CSCE 355
3/13/2023

CFLs, CFGs, PDAs

<u>Def</u>: A <u>pushdown automaton</u> (PDA) is a tuple $\langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ where

Q is finite set (elements are <u>states</u>)

Σ and Γ are alphabets

Σ is the <u>input alphabet</u>

Γ is the <u>stack alphabet</u>

$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \underline{\Gamma} \rightarrow \left(2^{(Q \times \Gamma^*)}\right) \cap \left(\text{finite sets}\right)$

such that $\delta(q, a, t)$ is finite for all

$q \in Q$, $a \in \Sigma \cup \{\varepsilon\}$, $t \in \Gamma$

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

$Z_0 \in \Gamma$ (the <u>bottom stack marker</u>)

$F \subseteq Q$ (the accepting states)

Ex: $L = \{0^n 1^n : n \geq 0\}$

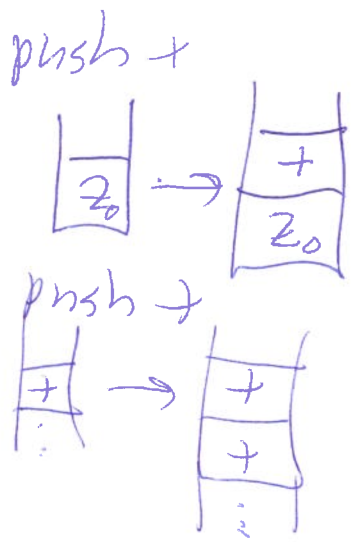
$Q = \{p, q, r\}$ $\qquad$ $q_0 = p$

$\Sigma = \{0, 1\}$ $\qquad$ $F = \{r\}$

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

$\delta(p, 0, Z_0) = \{(p, +Z_0)\}$

$\delta(p, 0, +) = \{(p, ++)\}$

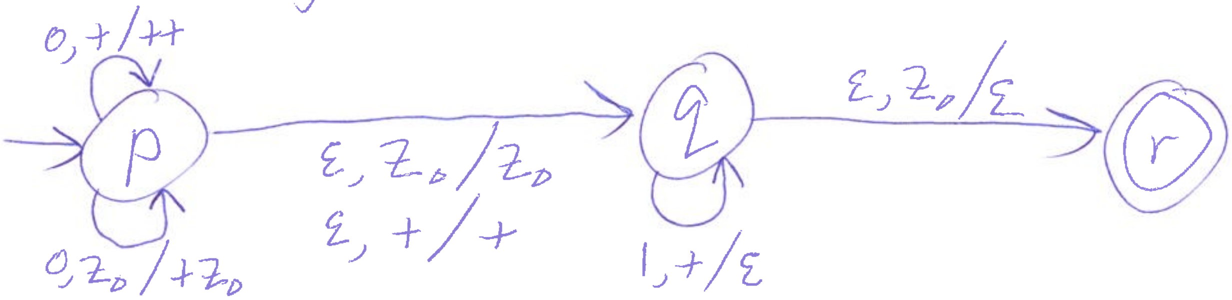


$\left.\begin{array}{l} \delta(p, \varepsilon, Z_0) = \{(q, Z_0)\} \\ \delta(p, \varepsilon, +) = \{(q, +)\} \end{array}\right\}$ stack is unchanged

