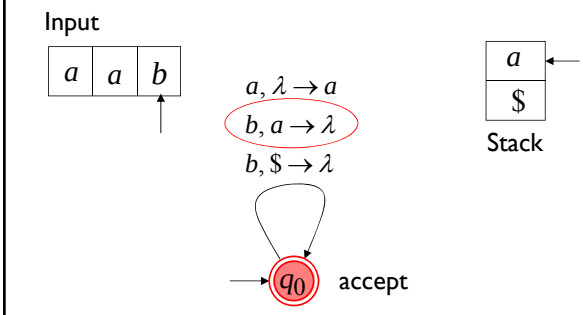
NPDA: Non-Deterministic PDA

Execution Example: Time 2



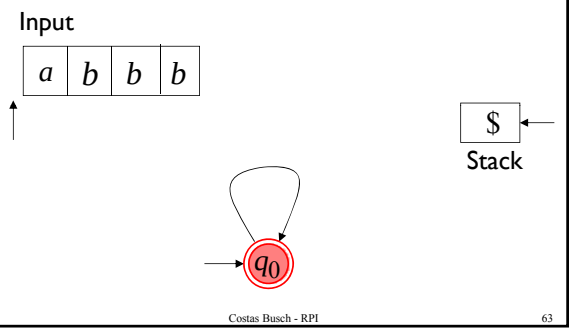
NPDA: Non-Deterministic PDA

Execution Example: Time 2



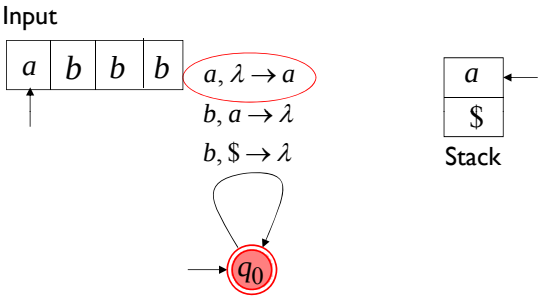
NPDA: Non-Deterministic PDA

Rejection Example: Time 0



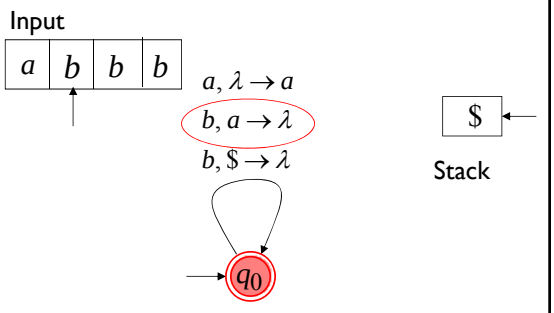
NPDA: Non-Deterministic PDA

Rejection Example: Time 1



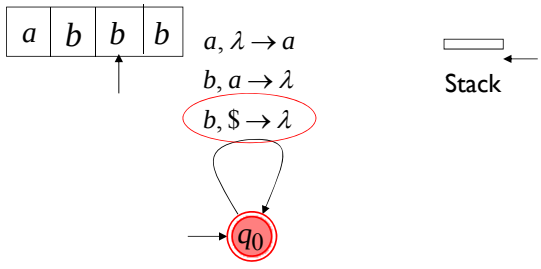
NPDA: Non-Deterministic PDA

Rejection Example: Time 2



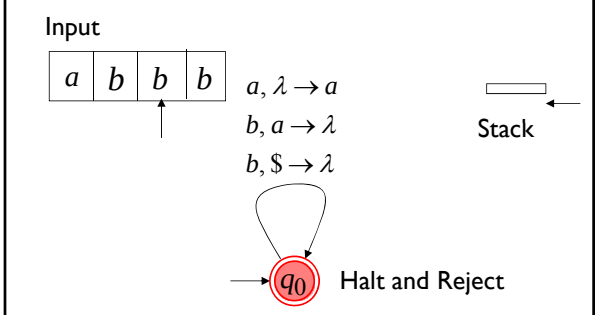
NPDA: Non-Deterministic PDA

Rejection Example: Time 3

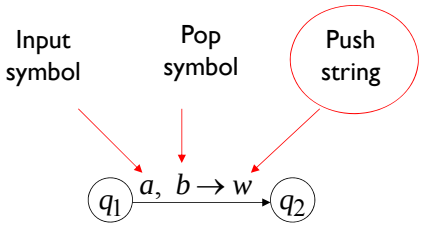


NPDA: Non-Deterministic PDA

Rejection Example: Time 4

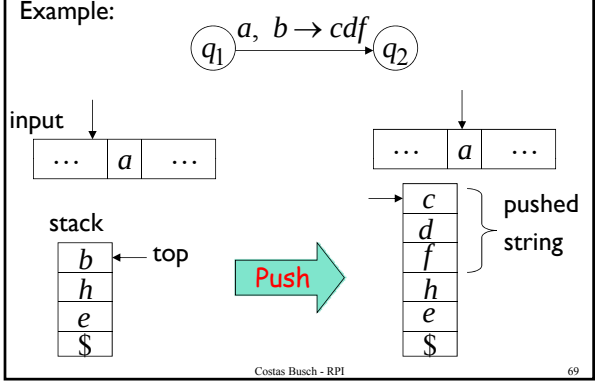


Pushing Strings



NPDA: Non-Deterministic PDA

Example:



NPDA: Non-Deterministic PDA

Another Example:

