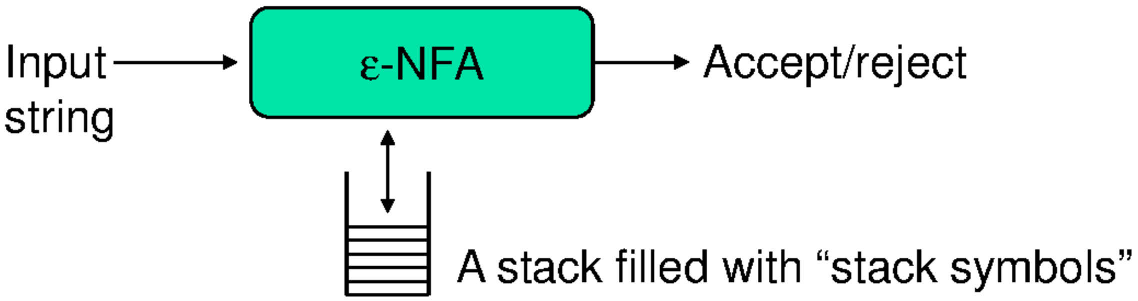
# PDA - the automata for CFLs

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



# Pushdown Automata - Definition

- A PDA P := ( $Q, \Sigma, \Gamma, \delta, q_0, Z_0, F$ ):
  - 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

*old state* *Stack top* *input symb.* *new state(s)* *new Stack top(s)*

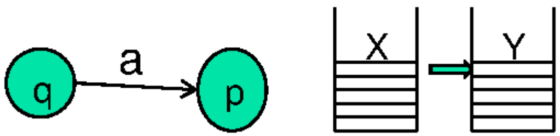
$$\delta : Q \times \Gamma \times \Sigma \Rightarrow Q \times \Gamma$$

# 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 $\Gamma^*$ (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=ε | Pop(X) |
| ii) | Y=X | Pop(X)<br>Push(X) |
| iii) | $Y=Z_1Z_2..Z_k$ | Pop(X)<br>Push($Z_k$)<br>Push($Z_{k-1}$)<br>…<br>Push($Z_2$)<br>Push($Z_1$) |

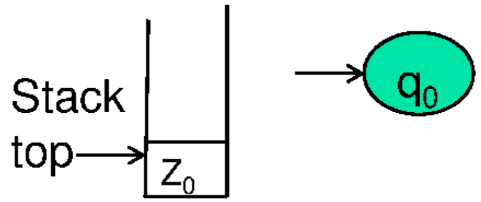
# Example

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

- CFG for $L_{wwr}$ : $S \Longrightarrow 0S0 \mid 1S1 \mid \varepsilon$
- 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 (boundary between w and $w^R$)

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

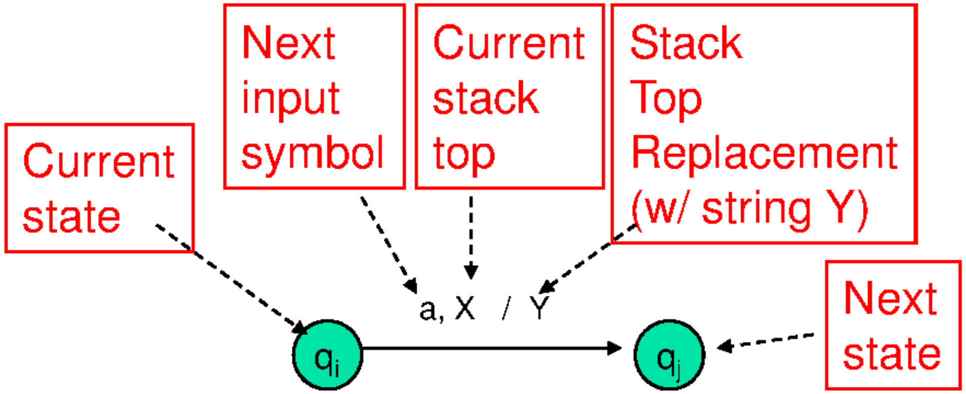
Shrink the stack by popping matching symbols ($w^R$-part)