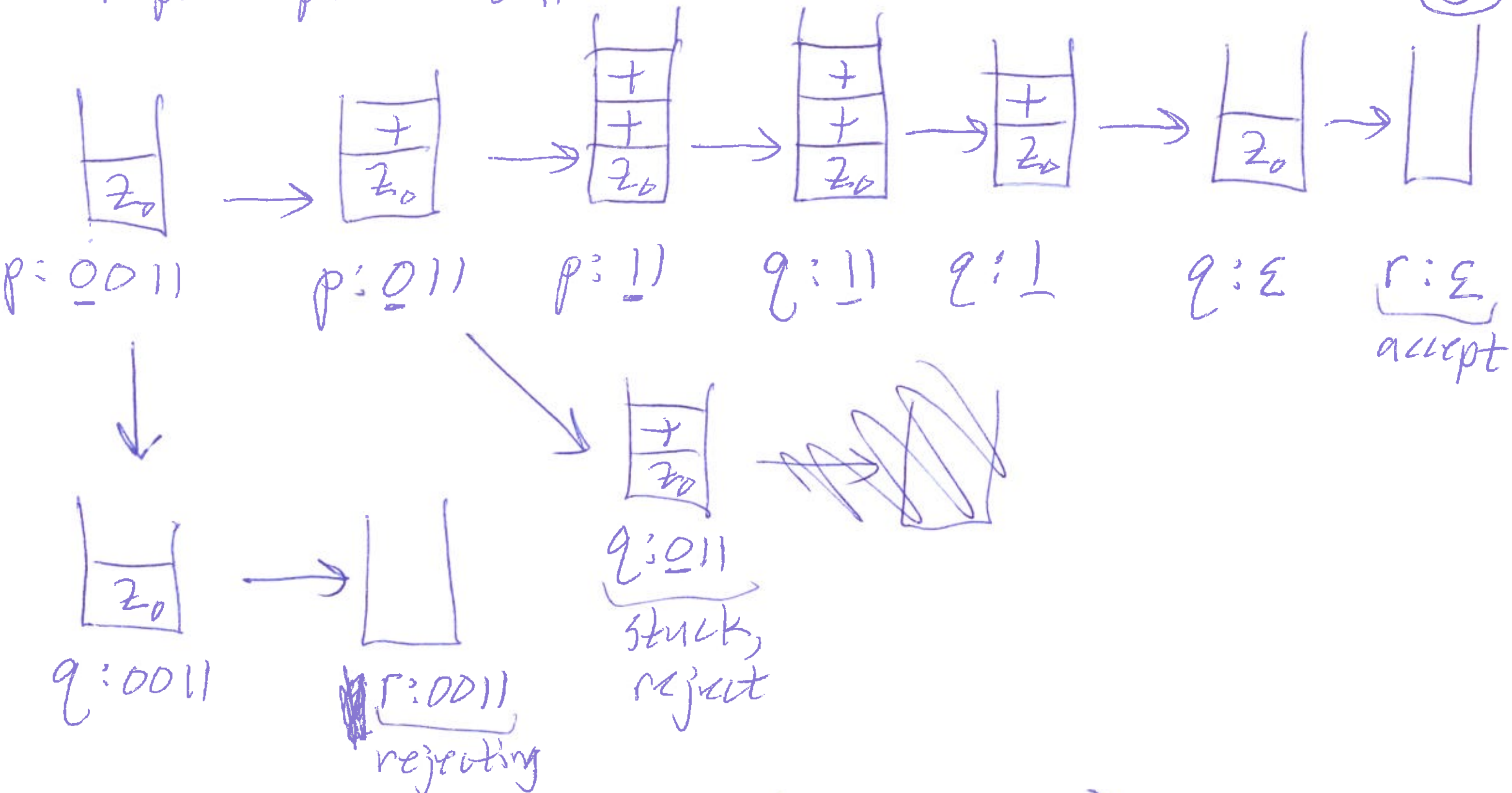
$\delta(q, 1, +) = \{(q, \varepsilon)\}$ — pop