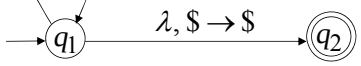
NPDA M

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

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

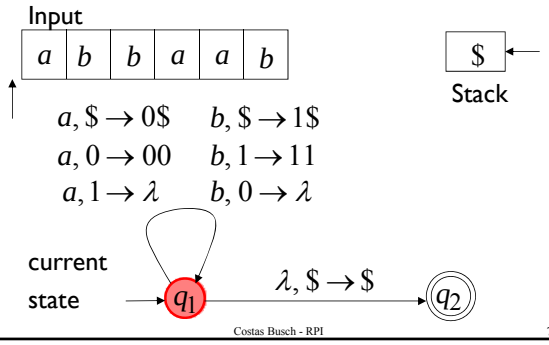
$a, 0 \to 00 \quad b, 1 \to 11$

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



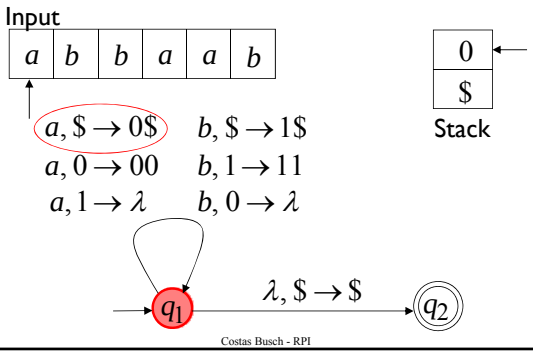
NPDA: Non-Deterministic PDA

Execution Example: Time 0



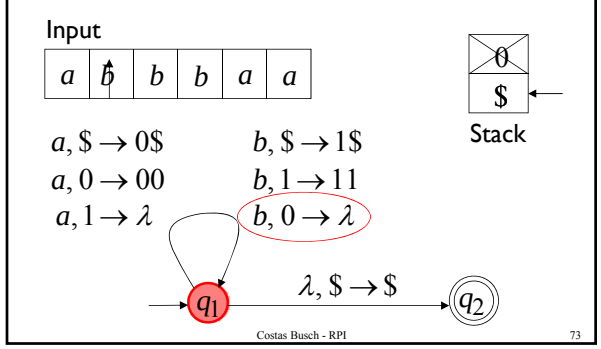
NPDA: Non-Deterministic PDA

Execution Example: Time 1



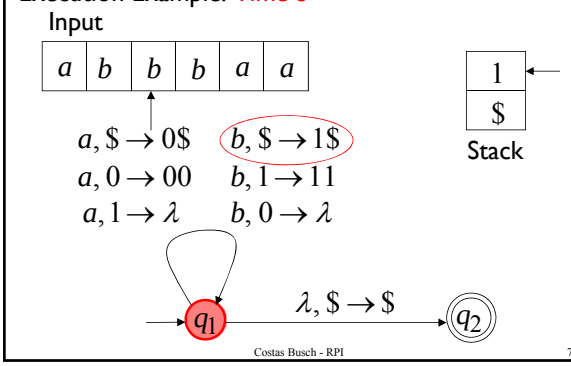
NPDA: Non-Deterministic PDA

Execution Example: Time 2



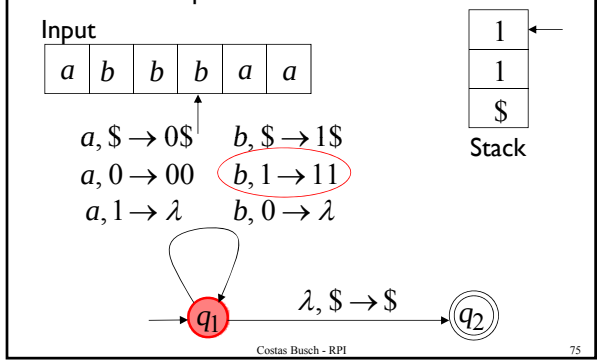
NPDA: Non-Deterministic PDA

Execution Example: Time 3



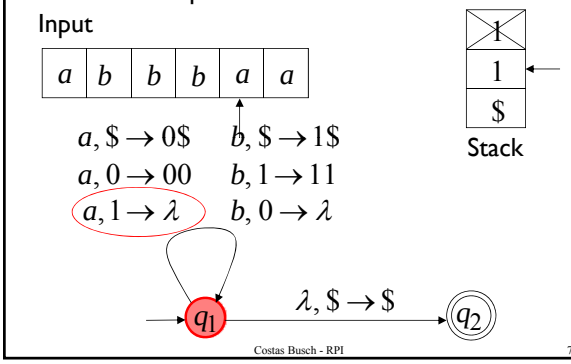
NPDA: Non-Deterministic PDA

Execution Example: Time 4