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

Enter acceptance state

# PDA as a state diagram

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

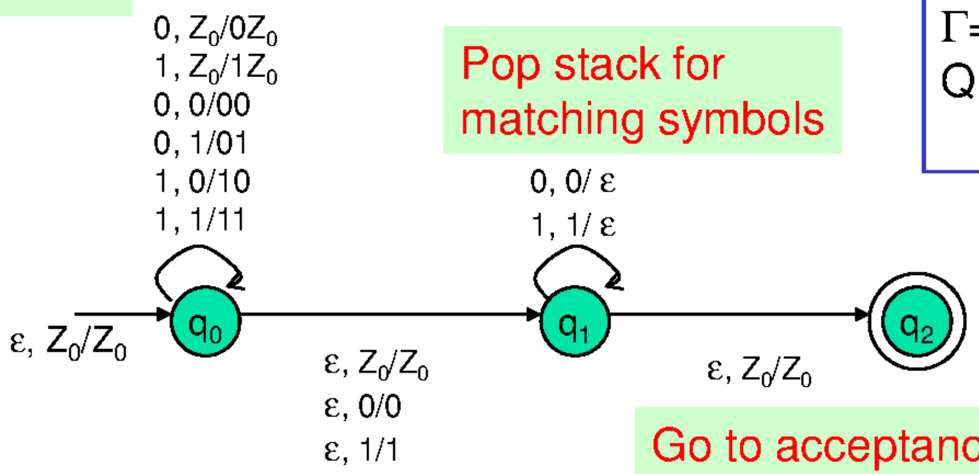


# PDA for $L_{wwr}$: Transition Diagram

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

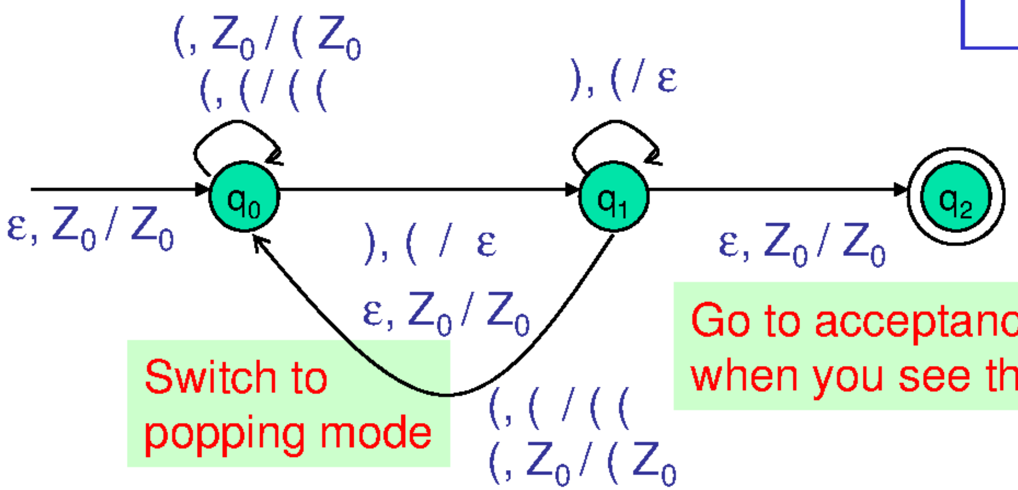
This would be a non-deterministic PDA

# Example 2: language of balanced paranthesis

Grow stack

Pop stack for matching symbols

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

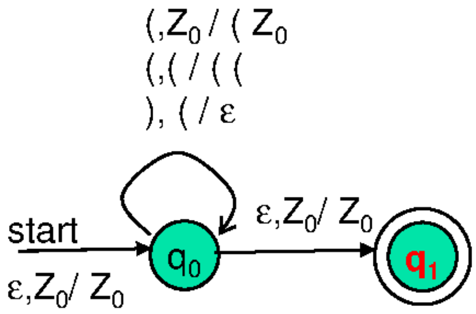



Switch to popping mode

Go to acceptance (<u>by final state</u>) when you see the stack bottom symbo

To allow adjacent blocks of nested paranthesis

# Example 2: language of balanced paranthesis (another design)




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

