

Chapter 6

Pushdown Automata

Pushdown Automata (PDA)

- PDA is a way to implement a context free grammar in a similar way we design finite automata for regular grammar
- PDA = Finite State Machine + a **Stack**
- Stack operations: **Push and POP**

Structure of Pushdown Automata

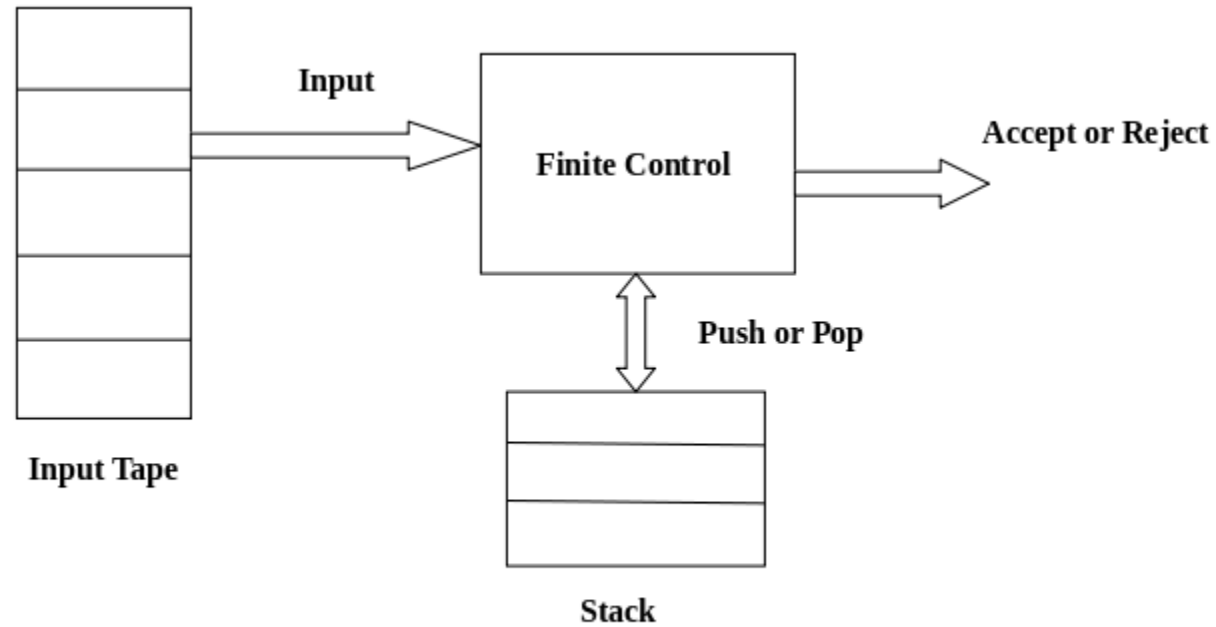


Fig: Pushdown Automata

Formal Definition of PDA

- $P = (Q, \Sigma, \Gamma, \delta, q_0, z_0, F)$
- Q = finite set of states
- Σ = finite set of input symbols
- Γ = A finite set of stack alphabet
- δ = the transition function
- q_0 = the start state
- z_0 = the start stack symbol
- F = the set of final states

Transition function

Input

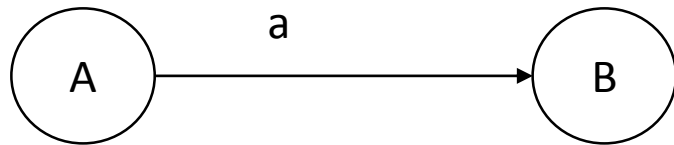
- δ takes argument as a triple where $\delta = (q, a, x)$
 - q is a state in Q
 - a is either an input symbol Σ or ϵ
 - x is a stack symbol, that is a member of Γ

Output

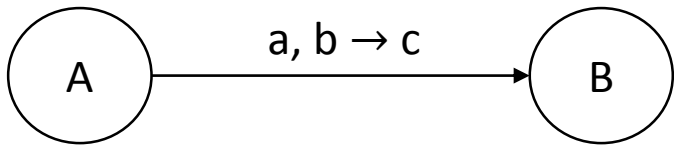
- Output of δ is a finite set of pairs (p, γ) where
- P is a new state
- γ is a string of stack symbols that replaces x at the top of the stack
- If
 - $\gamma = \epsilon$ then the stack is popped
 - $\gamma = x$ then the stack is unchanged
 - $\gamma = yz$ then x is replaced by z and y is pushed onto the stack.

