CSCE 355
3/11/2024

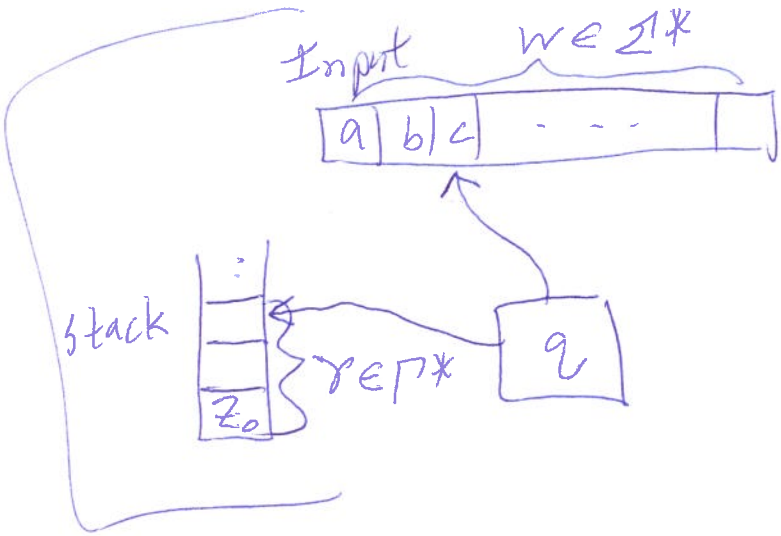
Before: <u>CFGs</u> for <u>CFLs</u>

context-free grammars — context-free language

①

Push-down automata (PDAs) — automata equipped with stacks.

<u>Def:</u> A push-down automaton (PDA) is a tuple $\langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ such that...



... Q is a finite set (the <u>state set</u>)

Σ is an alphabet (the <u>input alphabet</u>)

Γ is an alphabet (the <u>stack alphabet</u>)

$q_0 \in Q$ (the <u>start state</u>)

$Z_0 \in \Gamma$ (the stack bottom marker)

$F \subseteq Q$ (the set of <u>accepting states</u>)

$$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \to 2^{Q \times \Gamma^*}$$

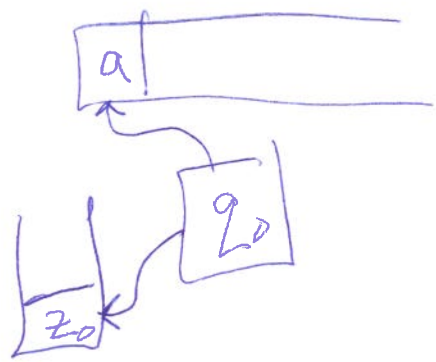
current state

next input symbol to ~~read~~ consume (or ε)

top of the stack currently

On input $w \in \Sigma^*$ ——

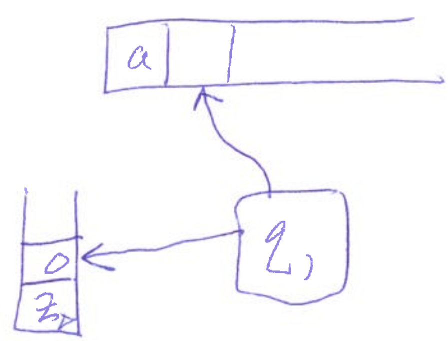
- initially, next symbol to consume is the the 1st symbol of w
- start state
- stack has Z_0 & nothing else



