# Chomsky Normal Form (CNF)

**Chomsky Normal Form (CNF)** is a very common form for context-free grammars (CFGs), and worth learning. As noted on Wikipedia, CNF is used in the CYK algorithm. The CYK algorithm will, given a CNF grammar and a string, decide whether the string is recognized by the grammar. And it does so quite efficiently! CYK can be modified easily to also generate a parse tree (automatically).

CNF is a grammar having the following general forms:

$A \rightarrow BC$ // two variables on the right
$A \rightarrow b$ // a single symbol on the right
$S \rightarrow \epsilon$ // the empty string

Every CFG can be converted to an equivalent one in CNF. There are a few variations on the conversion procedure into CNF. The one we covered in class has four steps:

1. Start with a new $S_0$ (a.k.a., eliminate start symbols on the right-hand side),
2. Eliminate $\epsilon$ rules of the form $A \rightarrow \epsilon$,
3. Eliminate unit rules of the form $A \rightarrow B$, and
4. Final clean-up into CNF formatting.

The following YouTube video does a nice job walking through the procedure using this specific method:
CNF Conversion Video