Instantaneous Description (ID)

(q, w, γ): state, remaining input, contents of the stack

If $(p, \alpha) \in \delta(q, a, X)$, then $(q, aw, X\beta) \vdash (p, w, \alpha\beta)$

Thm: If $(q, x, \alpha) \vdash^* (p, y, \beta)$, then $(q, xw, \alpha\gamma) \vdash^* (p, yw, \beta\gamma)$

Acceptance by final state: $L(P) = \{w | (q_0, w, Z_0) \vdash^* (q, \varepsilon, \alpha), q \in F\}$

Acceptance by empty stack: $L(P) = \{w | (q_0, w, Z_0) \vdash^* (q, \varepsilon, \varepsilon)\}$

Thm: L is accepted by PDA by final state iff it is accepted by PDA by empty stack.

Proof: When Z_0 is popped, enter an accepting state. For the other direction, when an accepting state is entered, pop all the stack.

Equivalence of CFGs and PDAs

From Grammar to PDA:

A left sentential form is $\underbrace{x}_{\in T^*}\overbrace{A\alpha}^{\text{tail}}$

The tail appears on the stack, and x is the prefix of the input that has been consumed so far.

Total input is $w = xy$, and hopefully $A\alpha \stackrel{*}{\Rightarrow} y$.

Suppose PDA is in $(q, y, A\alpha)$. It guesses $A \longrightarrow \beta$, and enters $(q, y, \beta\gamma)$.

The initial segment of β, if it has any terminal symbols, they are compared against the input and removed, until the first variable of β is exposed on top of the stack.

Accept by empty stack.

Ex. Consider $P \longrightarrow \varepsilon|0|1|0P0|1P1$

The PDA has transitions:

$\delta(q_0, \varepsilon, Z_0) = \{(q, PZ_0)\}$

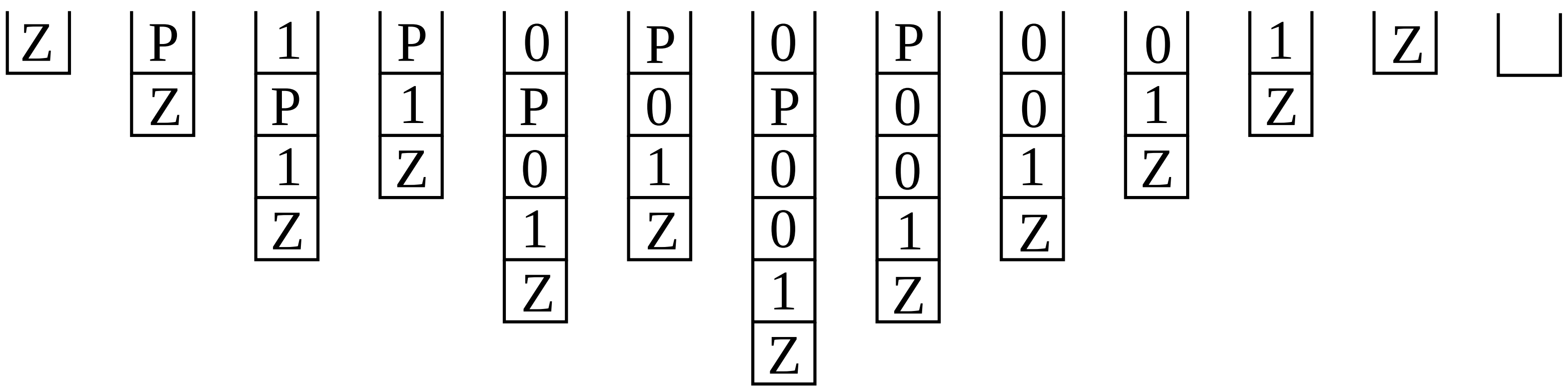
$\delta(q, \varepsilon, P) = \{(q, 0P0), (q, 0), (q, \varepsilon), (q, 1P1), (q, 1)\}$

$\delta(q, 0, 0) = \delta(q, 1, 1) = \{(q, \varepsilon)\}$

$\delta(q, 0, 1) = \delta(q, 1, 0) = \emptyset$

$\delta(q, \varepsilon, Z_0) = (q, \varepsilon)$

Consider: $P \Rightarrow 1P1 \Rightarrow 10P01 \Rightarrow 100P001 \Rightarrow 100001$



From PDA to grammar:

Idea: "net popping" of one symbol of the stack, while consuming some input.

Variables: $A_{[pXq]}$, for $p, q \in Q$, $X \in \Gamma$.

