CS3231
Tutorial 7

1. Remove useless symbols from the following grammar using the algorithm done in class.

$$S \to A|AA|AAA$$
$$A \to ABa|ACa|a$$
$$B \to ABa|Ab|\epsilon$$
$$C \to Cab|CC$$
$$D \to CD|Cd|CEa$$
$$E \to b$$

2. Eliminate ϵ productions from the grammar

$$S \to ABaC$$
$$A \to AB$$
$$B \to b|\epsilon$$
$$C \to D|\epsilon$$
$$D \to d$$

3. Remove all unit productions from the grammar

$$S \to CBa|D$$
$$A \to bbC$$
$$B \to Sc|ddd$$
$$C \to eA|f|C$$
$$D \to E|SABC$$
$$E \to gh$$

4. Convert the following to Chomsky normal form grammar without useless symbols:

$$S \to AB|CA$$
$$A \to a$$
$$B \to BC|AB$$
$$C \to aB|b|ACC|\epsilon$$

5. Could you give an algorithm to test whether the language generated by a CFG is (a) empty, (b) finite, (c) infinite?

6. Give an algorithm that constructs $Unit(A)$ for all nonterminals A in a CFG.

7. (Hard) Greibach Normal Form: A grammar is said to be in Greibach Normal Form, if all the productions in the grammar are of the form: $A \rightarrow a\alpha$, where a is a terminal and α is a string of zero or more variables (non-terminals). Prove that, for every non-empty context free language L, which does not contain ϵ, one can have a Greibach Normal Form grammar.

   Hint: Assume the original grammar given for the language L is in Chomsky Normal form. Assume that the variables in the grammar are $A_1, \ldots, A_m$. Let G_0 be the original grammar.

   (a) First, inductively define G_i (generating the same language L) to have the following properties:

   (P1) G_i has variables $A_1, \ldots, A_m$ and $B_1, \ldots, B_i$,

   (P2) all the productions of G_i are of form (i) $A_j \rightarrow \alpha$ (where α starts with either a terminal, or a variable A_r, with $r \geq \min(i+1, j+1)$), OR (ii) $B_j \rightarrow \alpha$, where α starts with a terminal or A_k for some k.

   The above can be achieved as follows. Suppose in G_{i-1} we have productions of the form $A_i \rightarrow \alpha_1 \mid \alpha_2 \mid \ldots \mid \alpha_r$ and $A_i \rightarrow A_i\beta_1 \mid A_i\beta_2 \mid \ldots \mid A_i\beta_w$, where α_s either start with a terminal or A_k for some $k > i$ (note that by inductive property P2, above holds). Now replace the above productions by:

   $A_i \rightarrow \alpha_1 B_i \mid \alpha_2 B_i \mid \ldots \mid \alpha_r B_i$

   $A_i \rightarrow \alpha_1 \mid \alpha_2 \mid \ldots \mid \alpha_r$

   and

   $B_i \rightarrow \beta_1 \mid \beta_2 \mid \ldots \mid \beta_w$,

   $B_i \rightarrow \beta_1 B_i \mid \beta_2 B_i \mid \ldots \mid \beta_w B_i$

   Here if β_r starts with a $B_{r'}$, $r' < i$, then replace $B_{r'}$ in these productions by the RHS of all productions of $B_{r'}$.

   (note that above is "correct" replacement as the language generated does not change).

   Now for $j > i$, replace each production in G_{i-1} of form $A_j \rightarrow A_i\gamma$ by the set of productions $A_j \rightarrow \alpha_1 B_i\gamma \mid \alpha_2 B_i\gamma \ldots \mid \alpha_r B_i\gamma$ and

   $A_j \rightarrow \alpha_1\gamma \mid \alpha_2\gamma \ldots \mid \alpha_r\gamma$.

   Now verify that the grammar so generated, G_i, satisfies the properties (P1) and (P2).

   (b) Let us rename the variables in G_m as

   B_i renamed to C_i

   A_j renamed to C_{m+j}.

   Then, we have the property that any production of form $C_r \rightarrow \alpha$, has α starting with either a terminal or a variable C_w, where $w > r$. Use this property to convert the grammar into Greibach normal form.