$\delta(q, \varepsilon, Z_0) = \{(r, \varepsilon)\}$ — pop

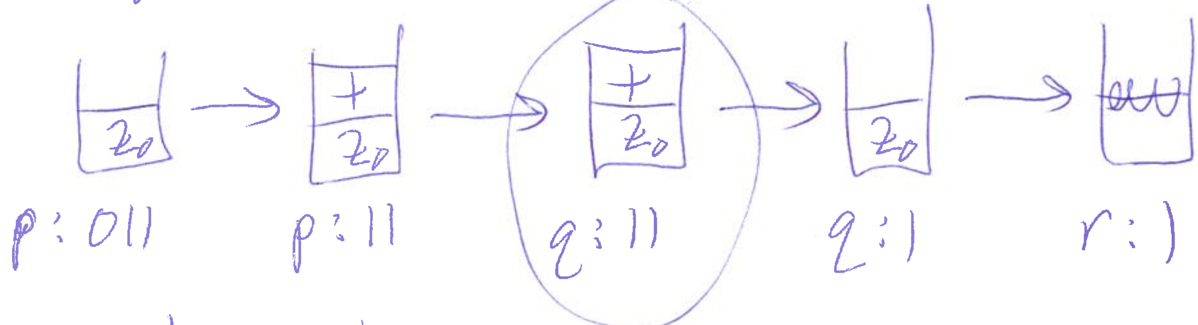
Transition diagram:



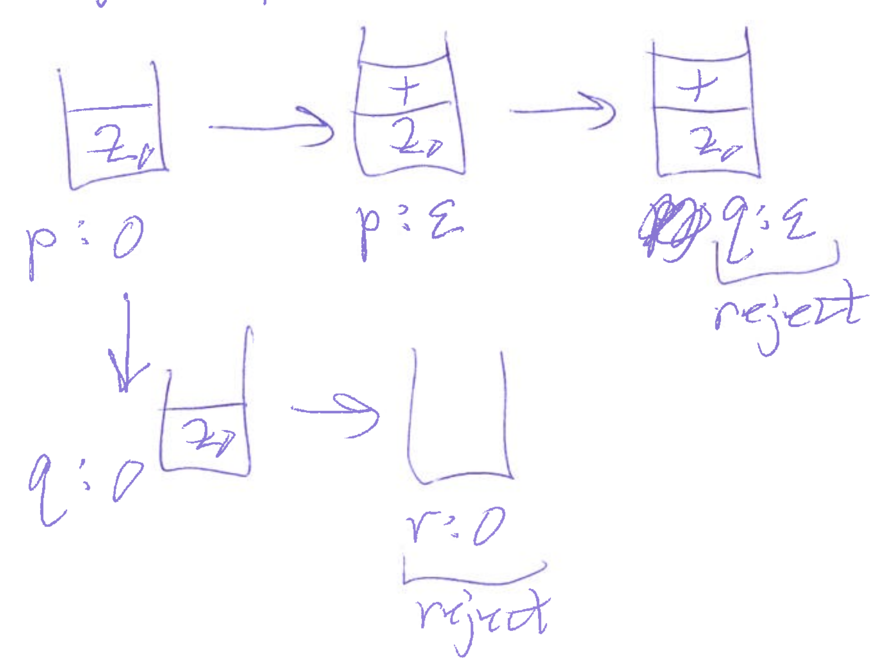
Sample input: 0011



Sample input: 011



Sample input: 0



Formally:

<u>Def</u>: Let $P = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ be a PDA. An <u>ID</u> (Instantaneous Description) of P (a <u>configuration</u> of P) is a triple

$$(q, w, \gamma)$$

where $q \in Q$ ("P is currently in state q")

$w \in \Sigma^*$ ("w is the remaining (unconsumed) portion of the input" — always a suffix of the input)

$\gamma \in \Gamma^*$ (P's stack contents is currently γ read left-to-right means top to bottom)

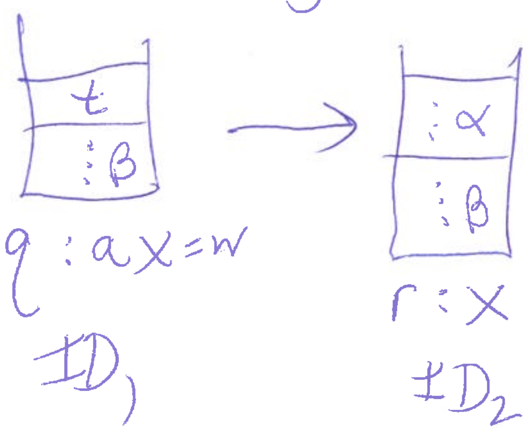
Let $ID_1 = (q, w, \gamma)$ be any ID of P. ~~Sup Let~~ Suppose $w = ax$ for some $a \in \Sigma \cup \{\varepsilon\}$ and $\gamma = t\beta$ for some $t \in \Gamma$ and $\beta \in \Gamma^*$