$A_{[pXq]} \stackrel{*}{\Rightarrow} w$ iff w takes PDA from state p to state q, and pops X off the stack.

Productions: for all p, $S \longrightarrow A_{[q_0 Z_0 p]}$, and whenever we have:

$$(r, Y_1 Y_2 \ldots Y_k) \in \delta(q, a, X)$$

$A_{[qXr_k]} \longrightarrow aA_{[rY_1r_1]}A_{[r_1Y_2r_2]} \ldots A_{[r_{k-1}Y_kr_k]}$
where $a \in \Sigma \cup \{\varepsilon\}$, $r_1, r_2, \ldots, r_k \in Q$ are all possible lists of states.

If $(r, \varepsilon) \in \delta(q, a, X)$, then we have $A_{[qXr]} \longrightarrow a$.

Claim: $A_{[qXp]} \stackrel{*}{\Rightarrow} w \iff (q, w, X) \vdash^* (p, \varepsilon, \varepsilon)$.

A PDA is deterministic if $|\delta(q, a, X)| \leq 1$, and the second condition is that if for some $a \in \Sigma$ $|\delta(q, a, X)| = 1$, then $|\delta(q, \varepsilon, X)| = 0$.

Thm: If L is regular, then $L = L(P)$ for some deterministic PDA P.

Proof: ignore the stack.

DPDAs that accept by final state are **not** equivalent to DPDAs that accept by empty stack.

L has the prefix property if there exists a pair (x, y), $x, y \in L$, such that $y = xz$ for some z.

Ex. $\{0\}^*$ has the prefix property.

Thm: L is accepted by a DPDA by empty stack $\iff$ L is accepted by a DPDA by final state **and** L does not have the prefix property.

Thm: If L is accepted by a DPDA, then L is unambiguous.

Eliminating useless symbols from CFG:

$X \in V \cup T$ is **useful** if there exists a derivation such that $S \overset{*}{\Rightarrow} \alpha X \beta \overset{*}{\Rightarrow} w \in T^*$

X is **generating** if $X \overset{*}{\Rightarrow} w \in T^*$

X is **reachable** if there exists a derivation $S \overset{*}{\Rightarrow} \alpha X \beta$

A symbol is useful if it is generating and reachable.

Generating symbols: Every symbol in T is generating, and if $A \longrightarrow \alpha$ is a production, and every symbol in α is generating (or $\alpha = \varepsilon$) then A is also generating.

Reachable symbols: S is reachable, and if A is reachable, and $A \longrightarrow \alpha$ is a production, then every symbol in α is reachable.

If L has a CFG, then $L - \{\varepsilon\}$ has a CFG without productions of the form $A \longrightarrow \varepsilon$

A variable is **nullable** if $A \overset{*}{\Rightarrow} \varepsilon$

To compute nullable variables: if $A \longrightarrow \varepsilon$ is a production, then A is nullable, if $B \longrightarrow C_1 C_2 \dots C_k$ is a production and all the C_i's are nullable, then so is B.

Once we have all the nullable variables, we eliminate ε-productions as follows: eliminate all $A \longrightarrow \varepsilon$.

If $A \longrightarrow X_1 X_2 \dots X_k$ is a production, and $m \leq k$ of the X_i's are nullable, then add the 2^m versions of the rule the the nullable variables present/absent (if $m = k$, do not add the case where they are *all* absent).

Eliminating unit productions: $A \longrightarrow B$
If $A \stackrel{*}{\Rightarrow} B$, then (A, B) is a unit pair.

Find all unit pairs: (A, A) is a unit pair, and if (A, B) is a unit pair, and $B \longrightarrow C$ is a production, then (A, C) is a unit pair.

To eliminate unit productions: compute all unit pairs, and if (A, B) is a unit pair and $B \longrightarrow \alpha$ is a non-unit production, add the production $A \longrightarrow \alpha$. Throw out all the unit productions.

A CFG is in **Chomsky Normal Form** if all the rules are of the form $A \longrightarrow BC$ and $A \longrightarrow a$.

Thm: Every CFL without ε has a CFG in CNF.

Proof: Eliminate ε-productions, unit productions, useless symbols. Arrange all bodies of length ≥ 2 to consist of only variables (by introducing new variables), and finally break bodies of length ≥ 3 into a cascade of productions, each with a body of length exactly 2.

Pumping Lemma for CFLs: There exists a p so that any s, $|s| \geq p$, can be written as $s = uvxyz$, and:

1. $uv^i xy^i z$ is in the language, for all $i \geq 0$,
2. $|vy| > 0$,
3. $|vxy| \leq p$

Proof: