CS310

Pushdown Automata

Sections: 2.2

page 109

October 10, 2008

Quick Review

- (CFG) 4-tuple (V, ∑, R, S)
  - V finite set of variables
  - ∑ finite set of terminals

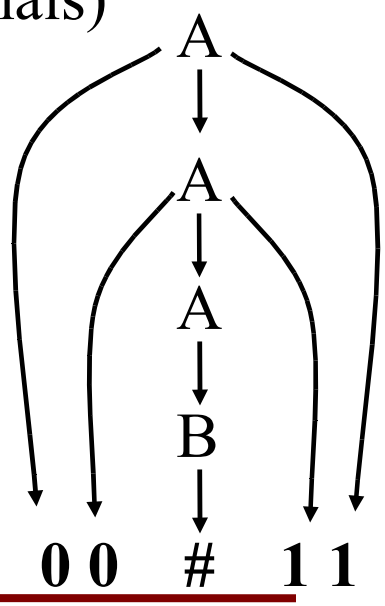
Example

A -> 0A1

A -> B

B -> #

  - R set of rules of form:
    - variable -> (string of variables and terminals)
  - S ϵ V, start variable
  - L(G) = { w ϵ ∑* | S -*> w}
    - w is in ∑* and can be derived from S



Chomsky Normal Form

- CNF presents a grammar in a standard, simplified form:

  A -> BC

  A -> *a*

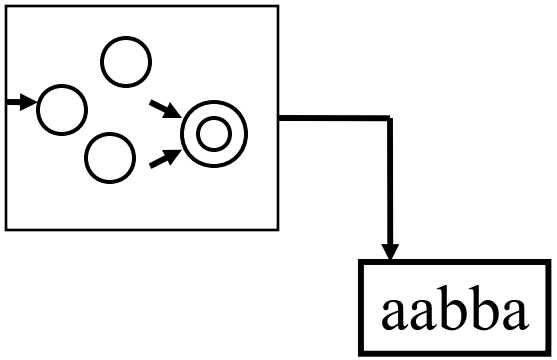
  S -> ε

  - Where A,B,C are variables
    - B and C are not the start variable
  - *a* is a terminal
  - The rule S -> ε is allowed so the language can generate the empty string (optional)

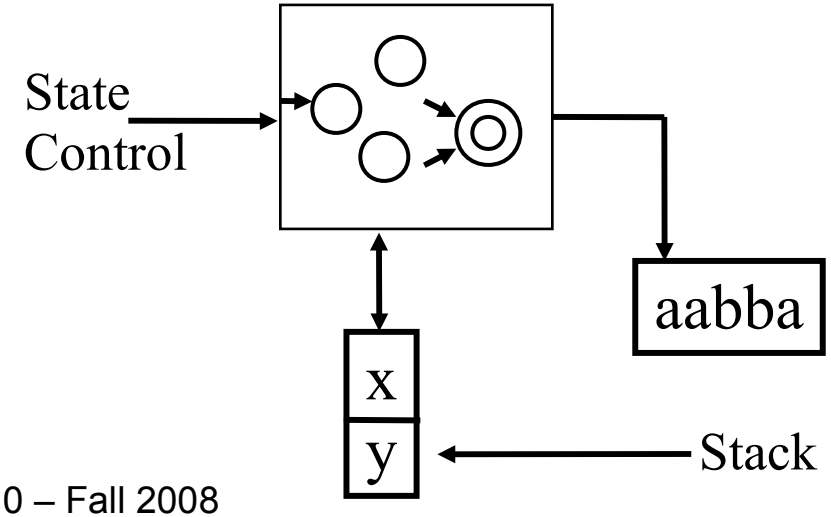
Pushdown Automata

- Machine to recognize Context Free Language
- Similar to an NFA, but contains a *stack*
  - An FA with memory added (LIFO!)

FA



Pushdown Automata



Pushdown Automata

- PDA may be deterministic or nondeterministic
  - Not equivalent! (unlike DFA & NFA)

- Define certain (state, input) to push data onto the stack
- Combine input string with stack data for δ

Pushdown Automata (Informally)

S -> X

X -> (X) | XX | ε

What language? Regular?