Pushdown Automata (Graphical Notation)

- Finite State Machine



- Pushdown Automata



a = Input Symbol

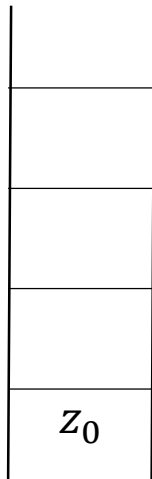
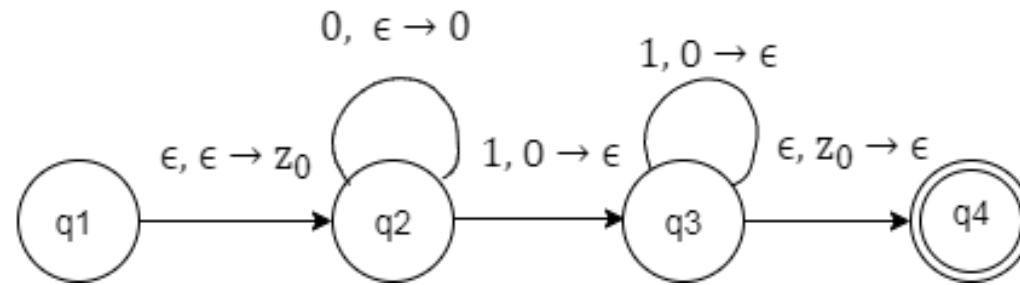
b = Symbol on the top of the stack. This symbol is popped, ϵ means the stack is neither read or popped

c = This symbol is pushed onto the stack, ϵ means nothing is pushed onto the stack

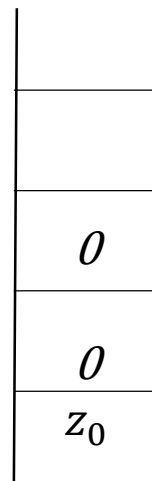
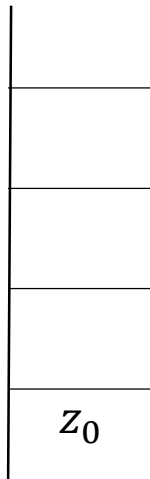
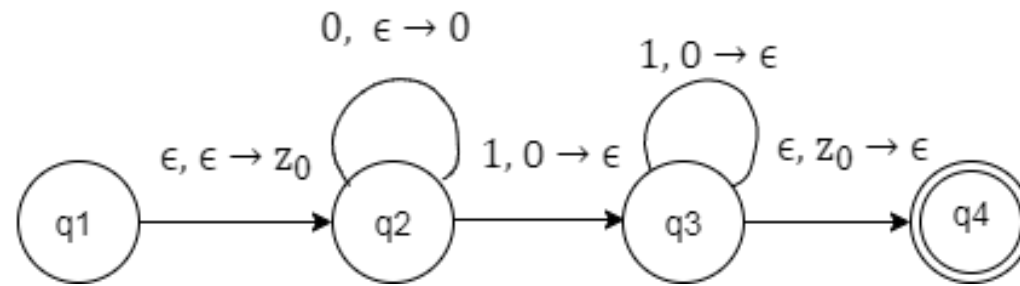
Construction of PDA

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

$$L = \{01, 0011, 0001111, \dots\}$$

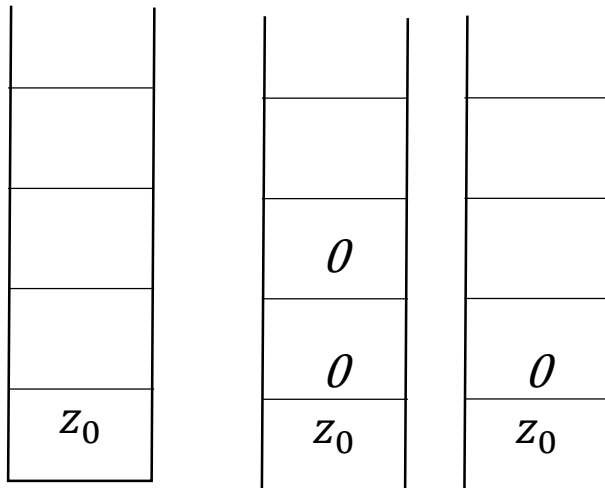
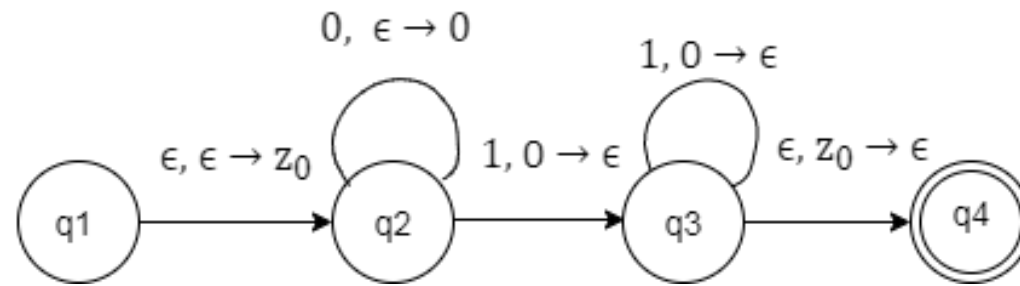


