Def: A CFG \( G = (V, \Sigma, R, S) \) is in Chomsky normal form (ChNF) if every rule is in \( V \to \Sigma \) or \( V \to VV \).

That is, every rule has the form \( A \to c \), \( c \in \Sigma \), or \( A \to BB \) or \( A \to AA \) or \( A \to BC \), \( B, C \in V \).

The rest of the page discusses the last two steps to ChNF, which meaning less.

Theorem: For every CFG \( G_0 \) we can build a CFG \( G_4 \) in ChNF such that \( L(G_4) = L(G_0) \). \( \{ \epsilon \} \).

(If \( \epsilon \in L(G_0) \), the text adds adding a new \( S' \).

\( S' \rightarrow E \) | right-hand sides of rules for \( S_3 \) in \( G_4 \).

Then the grammar \( G_4 \) is in "Relaxed ChNF" and gives \( L(G_4) = L(G_0) \) exactly.

The last two steps are "silly" IMHO:

We will get \( G_2 \) in which each rule is \( A \to c \) or \( A \to \hat{X} \) with \( |\hat{X}| \geq 2 \).

3. If a terminal \( c \) occurs in \( \hat{X} \), add an "alias variable" \( P_c \) and the rule \( P_c \to c \). Then replace every occurrence of \( c \) in \( \hat{X} \) by \( P_c \).

Doing this for all rules makes \( L(G_3) = L(G_2) \), all rules in \( V \cup \Sigma \) or \( V \cup V \).

4. Given a rule \( A \to \hat{X} \) where \( \hat{X} \) consists of 3 or more variables, break \( \hat{X} \) down in steps of 2, like so: \( A \to BCD \) becomes \( A \to BC, \phi \to CD, \phi \to DE \), where \( \phi \) are the top rules for the new variables \( B, C, D \).
**Def:** A variable $A$ is **nullable** if $A \Rightarrow^* \varepsilon$.

The first steps on the way to ChNF are:

1. Identify the subset $\text{NULLABLE}=V$ of nullable vars.
2. Alter the rules so that $\varepsilon$-rules $B \Rightarrow \varepsilon$ are no longer needed. *Text blends these steps.*

**Alg** for (1). Initialize $N = \{ A \in V : A \Rightarrow \varepsilon \text{ is a rule} \}$

```
bool changed = true
while (changed) {
    changed = false;
    foreach $B$ in $V \setminus N$:
        if ($B$ has a rule $B \Rightarrow X$ with $X \in N^*$):
            $N := N \cup \{B\}$
            changed = true;
}

```

output $N$ // = final NULLABLE. There are 2 occurrences of $S$ in the 1st rule, so we need $2^2 = 4$

**Thm:** $G_1$ derives all the nonempty balanced parentheses strings:

- $G_1 = \{ \text{combs: (1) The original rule } S \Rightarrow (S/S) \}
- (2) $S \Rightarrow (S/S)$
- (3) $S \Rightarrow S/S$
- (4) $S \Rightarrow S/S$

Deleting both occurs.

*Step 2 explained inaller rules.*

**Example:** $S \Rightarrow (S/S)S \mid \varepsilon$
Second step: "Bypass unit rules $A \rightarrow B$.

1. Build the graph $H$ of edges $(A, B)$ for all unit rules in $G$.

We want to identify cases where $A \rightarrow B \rightarrow C$.

2. Take $H^*$ to be the transitive closure of $H$ including "Gen-1" $(A, B)$ cases.

3. For each $(A, C) \in H^*$, and all rules $C \rightarrow \hat{X}$, add $A \rightarrow \hat{X}$ as a rule. **Sound.**

4. Delete all unit rules (incl any new ones) comprehensive because added $A \rightarrow \hat{X}$ rules cover all situations that had unit steps.

The resulting CFG $G_2$ has $L(G_2) = L(G_1) \cup \{a\}$ and all rules have the form $A \rightarrow c$ or $A \rightarrow X$ with $X \in VVV^*$ i.e. $|X| \geq 2$.

Final "silly" steps making RHS have length = 2 were already discussed orally.
Unit Rules Example:

\[ S \rightarrow AB | cA | cE \quad A \rightarrow SS | S | Sa \quad B \rightarrow AS | a \]

\[ \text{NULLABLE } = \{A, B, S\} \]

Grammar \( G_1 \) is:

\[ G_1: S \rightarrow AB | A | B | cA | c \quad A \rightarrow SS | S | Sa | a \quad B \rightarrow AS | a \]

Graph \( H \):

A and B get exactly the same here. "Yuck!"

We still need on aliases \( X_c \rightarrow c \quad X_a \rightarrow a \) here for CHNF. "Yuck!"