Context-Free Languages.

A Language L is context-free if there is a context-free grammar G with $L = L(G)$. $L = \{a^n b^n : n \geq 0\}$ is a context-free language since context-free grammar G : $S \to aSb \mid \lambda$ generates $L(G) = L$.

Grammar $G = (V, T, S, P)$

V : set of variables.

T : Set of terminal symbols.

S : Start variable.

P : set of productions : $A \to B$; A: variable; B: string of variables and terminals.

Production : variables $\to$ sequence of terminals or sequence of variables.

e.g: $S \to aSb$, $S \to \lambda$ $\iff$ $S \to aSb \mid \lambda$

Derivation steps : $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow \cdots$

In general we write $w_1 \overset{*}{\Rightarrow} w_n$ if $w_1 \Rightarrow w_2 \Rightarrow \cdots \Rightarrow w_n$

And $w \overset{*}{\Rightarrow} w$

Notice the difference between production and Derivation steps.

Examples: $L = \{ww^R : w \in \{a,b\}^*\}$ is a context-free language since there is a context-free grammar G : $S \to aSa \mid bSb \mid \lambda$ generates $L(G) = L$.

$L = \{w : n_a(w) = n_b(w)$, and $n_a(v) \geq n_b(v)$ in any prefix $v\}$ is a context-free language G : $S \to aSb \mid SS \mid \lambda$ such that $L(G) = L$.

Derivation Order:

$S \Rightarrow AB \Rightarrow \dots$

Leftmost derivation: at each step, we substitute the leftmost variable.

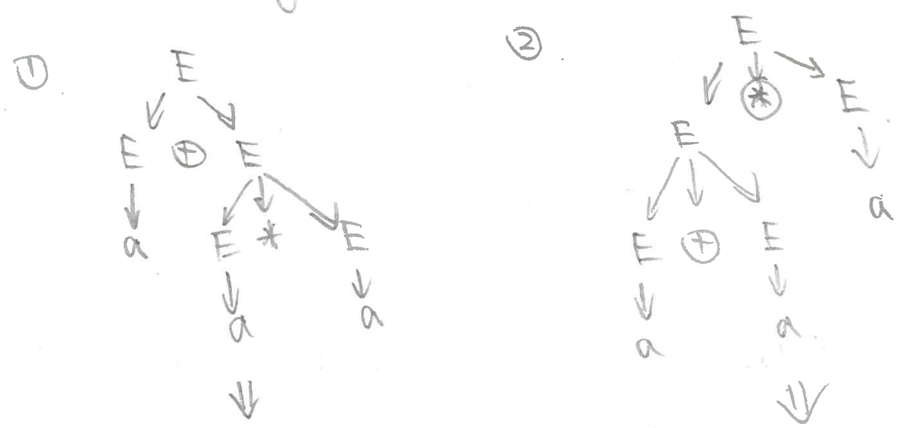
Rightmost derivation: at each step, we substitute the rightmost variable.

Ambiguous Grammar: a context-free grammar G is ambiguous if there is a string $w \in L(G)$ which has two different derivation trees or two leftmost derivations.

(Two different derivation trees give two different leftmost derivations, vice-vers

Example, given $G: E \to E+E \mid E*E \mid (E) \mid a$.

① string: $a+a*a$ has two derivation trees.



$a+a*a$ $\qquad$ $(a+a)*a$

② $a+a*a$ has two leftmost derivations:

① $E \Rightarrow E+E \Rightarrow a+E*E \Rightarrow a+a*E \Rightarrow a+a*a$

② $E \Rightarrow E*E \Rightarrow E+E*E \Rightarrow a+E*E \Rightarrow a+a*E \Rightarrow a+a*a$

Ambiguity is bad.

$L = \{a^n b^n c^m\} \cup \{a^n b^m c^m\}$, $n, m \geq 0$ $\qquad$ are ambiguous.

$L = \{a^n b^n c^n\}$

Simplifications of Context-Free Grammars

λ-production : $X \to \lambda$

Nullable variable : $A \Rightarrow \cdots \Rightarrow \lambda$,

Unit-Production : $X \to Y$. (a single variable in both sides). not terminal.

If there is a derivation consists of terminals :

$$\underbrace{S}_{\text{start variable}} \Rightarrow \cdots \Rightarrow xAy \Rightarrow \cdots \Rightarrow \underbrace{w}_{\text{All terminals.}} \in L(G).$$

Then variable A is useful.

Otherwise, variable A is useless since $\begin{cases} \text{No reachable from } S \\ \text{or} \\ \text{derivations never terminate.} \end{cases}$

A production $A \to x$ is useless if any of its variables is useless.

Removing Useless variables and productions.

Steps to simplify context-free grammar:

Step 1 : Remove Nullable variables.

Step 2 : Remove unit-Production.

Step 3 : Remove Useless variables.

Chomsky Normal Form.

Each production has form :

$A \to BC$ or $A \to a$ single terminal.

variable variable terminal.

Exactly two variables.

From any context-free grammar (which doesn't produce λ) not in Chomsky Normal Form, we can obtain an equivalent grammar in Chomsky Normal Form.

The procedure:

Step 1: Remove
i). Nullable variable
ii). Unit Productions
iii). Useless variables (Optional).

Step 2: For every symbol a: create a new variable T_a, such that $T_a \to a$.

Then in productions with length at least 2, replace a with T_a.

Step 3: Replace any production $A \to C_1 C_2 \cdots C_n$

with

$$A \to C_1 V_1$$
$$V_1 \to C_2 V_2$$
$$V_2 \to C_3 V_3$$
$$\cdots$$
$$V_{n-2} \to C_{n-1} C_n.$$

Greinbach Normal Form:

All productions has form: $A \to \underset{\substack{\text{terminal} \\ \text{symbol.}}}{a} \underbrace{V_1 V_2 \cdots V_k}_{\text{variables.}}$ $\quad k \geq 0$