Theory of Automata

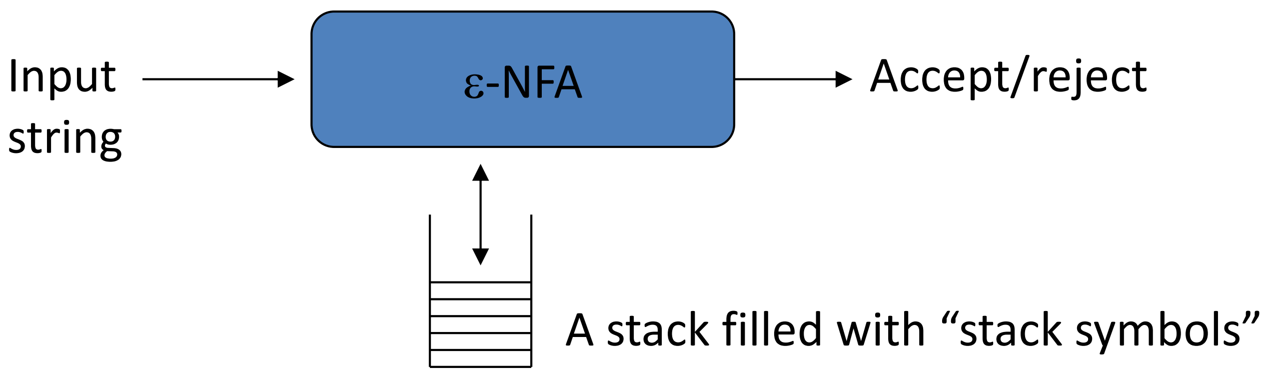
Pushdown Automata

Dr. Sabina Akhtar

Revision

PDA - the automata for CFLs

- What is?
  - FA to Reg Lang,   PDA is to CFL
- PDA == [ε -NFA + “a stack”]
- Why a stack?



Pushdown Automata - Definition

- A PDA P := (Q,∑,Γ, δ,q_0,Z_0,F):
  - Q: states of the ε-NFA
  - ∑: input alphabet
  - Γ : stack symbols
  - δ: transition function
  - q_0: start state
  - Z_0: Initial stack top symbol
  - F: Final/accepting states

δ : The Transition Function

input symb.

old state *Stack top* *new state(s)* *new Stack top(s)*

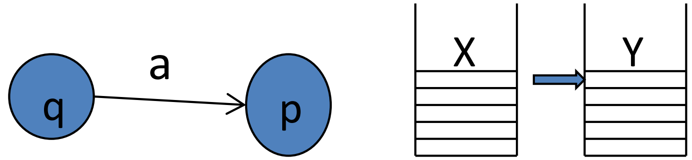
$$\delta : \; Q \times \Sigma \times \Gamma \Rightarrow Q \times \Gamma$$

δ : The Transition Function

δ(q,a,X) = {(p,Y), …}

Non-determinism

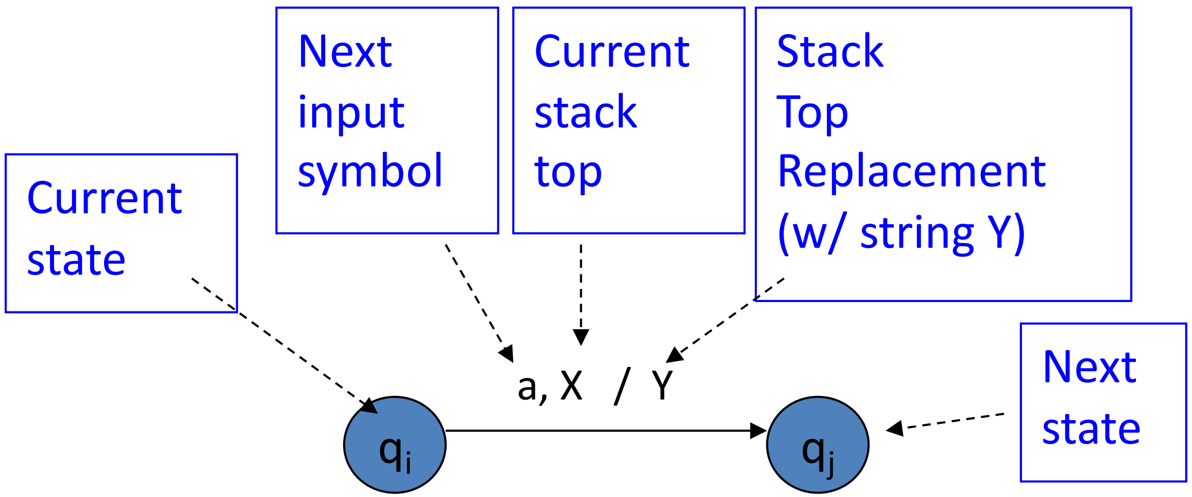
1. state transition from q to p
2. a is the next input symbol
3. X is the current stack *top* symbol
4. Y is the replacement for X; it is in Γ^* (a string of stack symbols)
   - i. Set Y = ε for: Pop(X)
   - ii. If Y=X: stack top is unchanged
   - iii. If $Y=Z_1Z_2 \ldots Z_k$: X is popped and is replaced by Y in reverse order (i.e., Z_1 will be the new stack top)



