CS381 Fall 2000	**Handout 19**
October 13, 2000	

1. Reading: D. Kozen *Automata and Computability*, Lecture 21 J. Hopcroft and J. Ullman *Introduction to Automata Theory, etc.*, section 4.5, 4.6.

2. The main message of this lecture:

> **Not all context free grammars are really needed for generating context free languages. In fact, some proper subclasses of CFG with rather simple productions suffices to generate all of CFL's. Such simplified though still general CFG's are called normal forms.**

Definition 19.1. *Chomsky Normal Form (CNF)* is a CFG whose productions are of the form $A \longrightarrow BC$ or $A \longrightarrow a$, where A, B, C are nonterminal letters, a is a terminal letter.

Example 19.2. Consider CFG

$$S \longrightarrow aSb, \quad S \longrightarrow \epsilon$$

that generates the language $L = \{a^n b^n \mid n \geq 0\}$. Let us try to find its Chomsky Normal Form, if any. First of all, we have to exclude ϵ since no CNF can generate the null string (why?). So, we will try to find a CNF that generates the language $\{a^n b^n \mid n > 0\}$.

1. Getting rid of ϵ is easy.

$$S \longrightarrow aSb, \quad S \longrightarrow ab$$

Is this method scalable? It surely is! Whenever a CFG has productions $A \longrightarrow \alpha B \gamma$ and $B \longrightarrow \epsilon$ add $A \longrightarrow \alpha\gamma$ and drop $B \longrightarrow \epsilon$. In particular, if there are no $A \longrightarrow \alpha B \gamma$'s then just drop $B \longrightarrow \epsilon$, we are not going to use it anyway!.

2. Replacing terminal letters in all nontrivial right-hand sides of productions by fresh nonterminal ones.

$$S \longrightarrow ASB, \quad S \longrightarrow AB, \quad A \longrightarrow a, \quad B \longrightarrow b$$

3. Shortening right-hand sides of productions (new nonterminal letters are needed!).

$$S \longrightarrow AC, \quad C \longrightarrow SB, \quad S \longrightarrow AB, \quad A \longrightarrow a, \quad B \longrightarrow b$$

Done!

Theorem 19.3. *For any CFG G there is a CNF G′ such that* $L(G') = L(G) - \{\epsilon\}$.

Proof. Get rid of ϵ . Replace terminal letters in all nontrivial right-hand sides of productions by fresh nonterminal ones. Shorten right-hand sides of productions by using new nonterminal letters.

Example 19.4. PAREN$-\{\epsilon\}$.

$$S \longrightarrow [S], \quad S \longrightarrow SS, \quad S \longrightarrow \epsilon$$

$$S \longrightarrow [S], \quad S \longrightarrow SS, \quad S \longrightarrow [\,]$$

$$S \longrightarrow LSR, \quad S \longrightarrow SS, \quad S \longrightarrow LR, \quad L \longrightarrow [, \quad R \longrightarrow]$$

$$S \longrightarrow AR, \quad A \longrightarrow LS, \quad S \longrightarrow SS, \quad S \longrightarrow LR, \quad L \longrightarrow [, \quad R \longrightarrow]$$

Example 19.5. Palindromes ($\neq \epsilon$).

$$S \longrightarrow aSa \mid bSb \mid a \mid b \mid \epsilon$$

$$S \longrightarrow aSa \mid bSb \mid aa \mid bb \mid a \mid b$$

$$S \longrightarrow ASA \mid BSB \mid AA \mid BB \mid A \mid B, \quad A \longrightarrow a, \quad B \longrightarrow b$$

$$S \longrightarrow AC \mid BD \mid AA \mid BB \mid A \mid B, \quad C \longrightarrow SA, \quad D \longrightarrow SB, \quad A \longrightarrow a, \quad B \longrightarrow b$$

Definition 19.6. *Greibach Normal Form (GNF)* is a CFG whose productions are of the form $A \longrightarrow aB_1B_2 \dots B_n$, where a is terminal and B_i's are nonterminal letters, $n \geq 0$. In particular, when $n = 0$ the production becomes $A \longrightarrow a$.

Theorem 19.7. *For any CFG G there is a GNF G' such that $L(G') = L(G) - \{\epsilon\}$.*

Proof. Skip it.

Example 19.8. The Greibach normal form

$$S \longrightarrow aB \mid bA, \quad A \longrightarrow aS \mid bAA \mid a, \quad B \longrightarrow bS \mid aBB \mid b$$

generates exactly the set of nonnull strings over $\{a, b\}$ with equally many a's and b's.

Claim: all generated strings have equal number of a's and b's. Proof: by a straightforward induction on the length of derivation we establish that 'a generated string (possibly with non-terminals) has equal number of a's and b's, case non sensitive.

Claim: all nonnull strings with equal number of a's and b's are generated by this grammar. The proof of this claim will be left as a homework problem. Meanwhile consider some examples of derivations.

$$S \longrightarrow aB \longrightarrow aaBB \longrightarrow aaaBBB \longrightarrow aaabBB \longrightarrow aaabbB \longrightarrow aaabbb$$

$$S \longrightarrow aB \longrightarrow aaBB \longrightarrow aabSB \longrightarrow aabaBB \longrightarrow aababB \longrightarrow aababb$$

$$S \longrightarrow aB \longrightarrow aaBB \longrightarrow aabB \longrightarrow aabbS \longrightarrow aabbbA \longrightarrow aabbba$$

Homework problems. Page 307 of Kozen's book: $\sharp$ 1 (remaining part), $\sharp$ 4 (CNF only).