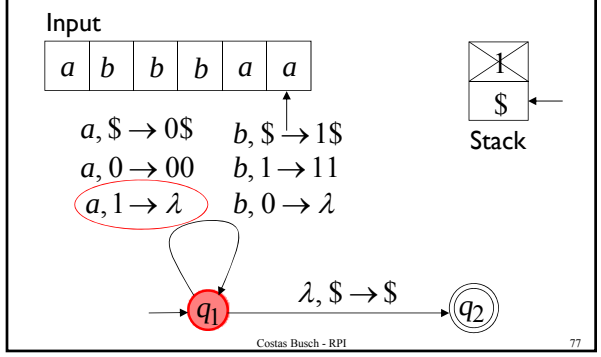
NPDA: Non-Deterministic PDA

Execution Example: Time 5



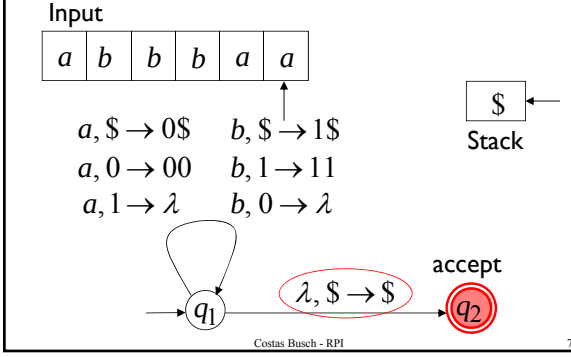
NPDA: Non-Deterministic PDA

Execution Example: Time 6



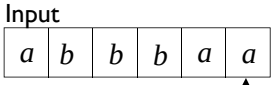
NPDA: Non-Deterministic PDA

Execution Example: Time 7



Execution Example: Time 8

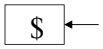
Input



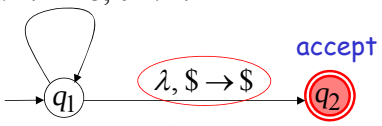
$a, \$ \to 0\$$ $b, \$ \to 1\$$

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

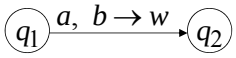
$a, 1 \to \lambda$ $b, 0 \to \lambda$



Stack

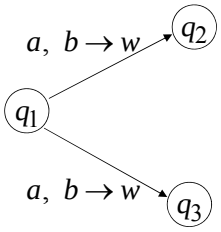


Formalities for NPDAs



Transition function:

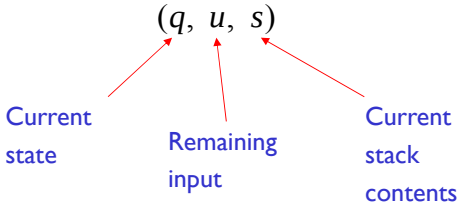
$$\delta(q_1, a, b) = \{(q_2, w)\}$$



Transition function:

