CONTEXT-FREE GRAMMARS AND PUSH-DOWN AUTOMATA

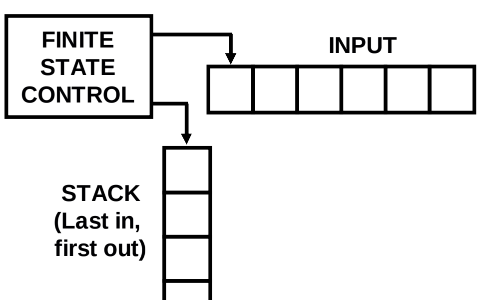
TUESDAY SEP 13

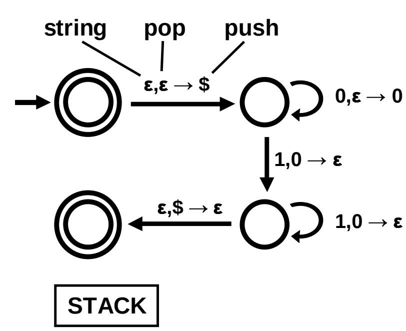
$\Sigma = \{0, 1\}$, $L = \{ 0^n1^n \mid n \geq 0 \}$

$\Sigma = \{a, b, c, \ldots, z\}$, $L = \{ w \mid w = w^R \}$

$\Sigma = \{ (,) \}$, $L = \{$ balanced strings of parens $\}$

PUSHDOWN AUTOMATA





Definition: A (non-deterministic) PDA is a tuple $P = (Q, \Sigma, \Gamma, \delta, q_0, F)$, where:

Q is a finite set of states

Σ is the input alphabet

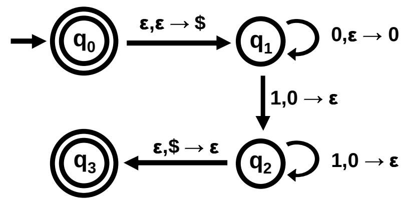
Γ is the stack alphabet

$\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow 2^{Q \times \Gamma_\varepsilon}$

$q_0 \in Q$ is the start state

$F \subseteq Q$ is the set of accept states

2^Q is the set of subsets of Q and $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$

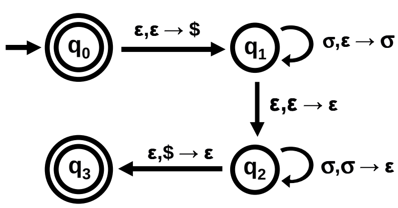


$Q = \{q_0, q_1, q_2, q_3\}$ $\quad \Sigma = \{0,1\}$ $\quad \Gamma = \{\$,0,1\}$

$\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow 2^{Q \times \Gamma_\varepsilon}$

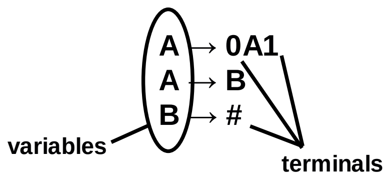
$\delta(q_1,1,0) = \{ (q_2,\varepsilon) \}$ $\quad \delta(q_2,1,1) = \emptyset$

$\Sigma = \{a, b, c, \ldots, z\}$



Build a PDA to recognize
$L = \{ a^i b^j c^k \mid i, j, k \geq 0 \text{ and } (i = j \text{ or } i = k) \}$

CONTEXT-FREE GRAMMARS



$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 00B11 \Rightarrow 00\#11$

SNOOP'S GRAMMAR

<PHRASE> → <FILLER><PHRASE>

<PHRASE> → <START WORD><END WORD>DUDE

<FILLER> → LIKE

<FILLER> → UMM

<START WORD> → FO

<START WORD> → FA

<END WORD> → SHO

<END WORD> → SHAZZY

<END WORD> → SHEEZY

CONTEXT-FREE GRAMMARS

A context-free grammar (CFG) is a tuple G = (V, Σ, R, S), where:

V is a finite set of variables

Σ is a finite set of terminals (disjoint from V)

R is set of production rules of the form A → W, where A ∈ V and W ∈ (V∪Σ)*

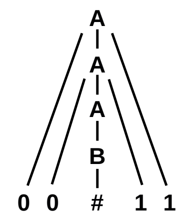
S ∈ V is the start variable

G = { {S}, {0,1}, R, S }   R = { S → 0S1, S → ε }

WRITE A CFG FOR PALINDROMES

WRITE A CFG FOR THE EMPTY SET

PARSE TREES



A ⇒ 0A1 ⇒ 00A11 ⇒ 00B11 ⇒ 00#11

<EXPR> → <EXPR> + <EXPR>

<EXPR> → <EXPR> x <EXPR>

<EXPR> → (<EXPR>)

<EXPR> → a

Build a parse tree for a + a x a

Definition: a string is derived ambiguously in a context-free grammar if it has two or more different parse trees

Definition: a grammar is ambiguous if it generates some string ambiguously

THE PUMPING LEMMA

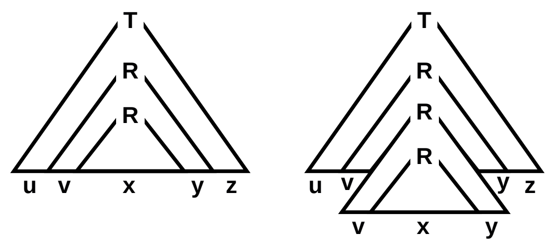
Let L be a context-free language

Then there exists P such that
if $w \in L$ and $|w| \geq P$

then w = uvxyz, where:

1. $|vy| > 0$
2. $|vxy| \leq P$
3. $uv^ixy^iz \in L$ for any $i \geq 0$

Idea: If w is long enough, then any parse tree for w must have a path that contains a variable more than once



Formal Proof:

Let b be the maximum number of symbols on the right-hand side of a rule

If the height of a parse tree is h, the length of the string generated is at most:

Let |V| be the number of variables in G

Define $P = b^{|V|+2}$

Let w be a string of length at least P

Let T be the parse tree for w with the smallest number of nodes.

T must have height at least |V|+2

Let T be the parse tree for w with the smallest number of nodes. T must have height at least |V|+2

The longest path in T must have $\geq$ |V|+1 variables

Select R to be the variable that repeats among the lowest |V|+1 variables

1. $|vy| > 0$
2. $|vxy| \leq P$

A Language is generated by a CFG
$\Leftrightarrow$
It is recognized by a PDA

WWW.FLAC.WS

Read chapter 2 of the book for next time