How would you solve this problem using a stack (forget the Pushdown Automata)?

Formal Definition

- 6-tuple!
  - Q: set of states
  - Σ: input alphabet
  - Γ: stack alphabet
  - δ: $Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow P(Q \times \Gamma_\varepsilon)$
    - input and top of stack to transition
    - Do not read or write from stack: $\Gamma_\varepsilon = \varepsilon$
  - $q_0 \in Q$: start state
  - $F \subseteq Q$: set of accept states

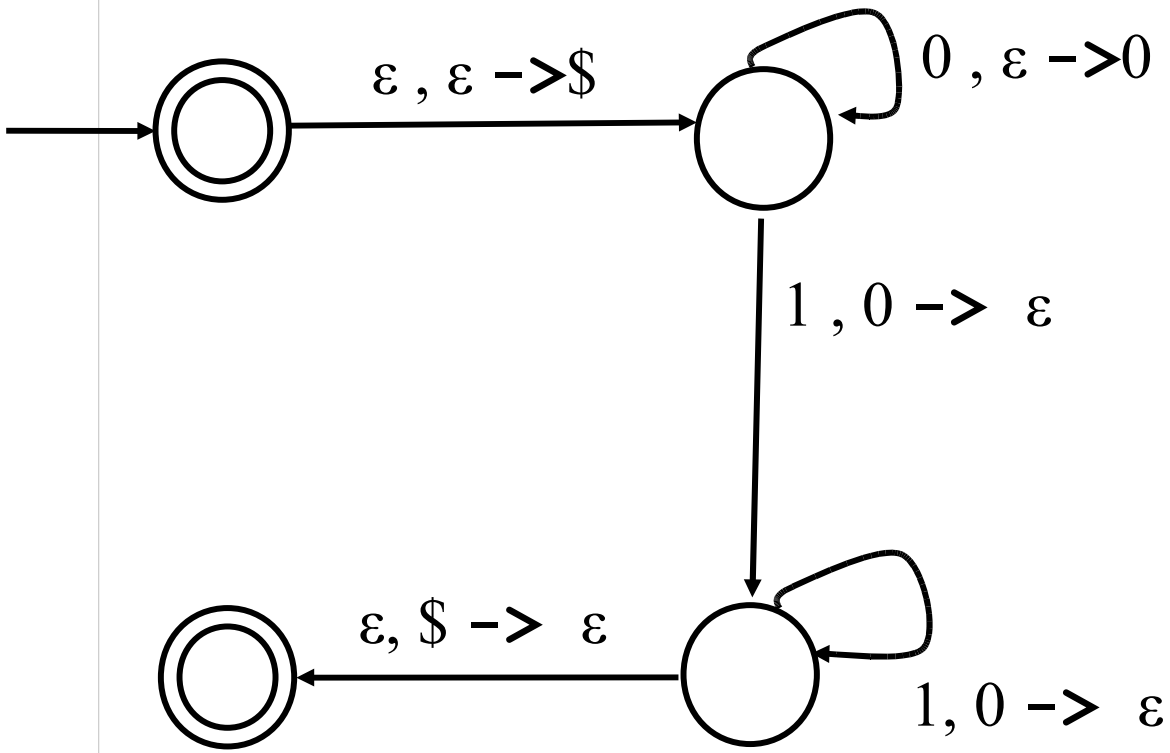
Example (Non-deterministic)

- $\{ 0^n 1^n \mid n \geq 0 \}$
  - q1 start state
  - $ special symbol

Input	0			1			ε		
Stack	0	$	ε	0	$	ε	0	$	ε
q1	ø	ø	ø	ø	ø	ø	ø	ø	{(q2,$)}
q2	ø	ø	{(q2,0)}	{(q3, ε)}	ø	ø	ø	ø	ø
q3	ø	ø	ø	{(q3, ε)}	ø	ø	ø	{(q3, ε)}	ø
q4	ø	ø	ø	ø	ø	ø	ø	ø	ø

Example

- Alternate notation:



a , b ->c

Read a from input, read b from stack, push c onto stack to take this transition

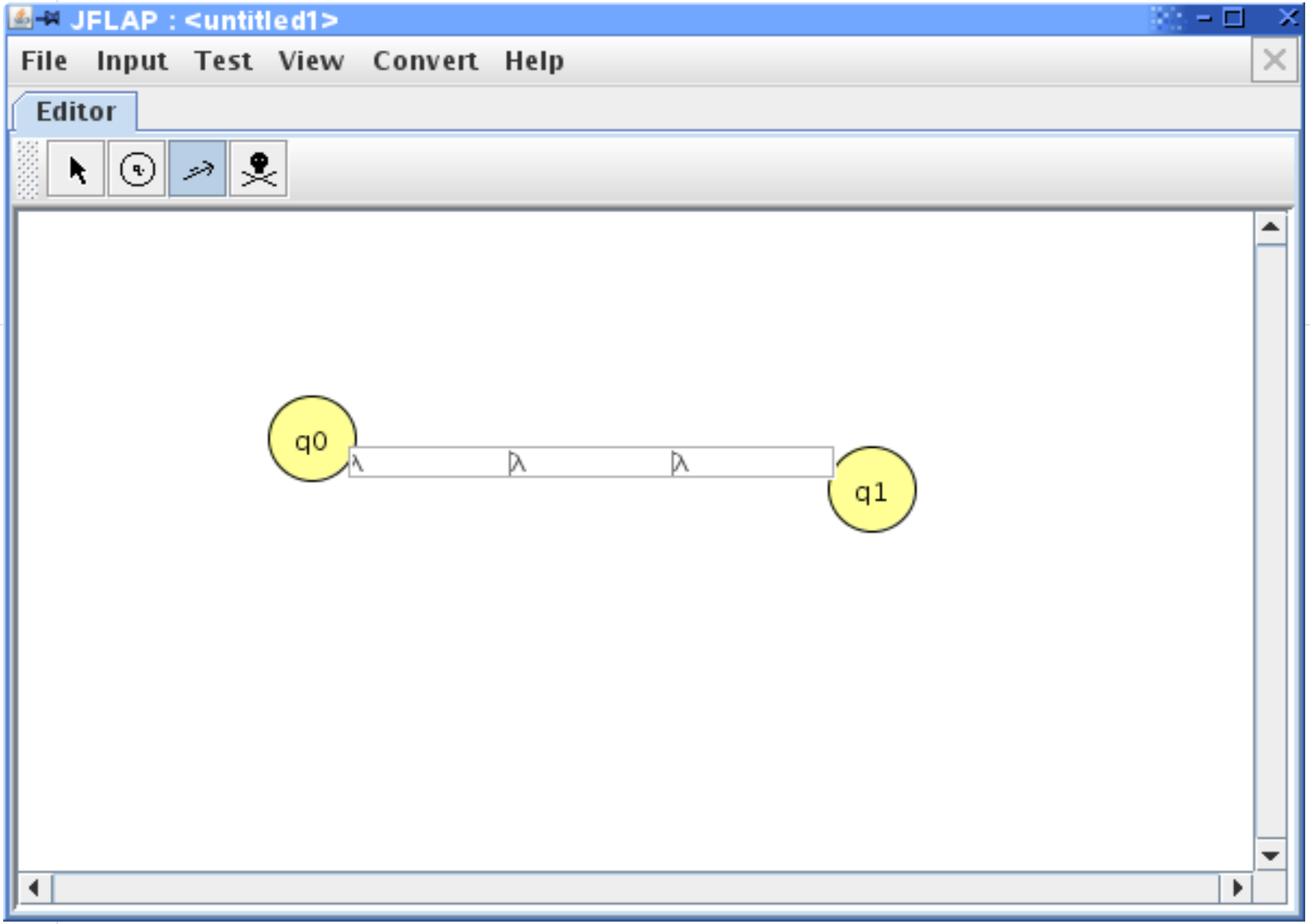
a = ε, read no input
b = ε, don't pop data from stack
c = ε, don't push data onto stack

Practice

- $\{ ww^R \mid w \in \{0, 1\}^* \}$

  hint: push symbols onto the stack, at each point guess that the middle of the string

  has been reached and begin popping from stack


JFLAP : <untitled1>
File
Input
Test
View
Convert
Help
Editor
q0
λ
λ
λ
q1