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

Instantaneous Description

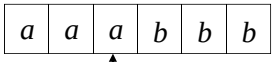


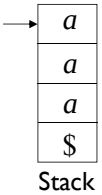
Instantaneous Description

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

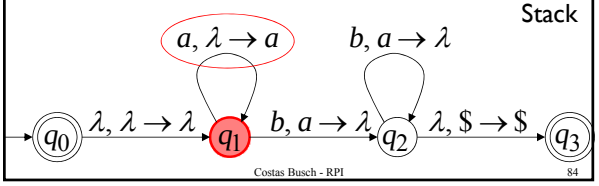
Time 4:

Input





Stack

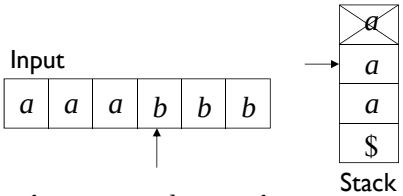


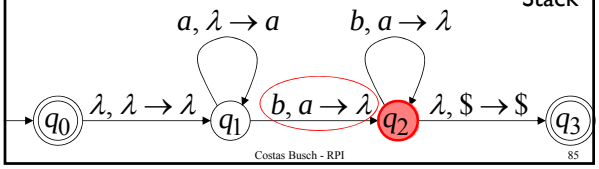
Instantaneous Description

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

Time 5:

Input





Instantaneous Description

We write:

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

Time 4 Time 5

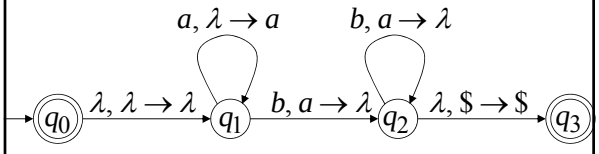
Instantaneous Description

A Computation:

$(q_0, aaabbb, \$) \succ (q_1, aaabbb, \$) \succ$

$(q_1, aabbb, a\$) \succ (q_1, abbb, aa\$) \succ (q_1, bbb, aaa\$) \succ$

$(q_2, bb, aa\$) \succ (q_2, b, a\$) \succ (q_2, \lambda, \$) \succ (q_3, \lambda, \$)$



Instantaneous Description

A Computation:

$(q_0, aaabbb, \$) \succ (q_1, aaabbb, \$) \succ$

$(q_1, aabbb, a\$) \succ (q_1, abbb, aa\$) \succ (q_1, bbb, aaa\$) \succ$

$(q_2, bb, aa\$) \succ (q_2, b, a\$) \succ (q_2, \lambda, \$) \succ (q_3, \lambda, \$)$

For convenience we write:

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

Formal Definition

Language $L(M)$ of NPDA: M

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

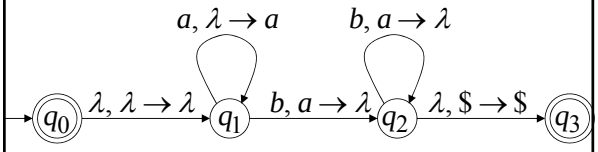
Initial state Final state

Instantaneous Description

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

$aaabbb \in L(M)$

NPDA M:

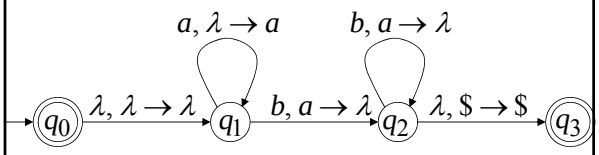


Instantaneous Description

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

$\Downarrow$

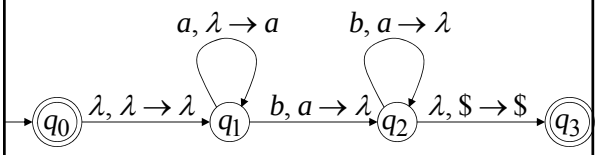
NPDA M: $a^n b^n \in L(M)$



Instantaneous Description

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

NPDA M:



Examples of PDAs

- Design a PDA whose language is
  $\{a^m b^n c^n d^m \mid m \geq 0, n \geq 0\}$.
- Design a PDA whose language is
  $\{a^m b^m c^n d^n \mid m \geq 0, n \geq 0\}$.
- Design a PDA whose language is
  $\{a^m b^n c^p d^q \mid m + n = p + q\}$.
- Design a PDA whose language is
  $\{a^m b^n c^k \mid m = n \text{ or } m = k\}$.