If $\delta(q, a, t)$ contains some pair

(r, α) for some $r \in Q$ and $\alpha \in \Gamma^*$

then ~~$ID_1 \vdash (q$~~ $ID_2 := (r, x, \alpha\beta)$

is a legal immediate <u>successor</u> of ID_1



Notation:

$$ID_1 \vdash ID_2$$

ID_2 is a successor of ID_1

<u>Ex:</u>

$(p, 0011, Z_0) \vdash (p, 011, +Z_0) \vdash (p, 11, ++Z_0)$

$\vdash (q, 11, ++Z_0) \vdash (q, 1, +Z_0)$

$\vdash (q, \varepsilon, Z_0) \vdash (r, \varepsilon, \varepsilon)$

<u>Def:</u> P a PDA as above. $w \in \Sigma^*$

The <u>initial ID</u> of P on <u>input w</u>

is (q_0, w, Z_0).

<u>Def</u>: P as above. A <u>computation path of P on input $w \in \Sigma^*$</u> is a sequence of IDs of P

$$ID_0 \vdash ID_1 \vdash \cdots \vdash ID_n \qquad (n \geq 0)$$

where ID_0 is the initial ID of P on input w and $ID_i \vdash ID_{i+1}$ for all $0 \leq i < n$.

Say that the path <u>ends</u> in ID_n.

<u>Def</u>: P, w as above.

<u>P accepts w via accepting state</u> if there exists a comp path of P on input w ending in an ID of the form (r, ε, γ) for some $r \in F$ ($\gamma \in \Gamma^*$ could be anything)

<u>P accepts w via empty stack</u> if there is a comp path that ends in an ID of the form, $(r, \varepsilon, \varepsilon)$ (r could be any state)

CSCE 355
3/15/2023

Definition: Let P be a PDA, with input alphabet Σ (1)

~~Pac~~ $L(P) = \{w \in \Sigma^* : P \text{ accepts } w \text{ via accepting state}\}$

$N(P) = \{w \in \Sigma^* : P \text{ accepts } w \text{ via empty stack}\}$

Possible that $L(P) \neq N(P)$ (in general)

Thm: (1) For every PDA P there exists a PDA P' such that $N(P') = L(P)$.

(2) For every PDA P there exists a PDA P' such that $L(P') = N(P)$.

