

Chomsky normal form

Every rule is of the form:

$$A \rightarrow BC$$

$$A \rightarrow a$$

In addition, we permit the rule $S \rightarrow \varepsilon$ for start symbol S .

Theorem

Every CFG has an equivalent CFG in Chomsky normal form.

Algorithm

1. If the start symbol S appears on the right hand side of a rule, create a new start symbol.
2. Remove ε in rules other than from the start symbol.
3. Remove unit rules.
4. Transform rules with mixed variables and terminals.
5. Split long rules in many shorter rules

PDA generate exactly the Context-free languages

We already saw how to transform CFGs to PDAs. There is also an algorithm to transform PDAs to CFGs. Since this algorithm has mostly theoretical purpose, we will not cover it in class.

Removing ε

1. Identify which variables are nullable
2. Add productions for each combination of nullable variable

Class exercise

The following context-free grammar has some ε production rules. Use the algorithm we saw in class to build an equivalent context-free grammar without ε production rules (except for the start symbol).

$$\begin{aligned} S &\rightarrow JC \mid K \\ J &\rightarrow aJbC \mid \varepsilon \\ C &\rightarrow cC \mid \varepsilon \\ K &\rightarrow aKc \mid B \\ B &\rightarrow bB \mid \varepsilon \end{aligned}$$