Construction of PDA



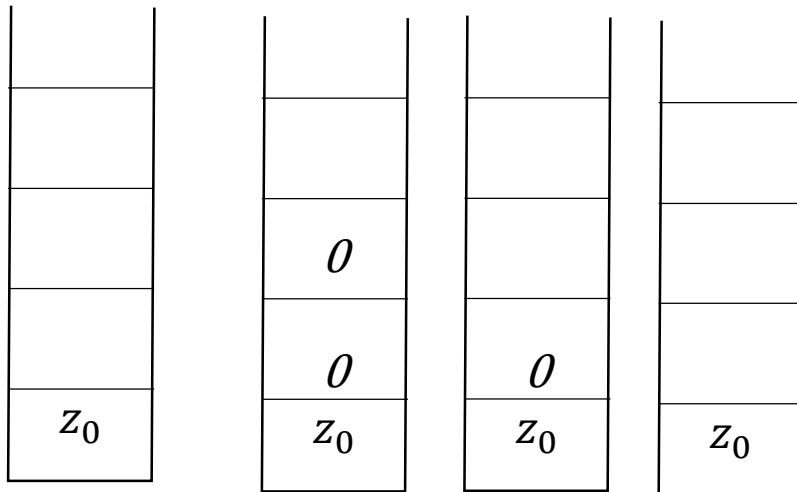
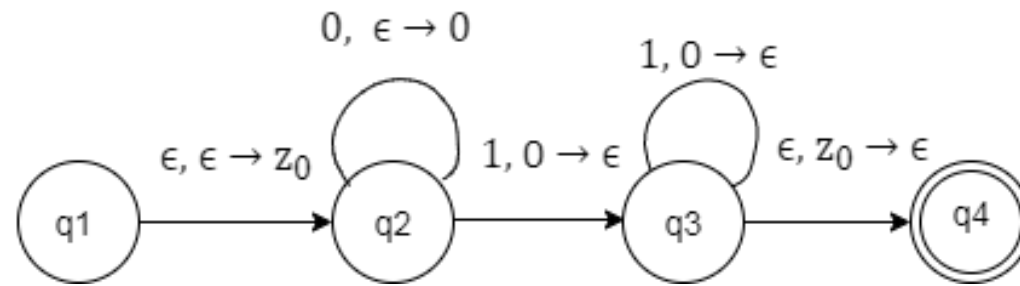
0011

Construction of PDA



0011

Construction of PDA

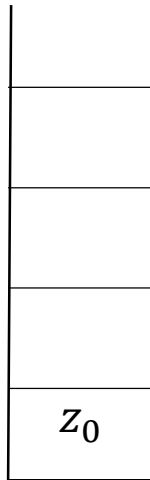
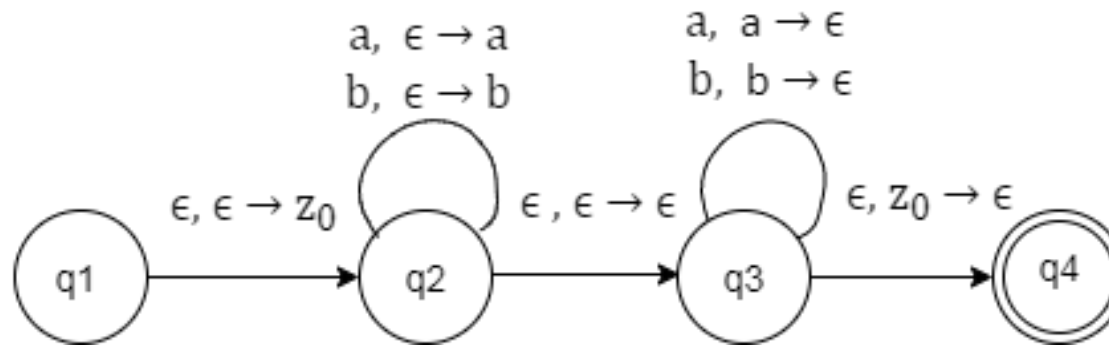