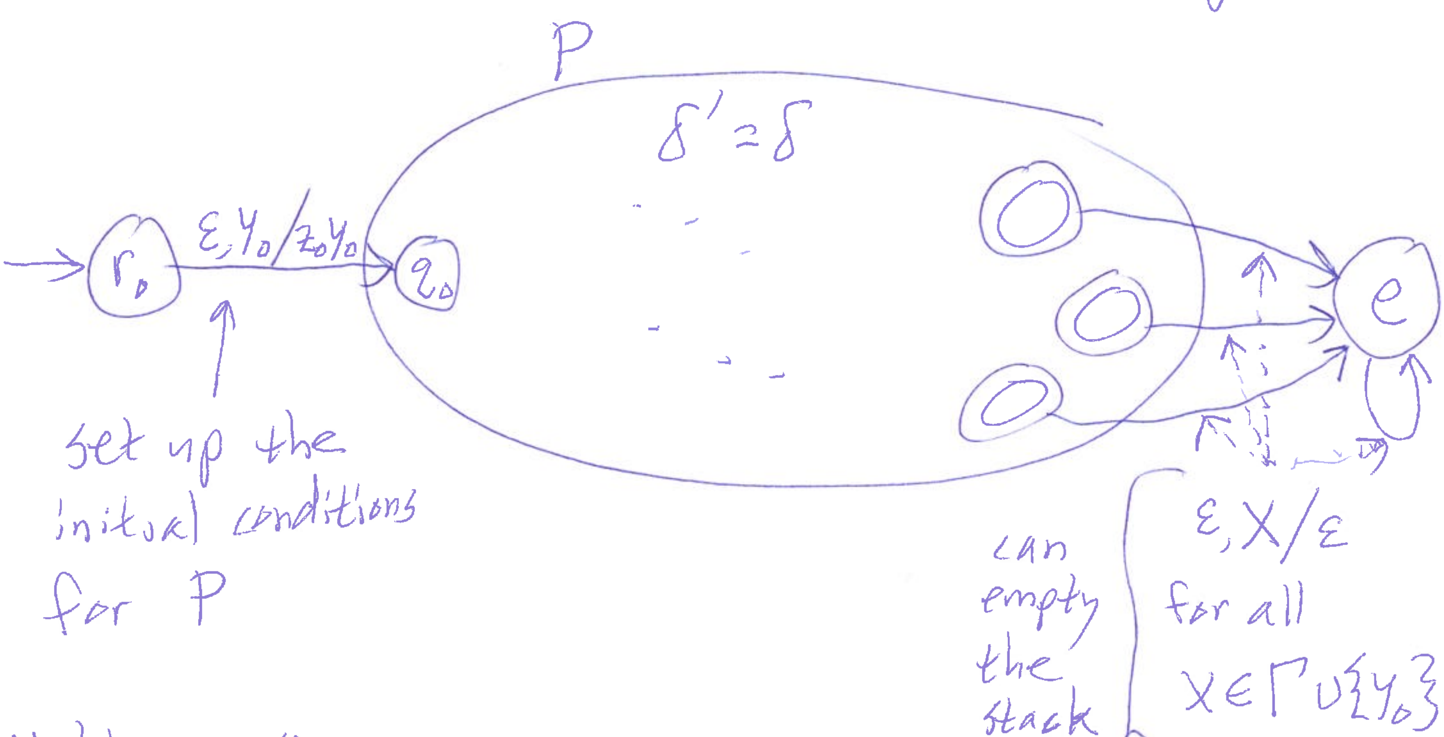
Corollary: $\{L(P) : P \text{ is a PDA}\} = \{N(P) : P \text{ is a PDA}\}$.

[later: = CFLs]

Proof of (1): Let $P = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ be any PDA. Design P' as follows:

$$P' := \langle Q \cup \{r_0, e\}, \Sigma, \Gamma \cup \{Y_0\}, \delta', r_0, Y_0, \emptyset \rangle$$

where $r_0 \neq e$, $r_0, e \notin Q$, $Y_0 \notin \Gamma$, and δ' is given by the following transition diagram:



Notice: If P accepts a string w via accepting state, then P' accepts via empty stack.

