Chomsky Normal Form

Def 0.1 A grammar Let $G = (N, \Sigma, P, S)$ is in *Chomsky Normal Form* if every production is either of the form $A \rightarrow BC$ or $A \rightarrow \sigma$ where $\sigma \in \Sigma$.

NOTE- the algorithm below is a bit different than the one we did in class. But the ideas are similar.

Theorem 0.2 *There exists an algorithm that will, given a CFG $G = (N, \Sigma, P, S)$ with no useless nonterminals, no e-productions, no unit productions, output a grammar $G' = (N', \Sigma, P', S')$ in Chomsky Normal Form such that such that $L(G) = L(G')$.*

Proof:
Look at each rule of the form $A \rightarrow \alpha_1\alpha_2 \cdots \alpha_L$. Note that $L \neq 1$ since that would be a unit production. If $L = 2$ then we do nothing since the production is already of the right form. So we assume $L \geq 3$. We do the following.

1. Replace every terminal α_i with nonterminal $[\alpha_i]$ and add the rule $[\alpha_i] \rightarrow \alpha_i$.

2. Note that the rule is now of the form

   $A \rightarrow \beta_1 \cdots \beta_L$

   where each β_i is a nonterminal.

   Replace this with the following:

   $A \rightarrow [\beta_1 \cdots \beta_{L-1}]\beta_L$

   $[\beta_1 \cdots \beta_{L-1}] \rightarrow [\beta_1 \cdots \beta_{L-2}]\beta_{L-1}$

   $[\beta_1 \cdots \beta_{L-2}] \rightarrow [\beta_1 \cdots \beta_{L-3}]\beta_{L-2}$

   etc until

   $[\beta_1\beta_2\beta_3]] \rightarrow [\beta_1\beta_2]\beta_3$

   $[\beta_1\beta_2] \rightarrow \beta_1\beta_2$.

∎