Defn: Let $G = (V, T, P, S)$ be a CFG. A symbol $X \in V \cup T$ is said to be

- *generating* if $X \Rightarrow^* w$ for some $w \in T^*$;
- *reachable* if $S \Rightarrow^* \alpha X \beta$ for some $\alpha \beta \in (V \cup T)^*$;
- *useful* if $S \Rightarrow^* \alpha X \beta \Rightarrow^* w$ for some $w \in T^*$, $\alpha \beta \in (V \cup T)^*$; or
- *useless* if it is not useful.

Chomsky Normal Form

Defn: A CFG is said to be in *Chomsky Normal Form (CNF)* if every symbol is useful and every production is of one of the following forms:

- $A \rightarrow BC$, where $A$ and $B$ are variables; or
- $A \rightarrow a$, where $A$ is a variable and $a$ is a terminal.

Theorem 7.16: Let $G$ be a CFG such that $L(G) - \{\epsilon\} \neq \emptyset$. Then we can effectively construct a CNF grammar $G'$ such that $L(G') = L(G) - \{\epsilon\}$. 
Theorem 7.2: Let $G = (V, T, P, S)$ be a CFG such that $L(G) \neq \emptyset$, and let

- $V_g$ be the set of generating variables in $V$;
- $P_g$ be the set of productions in $P$ whose variables all belong to $V_g$;
- $V_u$ and $T_u$ be the sets of reachable variables and terminals, resp., in $G_g = (V_g, T, P_g, S)$;
- and $P_u$ be the set of productions in $P_g$ containing only symbols in $V_u \cup T_u$.

Then $G_u = (V_u, T_u, P_u, S)$ has no useless symbols, and $L(G_u) = L(G)$.

$G_u$ is a CFG:

- Because $L(G) \neq \emptyset$, $S \in V_g$; hence, $G_g$ is a CFG.
- $S \in V_u$, so $G_u$ is a CFG.
$G_u$ has no useless symbols:

- Let $X \in V_u \cup T_u$.
- Because $X$ is reachable in $G_g$, $S \Rightarrow G_g \cdots \Rightarrow \alpha X \beta$.
- Because every symbol in $V_g \cup T_g$ is generating in $G$, $\alpha X \beta \Rightarrow G_g \cdots \Rightarrow w$ for some $w \in T^*$.
- Clearly, $S \Rightarrow G_g \cdots \Rightarrow \alpha X \beta \Rightarrow G_g \cdots \Rightarrow w$.
- Clearly, $S \Rightarrow G_u \alpha X \beta \Rightarrow G_u w$, so $X$ is useful in $G_u$.

$L(G_u) = L(G)$:

- Because $P_u \subseteq P$, $L(G_u) \subseteq L(G)$.
- Let $w \in L(G)$.
- Then $S \Rightarrow G \cdots \Rightarrow w$.
- Clearly, $S \Rightarrow G_g \cdots \Rightarrow w$.
- Clearly, $S \Rightarrow G_u^* w$. 


genvars\((V, T, P, S)\) 
\[V_g \leftarrow \emptyset\]

// Invariant: \(V_g \subseteq V\), each \(A \in V_g\) is generating 
repeat
\[V' \leftarrow V_g\]
for each \(A \rightarrow \alpha \in P, A \notin V'_g\) do
if \(\alpha \in (V'_g \cup T)^*\) then \(V_g \leftarrow V_g \cup \{A\}\)
until \(V'_g = V_g\)
// \(V_g \subseteq V\), each \(A \in V_g\) is generating, and for 
// each \(A \rightarrow \alpha \in P\) s.t. \(\alpha \in (V_g \cup T)^*\), \(A \in V_g\)
return \(V_g\)

Claim: \(\text{genvars}(V, T, P, S)\) always terminates.

Proof sketch:

• If \(V_g\) does not increase in size, the loop terminates.

• \(V_g \subseteq V\).

• because \(V\) is finite, \(V_g\) can increase in size only finitely many times.
Claim: \textit{genvars}(V, T, P, S) returns the set of all generating variables in \((V, T, P, S)\).

Proof sketch:

- Clearly, \(V_g\) contains only generating variables.
- We will show by induction on \(\Rightarrow^*\) that if \(\alpha \Rightarrow^* w \in T^*\), then \(\alpha \in (V_g \cup T)^*\).
- It will follow that if \(A\) is a generating variable, then \(A \in V_g\).

Base: \(\alpha = w\). Then \(\alpha \in T^* \subseteq (V_g \cup T)^*\).

IH: Assume that given \(\alpha \Rightarrow^* w \in T^*\), we have \(\alpha \in (V_g \cup T)^*\).

IS: Suppose \(\beta \Rightarrow \alpha \Rightarrow^* w\).

- Then \(\beta = \gamma_1 A \gamma_3\) and \(\alpha = \gamma_1 \gamma_2 \gamma_3\) where \(A \rightarrow \gamma_2 \in P\).
- Because \(\alpha \in (V_g \cup T)^*\), \(\gamma_2 \in (V_g \cup T)^*\), so \(A \in V_g\).
- Therefore, \(\beta \in (V_g \cup T)^*\).
Claim: There is an algorithm to find all reachable symbols in a given CFG.

Proof sketch:

- Construct a directed graph whose nodes are the symbols of $G$ and such that $(X, Y)$ is an edge if there is a production $X \to \alpha Y \beta$.
- Find all nodes reachable from $S$ using depth-first search.

Corollary: For a CFG $G$ such that $L(G) \neq \emptyset$, we can effectively construct a CFG $G'$ with no useless symbols such that $L(G') = L(G)$. 

11