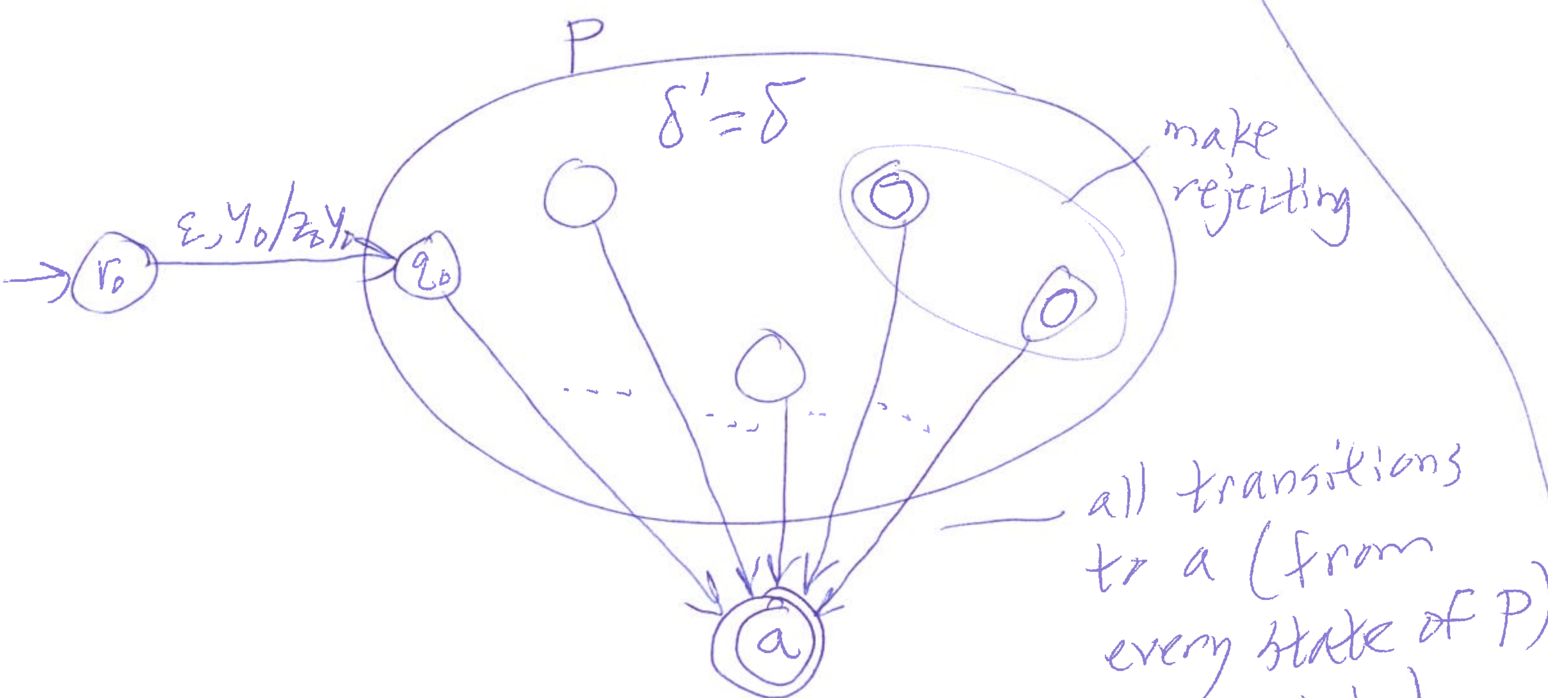
Also need the converse: If P' accepts via empty stack, then P accepts via accepting state.

True, but having a new bottom stack marker for P' is essential. (If P rejects, then P' rejects.)

$\therefore N(P') = L(P)$ (via hand-wave) // proof of (1.)

Proof of (2.) Let $P = \langle Q, \Sigma, \Gamma, \delta, q_0, Z_0, F \rangle$ be any PDA. We construct P' so that $L(P') = N(P)$. $P' = \langle Q \cup \{r_0, a\}, \Sigma, \Gamma \cup \{Y_0\}, \delta', r_0, Y_0, \{a\} \rangle$

and δ' is as follows:



Note: If P accepts via empty stack, then Y_0 will be exposed, allowing a transition to state a, and conversely — this is the only way to get to state a, but must make all accepting states of P rejecting in P.

$\therefore L(P') = N(P)$ (by hand-wave) //proof of (2) $\blacksquare$

Next up: Goal $\quad$ CFG $\Longleftrightarrow$ PDA

[Shows that CFLs are characterized by PDAs]

Today: CFG $\Longrightarrow$ PDA (with 1 state!)

Theorem: Let G be a CFG. There exists a PDA P such that $N(P) = L(G)$.