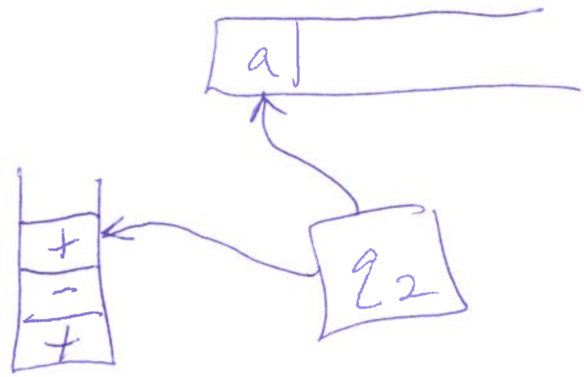
Example: suppose a is being scanned and

$$\delta(q_0, a, Z_0) \text{ contains } (q_1, 0Z_0)$$

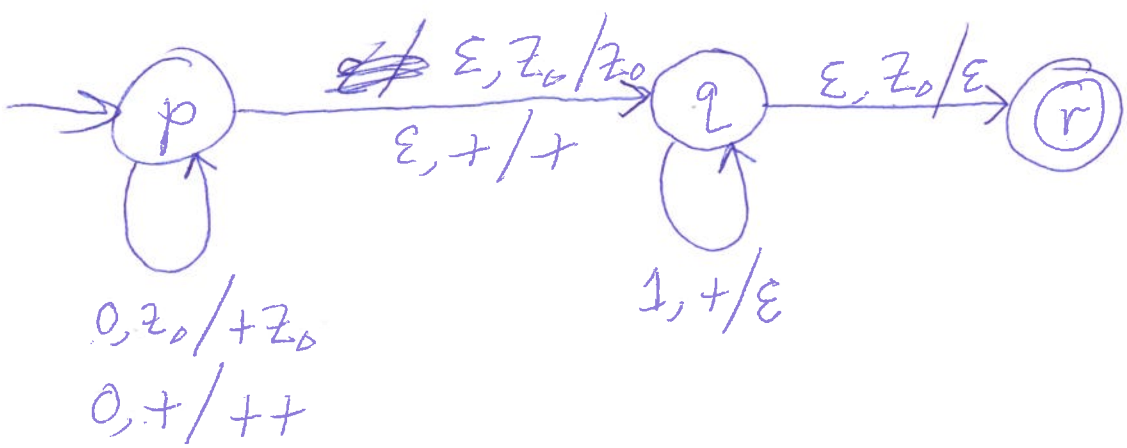
then in the next step, the PDA could be in this configuration:



if $\delta(q_0, \varepsilon, Z_0)$ contains $(q_2, +-+)$
then next configuration could be



Ex: PDA that recognizes $\{0^n 1^n : n \geq 0\}$:
Transition diagram:



Sample inputs: 0011 accepted
001 rejected ~~can't get to r~~ stack
00111 rejected (can get to r, but not by reading the entire input)
10 rejected (same reason as 00111)

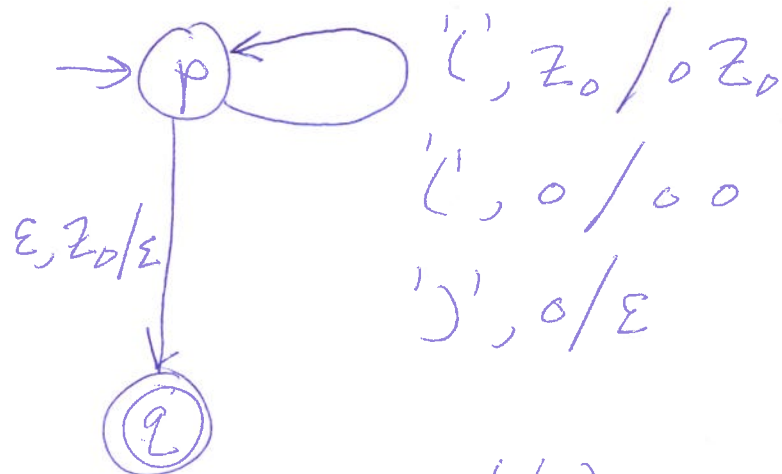
[you can always get to r]

Ex: PDA that recognizes the language of well-balanced strings of parentheses

$$\Sigma = \{ '(', ')' \}$$

$$\Gamma = \{ Z_0, o \}$$

(o = "opening paren")



(( )

$$\langle \{p, q\}, \{ '(', ')' \}, \{Z_0, o\}, \delta, p, Z_0, \{q\} \rangle$$

where

$$\delta(p, '(', Z_0) = \{ (p, o Z_0) \}$$

$$\delta(p, '(', o) = \{ (p, o o) \}$$

$$\delta(p, ')', o) = \{ (p, \varepsilon) \}$$

$$\delta(p, \varepsilon, Z_0) = \{ (q, \varepsilon) \}$$

[All other $\delta(\ldots) = \emptyset$]

Def: Fix a PDA $P = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$

An *instantaneous description* (ID) of P on input $w \in \Sigma^*$ is a triple

$$(q, x, \gamma)$$

where $q \in Q$ (the current state)

and x is a suffix of w $(x \in \Sigma^*)$
(the portion of the input yet to be consumed)

and $\gamma \in \Gamma^*$ (the current contents of the stack (top-to-bottom))

(q, x, γ) is a "snapshot" of a computation path of P on input w with enough information to determine the possibilities of the future behavior of P on input w.

This is also called a *configuration* of P on input w.

The *initial* ~~confn~~ *configuration* of P on input w is (q_0, w, Z_0)

An *accepting configuration* is any ID of the form (q, ε, γ) where $q \in F$ & $\gamma \in \Gamma^*$ is arbitrary.

Def: P, w are as above. We define the *successor relation* on the set of IDs of P as follows: For any $q, r \in Q$ any $a \in \Sigma \cup \{\varepsilon\}$, any $X \in \Gamma$ and any $x \in \Sigma^*$ and any $\beta, \gamma \in \Gamma^*$, say that

$$(q, ax, X\gamma) \vdash (r, x, \beta\gamma)$$

just when $(r, \beta) \in \delta(q, a, X)$

"$\vdash$" means going from left-hand ID to right-hand ID is possible in a single step.