	Y = ?	Action
i)	$Y=\varepsilon$	Pop(X)
ii)	Y=X	Pop(X) Push(X)
iii)	$Y=Z_1Z_2..Z_k$	Pop(X) Push(Z_k) Push(Z_{k-1}) … Push(Z_2) Push(Z_1)

PDA as a state diagram

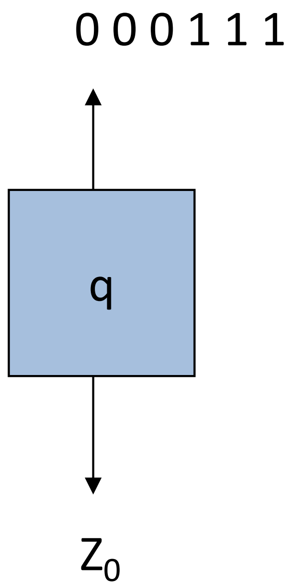
$\delta(q_i, a, X) = \{(q_j, Y)\}$



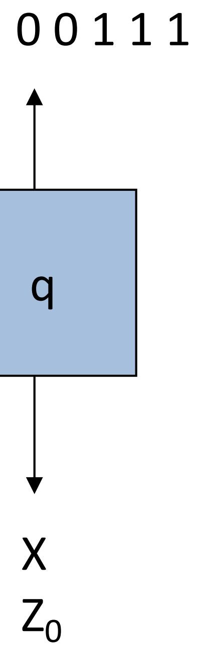
Class Activity

- Design a PDA to accept $\{0^n 1^n \mid n \geq 1\}$.
- The states:
  - q = start state. We are in state q if we have seen only 0's so far.
  - p = we've seen at least one 1 and may now proceed only if the inputs are 1's.
  - f = final state; accept.