# 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:

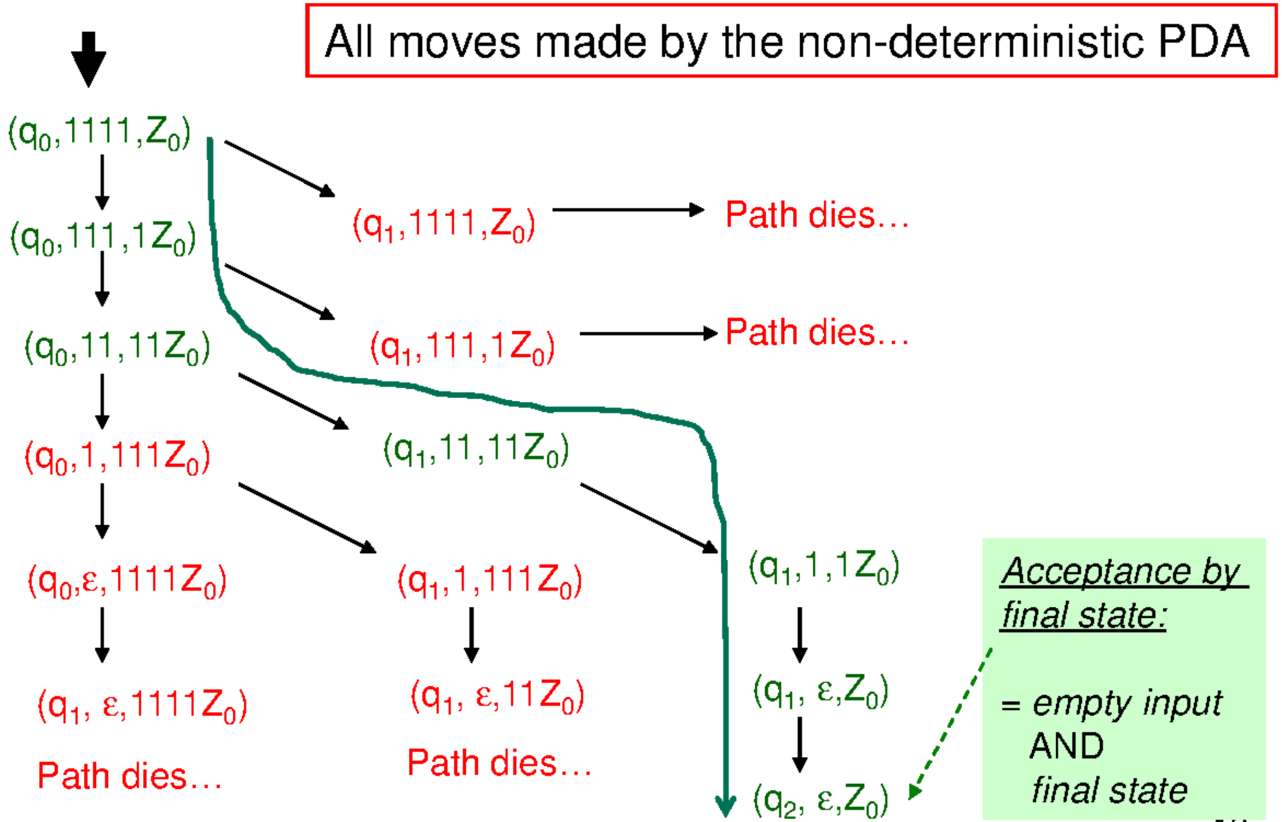
- (q, a, X ) |--- (p,ε,A)
- (q, aw, XB ) |--- (p,w,AB)

---

|--- 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"?



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:
    - $\{w \mid (q_0,w,Z_0) \vdash^* (q,\varepsilon, A)\}$, 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:
    - $\{w \mid (q_0,w,Z_0) \vdash^* (q, \varepsilon, \varepsilon)\}$, for any $q \in Q$.

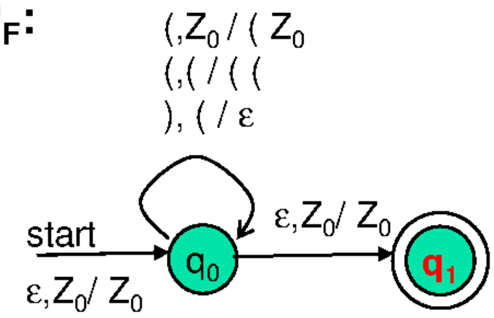
Q) Does a PDA that accepts by empty stack need any final state specified in the design?

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

# Example: L of balanced parenthesis

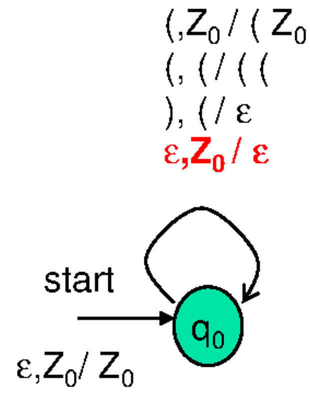
<u>PDA that accepts by final state</u>

**$P_F$:**




<u>An equivalent PDA that accepts by empty stack</u>

**$P_N$:**




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