0011

Since we have found z_0 as the final stack element, the string is accepted

Construction of PDA (Even palindrome)

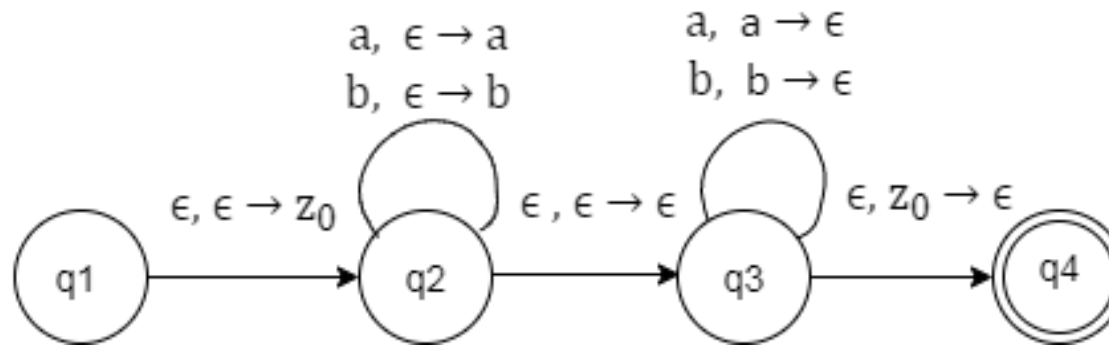
- $L = \{WW^R \mid W = (a + b)^+\}$



a b b a

Construction of PDA (Even palindrome)

- $L = \{WW^R \mid W = (a + b)^+\}$

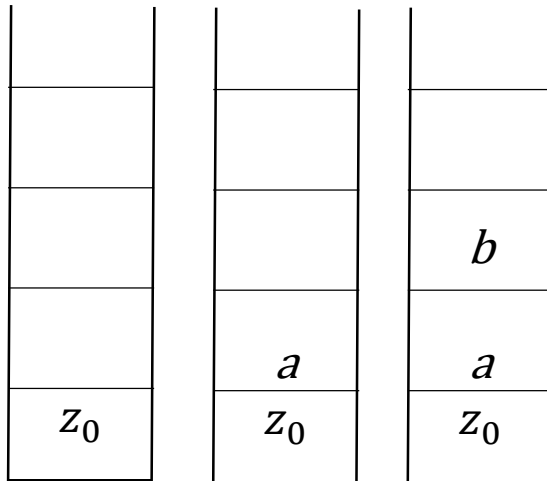
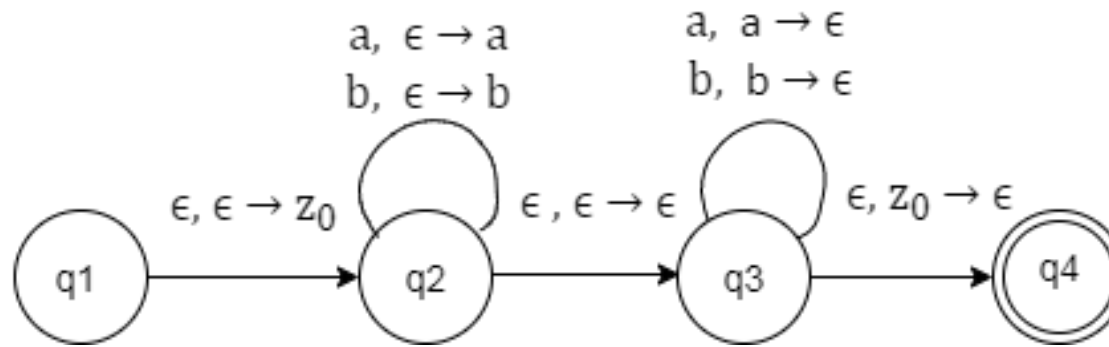


a b | b a

		<i>b</i>
	<i>a</i>	<i>a</i>
<i>z</i> ₀	<i>z</i> ₀	<i>z</i> ₀

Construction of PDA (Even palindrome)