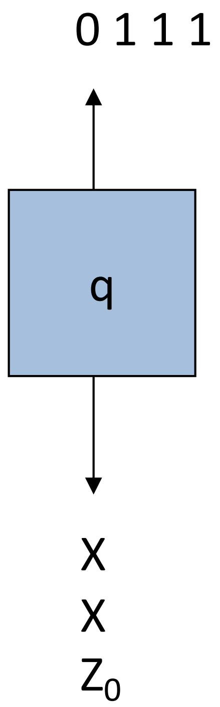
Actions of the Example PDA



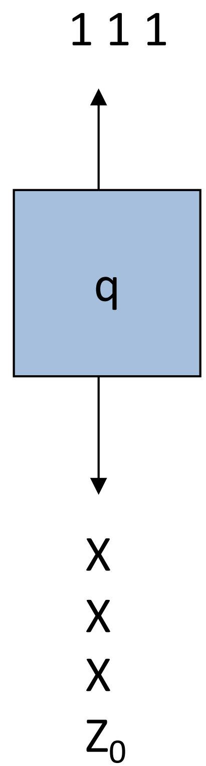
Actions of the Example PDA



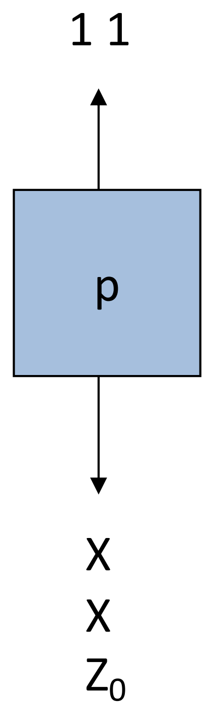
Actions of the Example PDA



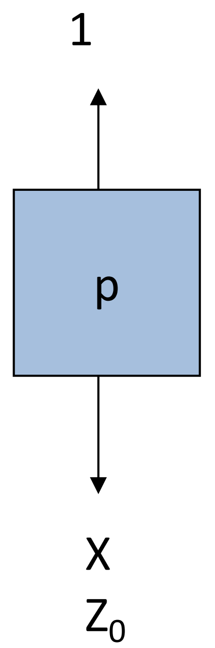
Actions of the Example PDA



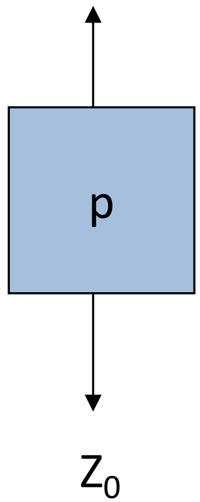
Actions of the Example PDA



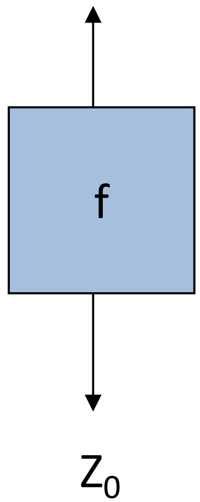
Actions of the Example PDA



Actions of the Example PDA



Actions of the Example PDA



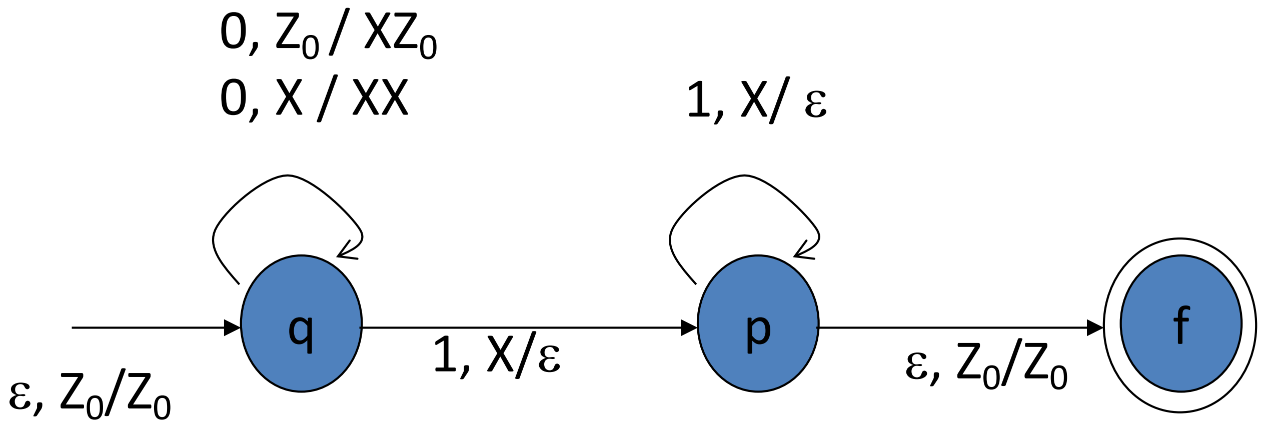
Example: PDA – (2)

- The stack symbols:
  - Z_0 = start symbol. Also marks the bottom of the stack, so we know when we have counted the same number of 1's as 0's.
  - X = marker, used to count the number of 0's seen on the input.

Example: PDA – (3)

- The transitions:
  - $\delta(q, 0, Z_0) = \{(q, XZ_0)\}$.
  - $\delta(q, 0, X) = \{(q, XX)\}$. These two rules cause one X to be pushed onto the stack for each 0 read from the input.
  - $\delta(q, 1, X) = \{(p, \epsilon)\}$. When we see a 1, go to state p and pop one X.
  - $\delta(p, 1, X) = \{(p, \epsilon)\}$. Pop one X per 1.
  - $\delta(p, \epsilon, Z_0) = \{(f, Z_0)\}$. Accept at bottom.