Proof: By explicit construction. Let

$$G := \langle V, \Sigma, S, P \rangle.$$

Then

$$P := \langle \{q\}, \Sigma, \underbrace{\Sigma \cup V}_{\substack{\text{all grammar}\\ \text{symbols}}}, \delta, q, S, \emptyset \rangle$$

where δ is as follows: For every $a \in \Sigma$

$\delta(q, a, a) = \{(q, \varepsilon)\}$ "matching a"

and for every production $A \to \gamma$ of G, $\delta(q, \varepsilon, A)$ contains (q, γ)

In otherwords, $\forall A \in V$,

$$\delta(q, \varepsilon, A) = \{(q, \gamma) : A \to \gamma \text{ is a production of } G\}$$

"expanding A"

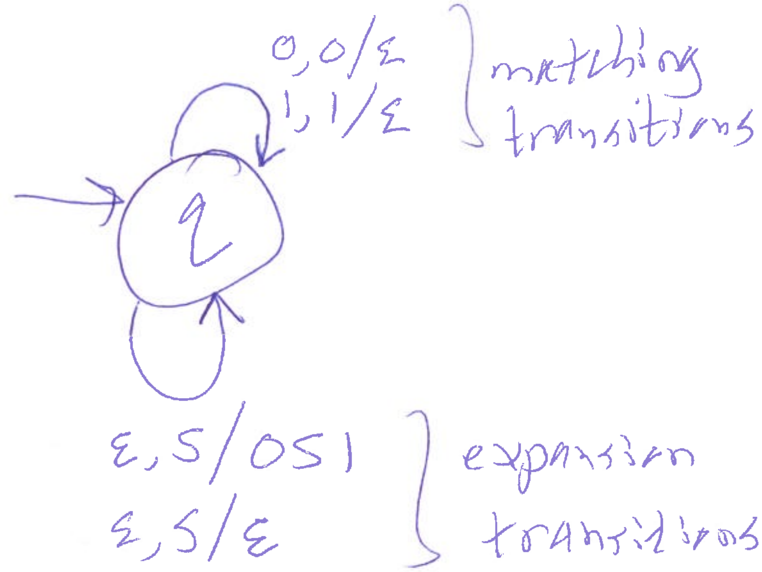
~~All other~~ No other allowed δ-transitions.

Ex: $L = \{0^n 1^n : n \geq 0\}$

$$G = \begin{array}{l} S \to 0S1 \\ S \to \varepsilon \end{array}$$

$P = \langle \{q\}, \{0, 1\}, \{0, 1, S\}, \delta, q, S, \emptyset \rangle$

and δ is as follows:



Sample input: $0011 \in L$

Derivation: $\underline{S} \Rightarrow 0\underline{S1} \Rightarrow \underline{00S11} \Rightarrow 0011$

Accepting path of P:

$(q, 0011, S) \overset{\text{expand}}{\vdash} (q, \underline{0}011, \underline{0}S1) \overset{\text{match } 0}{\vdash} (q, 011, S1)$

$\overset{\text{exp}}{\vdash} (q, \underline{0}11, \underline{0}S11) \overset{\text{match } 0}{\vdash} (q, 11, S11)$

$\overset{\text{exp}}{\vdash} (q, 11, 11) \overset{\text{match } 1}{\vdash} (q, 1, 1) \overset{\text{match } 1}{\vdash} \underbrace{(q, \varepsilon, \varepsilon)}_{\text{accept}}$

Ex: Input 001

$(q, 001, \underline{S}) \overset{\text{exp}}{\vdash} (q, \underline{0}01, \underline{0}S1) \overset{\text{match } 0}{\vdash} (q, 01, \underline{S}1)$

$\overset{\text{exp}}{\vdash} (q, \underline{0}1, \underline{0}S11) \overset{\text{match } 0}{\vdash} (q, 1, S11)$

$\overset{\text{exp}}{\vdash} (q, \underline{1}, \underline{1}1) \overset{\text{match } 1}{\vdash} (q, \varepsilon, 1)$ stuck, reject