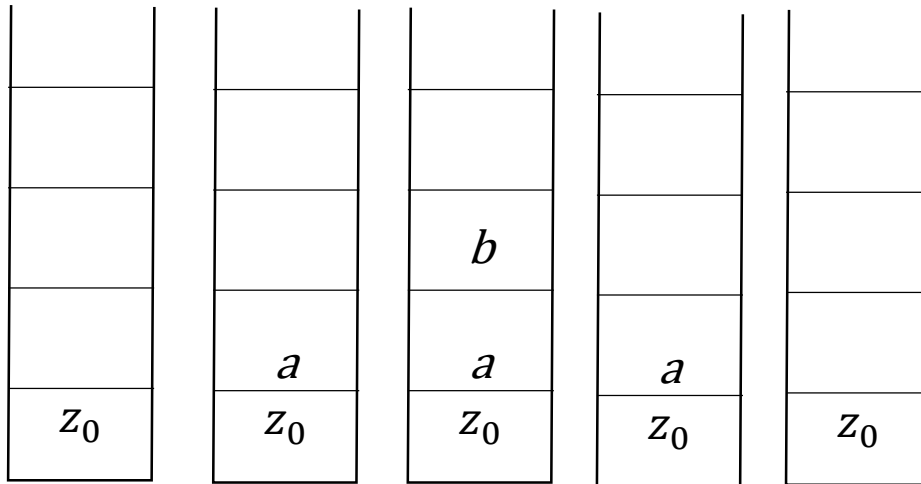
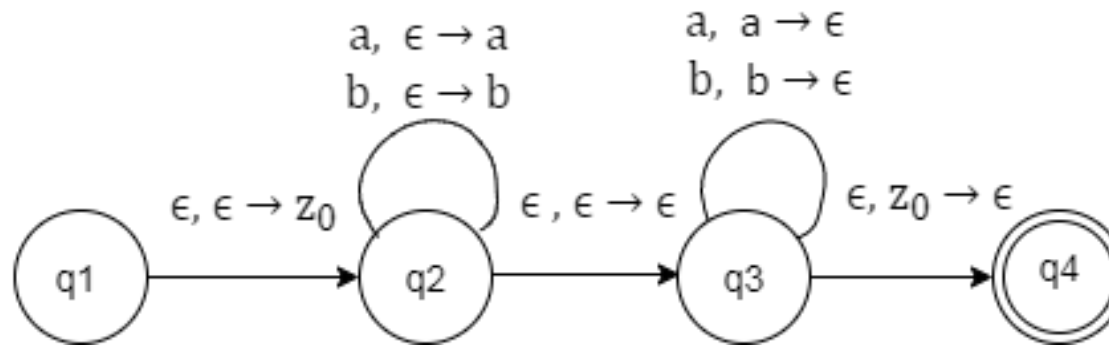
- $L = \{WW^R \mid W = (a + b)^+\}$



a b | b a

Construction of PDA (Even palindrome)

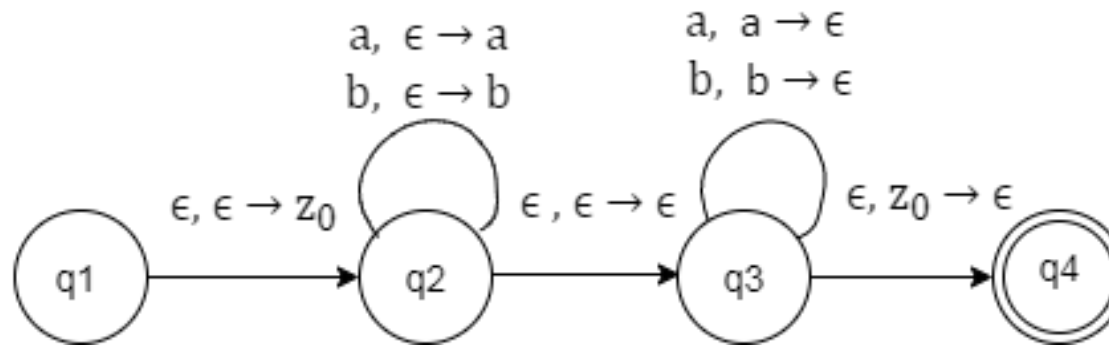
- $L = \{WW^R \mid W = (a + b)^+\}$



a b b a

Construction of PDA (Even palindrome)

- $L = \{WW^R \mid W = (a + b)^+\}$



a b a b

		<i>b</i>
	<i>a</i>	<i>a</i>
<i>z</i> ₀	<i>z</i> ₀	<i>z</i> ₀

After pushing *ab* we need to pop *a* from the stack, but the top of the stack does not contain *a*, As the PDA cannot pop the required value, so the string can not be accepted

Homework

- Design a PDA for odd palindrome