Getting Rid of Useless Nonterminals
and Getting rid of e-productions

1 Getting Rid of Useless Nonterminals

Def 1.1 Let $G = (N, \Sigma, P, S)$ be a CFG. Let $A \in N$ and $\alpha \in (N \cup \sigma)^*$. $A \Rightarrow \alpha$ means that there is a sequence of applications of productions that take you from A to α. (This is often written with a G under the $\Rightarrow$ and a * over it.)

Def 1.2 Let $G = (N, \Sigma, P, S)$ be a CFG such that $L(G) \neq \emptyset$. A nonterminal A is *useful* if the following two hold.

- There exists $w \in \Sigma^*$ such that $A \Rightarrow w$.
- There exists $\alpha, \beta \in (N \cup \Sigma)^*$ such that $S \Rightarrow \alpha A \beta$.

Note 1.3 If $L(G) = \emptyset$ then it's not clear how you can define useful nonterminals since S would be useless. To avoid this problem we only deal with G such that $L(G) \neq \emptyset$.

Theorem 1.4 *There exists an algorithm that will, given a CFG* $G = (N, \Sigma, P, S)$ *such that* $L(G) \neq \emptyset$ *output another CFG* $G = (N', \Sigma, P', S')$ *such that* $L(G) = L(G')$ *and* N' *has no useless nonterminals. The algorithm will run in time* $O(|P||N|)$.

Proof: We give the algorithm, show that it works in the correct time, but do not prove that it works.
Actually we give two algorithms.
The first one finds all nonterminals A such that there exists $w \in \Sigma^*$ such that $A \Rightarrow w$. We call this set $USEFULL^{\text{alphabet}}$. During the algorithm $USEFULL_i^{\text{alphabet}}$ will be the set of nonterminals that can get to some element of Σ^* within i steps.

1. Input (N, Σ, P, S).
2. $USEFULL_1^{\text{alphabet}} = \{A :$ there is a production of the form $A \rightarrow w$ where $w \in \Sigma^*$ $\}$

3. For $i = 1$ to $|N| - 1$

   $USEFULL_{i+1}^{\text{alphabet}} = USEFULL_i^{\text{alphabet}} \cup \{A : \exists\alpha \in (USEFULL_i^{\text{alphabet}} \cup \Sigma)^*$ and a producti

4. $USEFULL^{\text{alphabet}} = USEFULL_{|N||}^{\text{alphabet}}$.

Clearly this algorithm works in $O(|P||N|)$ steps.
The second one finds all nonterminals A such that there exists $\alpha, \beta \in (N \cup \Sigma)^*$ such that $S \Rightarrow \alpha A \beta$. We call this set $USEFULL^S$. During the algorithm $USEFULL_i^S$ will be the set of nonterminals that S can get to within i steps.

1. Input (N, Σ, P, S).

2. $USEFULL_0^S = \{S\}$

3. For $i = 0$ to $|N| - 1$

   $USEFULL_{i+1}^S = USEFULL_i^S \cup \{A : \exists B \in USEFULL_i^S, \exists\alpha, \beta \in (N \cup \Sigma)^\star, B \to \alpha A \beta.$

4. $USEFULL^S = USEFULL_{|N|}^S$

The algorithm clearly runs in $O(|P||N|)$ steps
Here is our algorithm: Run the first algorithm to obtain $USEFULL^{\text{alphabet}}$. REMOVE all nonterminals that are not in $USEFULL^{\text{alphabet}}$ from N. REMOVE all productions that use the nonterminals that are not in $USEFULL^{\text{alphabet}}$. We still call the grammar $G = (N, \Sigma, P, S)$ even though N and P have changed.
Then do the second algorithm (on the reduced grammar). REMOVE all nonterminals that are not in $USEFULL^S$ from N. REMOVE all productions that use the nonterminals that are not in $USEFULL^S$. The resulting grammar is (N', Σ, P', S). ▮

Note 1.5 For you to think about: if we applied these algorithms in the reverse order then they might not get rid of all the useless nonterminals.

2 Getting Rid of e-Productions

I describe how to get rid of e-production informally. Doing it formally will be one of your HW problems.
If $A \rightarrow e$ is a production then you remove it, but then for every production of the form
$B \rightarrow \alpha A \beta$
you add the productino
$B \rightarrow \alpha A \beta$.
You keep doing this iteratively.