Solution



0, Z_0 / XZ_0
0, X / XX
1, X/ ε
ε, Z_0/Z_0
q
1, X/ε
p
ε, Z_0/Z_0
f

There are two types of PDAs that one can design:
those that accept by final state or by empty stack

Acceptance by...

- ***PDAs that accept by final state:***
  - For a PDA P, the language accepted by P, denoted by L(P) by *final state*, is:
    - $\{w \mid (q_0,w,Z_0) \mid\!\text{---}^* (q,\varepsilon, A)\}$, s.t., $q \in F$

Checklist:
- input exhausted?
- in a final state?

- ***PDAs that accept by empty stack:***
  - For a PDA P, the language accepted by P, denoted by N(P) by *empty stack*, is:
    - $\{w \mid (q_0,w,Z_0) \mid\!\text{---}^* (q, \varepsilon, \varepsilon)\}$, for any $q \in Q$.

Q) Does a PDA that accepts by empty stack need any final state specified in the design?

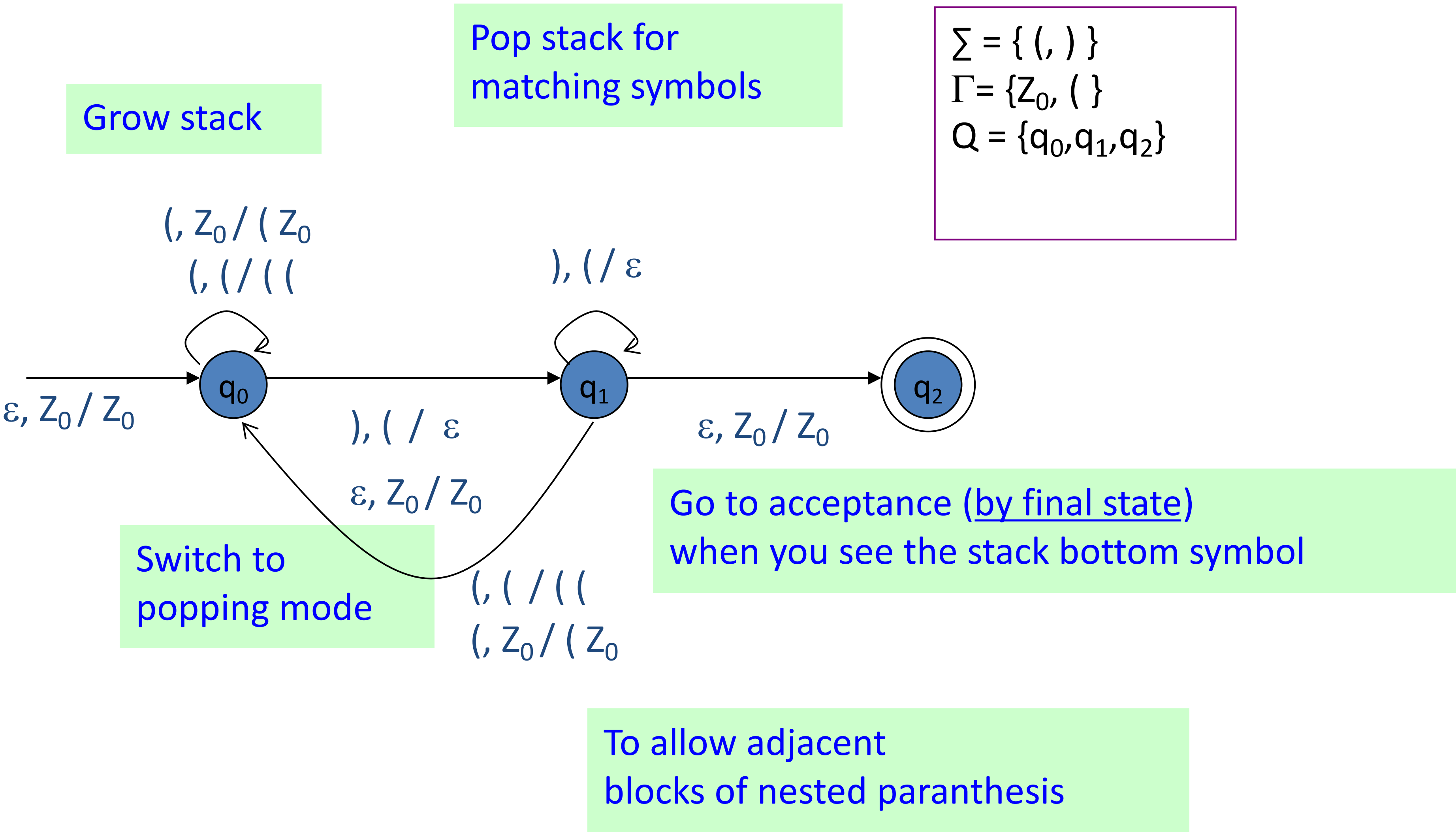
Checklist:
- input exhausted?
- is the stack empty?

Class Activity

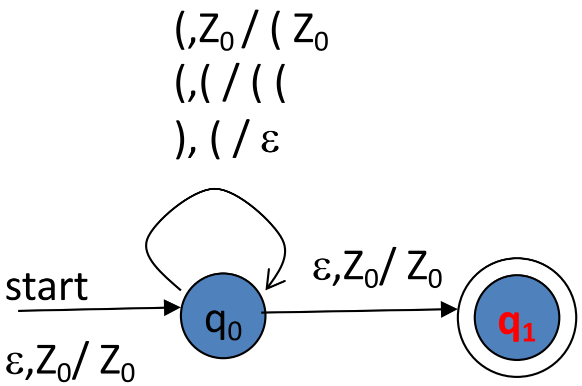
- Design PDA for the language of balanced paranthesis with acceptance by final state and by empty stack. Given 7 tuple specifications.

Example 2: language of balanced paranthesis

$\Sigma = \{ (,) \}$
$\Gamma = \{Z_0, (\}$
$Q = \{q_0, q_1, q_2\}$



Example 2: language of balanced paranthesis (another design)

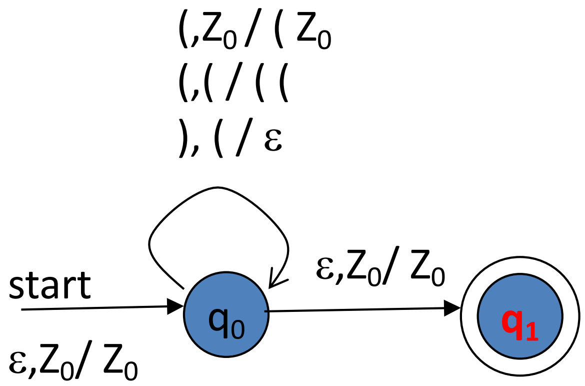


Σ = { (,) }
Γ= {Z_0, (}
Q = {q_0,q_1}

Example: L of balanced parenthesis

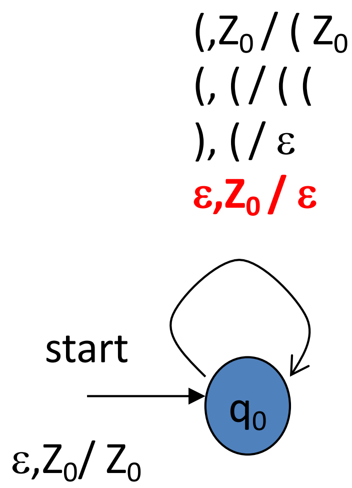
PDA that accepts by final state

P_F:



An equivalent PDA that accepts by empty stack

P_N:



How will these two PDAs work on the input: ((( )) ( )) ( )

Class Activity

- Design PDA for

$$L = \{0^n 1^m 0^m 1^n \mid n, m \geq 1\}$$

Provide formal specifications as well.

Class Activity

- Design PDA for

L = {w | w contains equal number of 1's and 0's}

NON DETERMINISTIC PDA

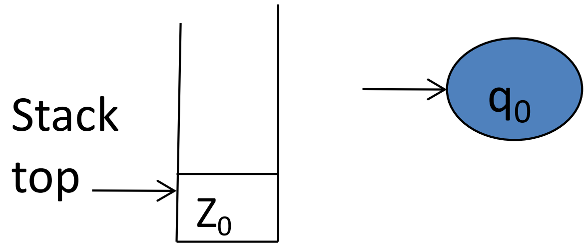
Example

Let $L_{wwr} = \{ww^R \mid w \text{ is in } (0+1)^*\}$

- CFG for L_{wwr} : $S ==> 0S0 \mid 1S1 \mid \varepsilon$
- PDA for L_{wwr} :
- $P := (Q, \sum, \Gamma, \delta, q_0, Z_0, F)$

  $= (\{q_0, q_1, q_2\}, \{0,1\}, \{0,1,Z_0\}, \delta, q_0, Z_0, \{q_2\})$

Initial state of the PDA:



PDA for L_{wwr}

1. $\delta(q_0,0, Z_0)=\{(q_0,0Z_0)\}$
2. $\delta(q_0,1, Z_0)=\{(q_0,1Z_0)\}$

First symbol push on stack

3. $\delta(q_0,0, 0)=\{(q_0,00)\}$
4. $\delta(q_0,0, 1)=\{(q_0,01)\}$
5. $\delta(q_0,1, 0)=\{(q_0,10)\}$
6. $\delta(q_0,1, 1)=\{(q_0,11)\}$

Grow the stack by pushing new symbols on top of old (w-part)

7. $\delta(q_0, \varepsilon, 0)=\{(q_1, 0)\}$
8. $\delta(q_0, \varepsilon, 1)=\{(q_1, 1)\}$
9. $\delta(q_0, \varepsilon, Z_0)=\{(q_1, Z_0)\}$

Switch to popping mode, nondeterministically (boundary between w and w^R)

10. $\delta(q_1,0, 0)=\{(q_1, \varepsilon)\}$
11. $\delta(q_1,1, 1)=\{(q_1, \varepsilon)\}$

Shrink the stack by popping matching symbols (w^R-part)

12. $\delta(q_1, \varepsilon, Z_0)=\{(\mathbf{q_2}, Z_0)\}$

Enter acceptance state

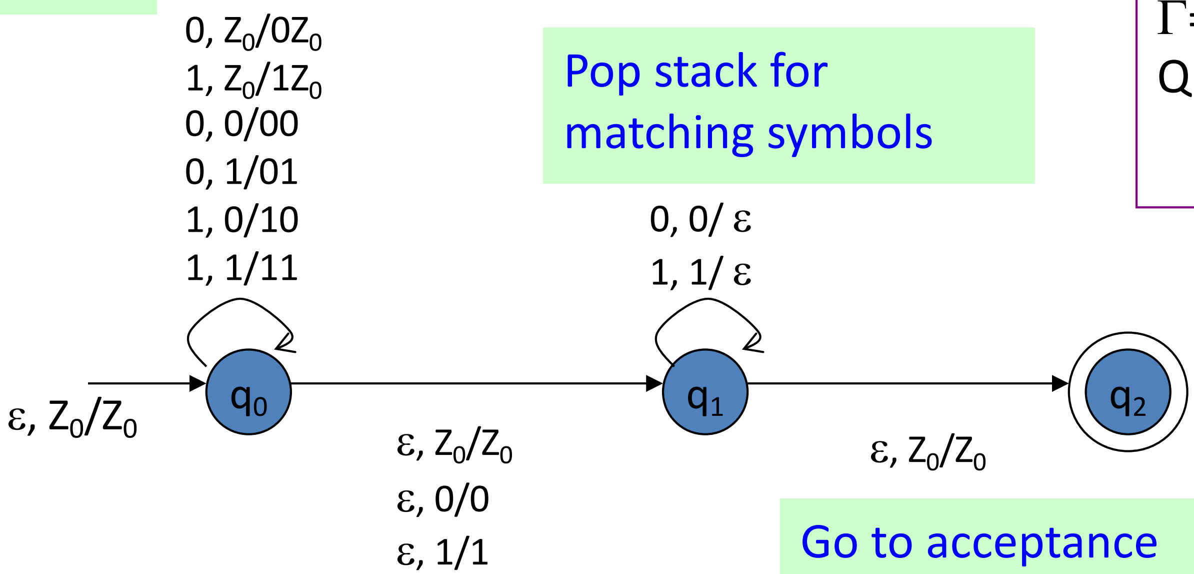
Draw the Transition diagram

PDA for L_{wwr}: Transition Diagram

Grow stack

Pop stack for matching symbols

$\Sigma = \{0, 1\}$
$\Gamma = \{Z_0, 0, 1\}$
$Q = \{q_0, q_1, q_2\}$



Go to acceptance

Switch to popping mode

This would be a non-deterministic PDA

PDA's Instantaneous Description (ID)

A PDA has a configuration at any given instance: **(q,w,y)**

- q - current state
- w - remainder of the input (i.e., unconsumed part)
- y - current stack contents as a string from top to bottom of stack

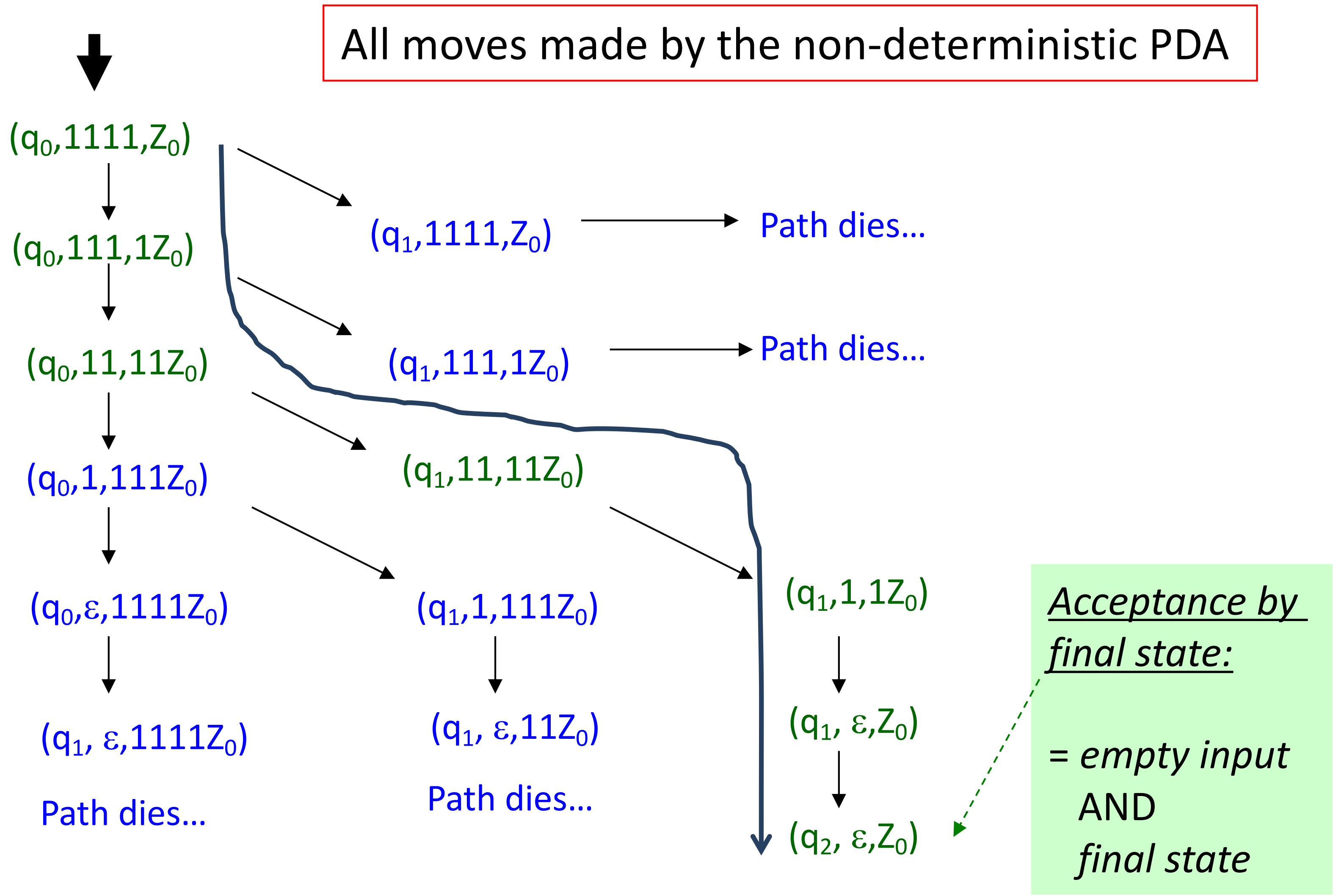
If δ(q,a, X)={(p, A)} is a transition, then the following are also true:

- (q, a, X) |--- (p,ε,A)
- (q, aw, XB) |--- (p,w,AB)

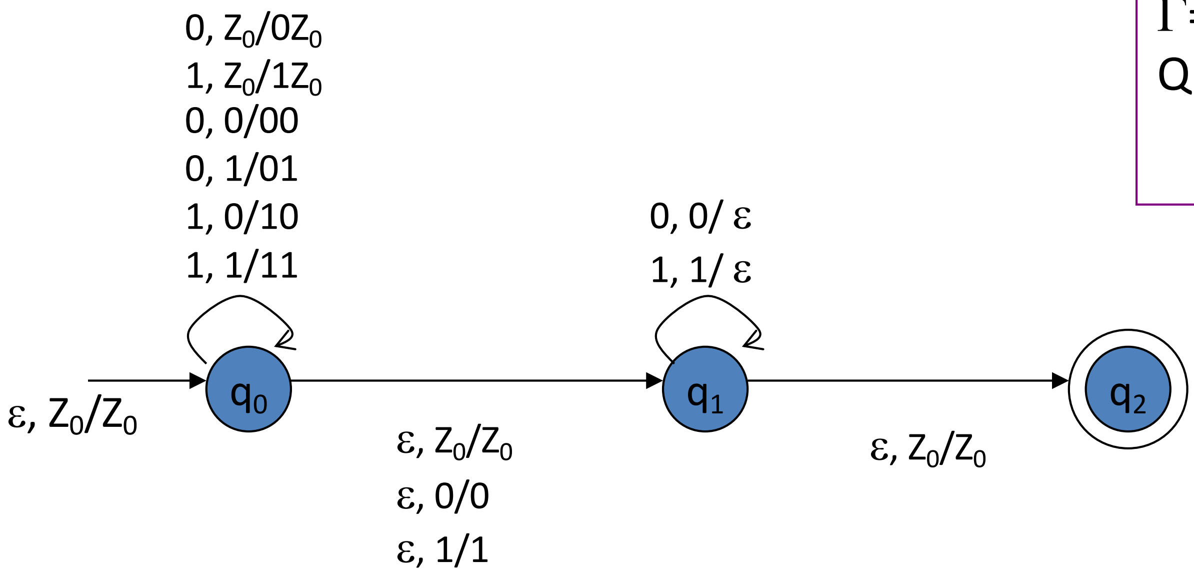
|--- sign is called a "turnstile notation" and represents one move

|---* sign represents a sequence of moves

How does the PDA for L_{wwr} work on input "1111"?



PDA for L_{wwr}: Transition Diagram



$\sum = \{0, 1\}$
$\Gamma = \{Z_0, 0, 1\}$
$Q = \{q_0, q_1, q_2\}$

How will this PDA work on the input: 01100110
Show all the reachable ID's.

Class Activity

- Design PDA for

  $L = \{w \mid w \text{ is an odd length palindrome}\}$

Deterministic PDA's

- To be deterministic, there must be at most one choice of move for any state q, input symbol *a*, and stack symbol X.
- In addition, there must not be a choice between using input ϵ or real input.
- Formally, δ(q, a, X) and δ(q, ϵ, X) cannot both be nonempty.

References

- Book Chapter
- Lectures from Stanford University
  - http://infolab.stanford.edu/~ullman/ialc/spr10/spr10.html#LECTURE%20NOTES
- Lectures from Washington State University
  - http://www.eecs.wsu.edu/~ananth/CptS317/Lectures/