Formal Languages, Automata and Codes

Oleg Gutik

Lecture 18

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

6.2 Two Important Normal Forms

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

6.2 Two Important Normal Forms

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

6.2 Two Important Normal Forms

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

There are many kinds of normal forms we can establish for context-free grammars. Some of these, because of their wide usefulness, have been studied extensively. We consider two of them briefly.

Chomsky Normal Form

One kind of normal form we can look for is one in which the number of symbols on the right of a production is strictly limited. In particular, we can ask that the string on the right of a production consist of no more than two symbols. One instance of this is the *Chomsky normal form*.

Definition 6.4

A context-free grammar is in *Chomsky normal form* if all productions are of the form

$$A \to BC$$

or

$$A \to a,$$

where A, B, C are in V, and a is in T.

6.2 Two Important Normal Forms

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Example 6.7

The grammar

$$S \to AS|a,$$
$$A \to SA|b$$

is in Chomsky normal form. The grammar

$$S \to AS|AAS,$$
$$A \to SA|aa$$

is not; both productions $S \to AAS$ and $A \to aa$ violate the conditions of Definition 6.4.

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \qquad (1)$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \quad \text{if} \quad x_i \quad \text{is in} \quad V,$$

and

$$C_i = B_a \quad \text{if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \qquad (1)$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \qquad (1)$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \quad \text{if} \quad x_i \quad \text{is in} \quad V,$$

and

$$C_i = B_a \quad \text{if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \quad \text{if} \quad x_i \quad \text{is in} \quad V,$$

and

$$C_i = B_a \quad \text{if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \quad (1)$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \;\text{ if }\; x_i \;\text{ is in }\; V,$$

and

$$C_i = B_a \;\text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \ \text{ if } \ x_i \ \text{ is in } \ V,$$

and

$$C_i = B_a \ \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \quad \text{if} \quad x_i \quad \text{is in} \quad V,$$

and

$$C_i = B_a \quad \text{if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \qquad (1)$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \quad \text{if} \quad x_i \quad \text{is in} \quad V,$$

and

$$C_i = B_a \quad \text{if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \quad \text{if} \quad x_i \quad \text{is in} \quad V,$$

and

$$C_i = B_a \quad \text{if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

Theorem 6.6

Any context-free grammar $G = (V, T, S, P)$ with $\lambda \notin L(G)$ has an equivalent grammar $\widehat{G} = (\widehat{V}, \widehat{T}, S, \widehat{P})$ in Chomsky normal form.

Proof. Because of Theorem 6.5, we can assume without loss of generality that G has no λ-productions and no unit-productions. The construction of $\widehat{G}$ will be done in two steps.

Step 1. Construct a grammar $G_1 = (V_1, T, S, P_1)$ from G by considering all productions in P in the form

$$A \to x_1 x_2 \cdots x_n, \tag{1}$$

where each x_i is a symbol either in V or in T. If $n = 1$ then x_1 must be a terminal since we have no unit-productions. In this case, put the production into P_1. If $n \geqslant 2$ the introduce new variables B_a for each $a \in T$. For each production of P in the form (1) we put into P_1 the production

$$A \to C_1 C_2 \cdots C_n,$$

where

$$C_i = x_i \text{ if } x_i \text{ is in } V,$$

and

$$C_i = B_a \text{ if } x_i = a.$$

For every B_a we also put into P_1 the production

$$B_a \to a.$$

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \qquad (2)$$

or

$$A \to C_1 C_2 \cdots C_n, \qquad (3)$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \qquad (2)$$

or

$$A \to C_1 C_2 \cdots C_n, \qquad (3)$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots & \quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots & \quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \quad (2)$$

or

$$A \to C_1 C_2 \cdots C_n, \quad (3)$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots &\quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \quad (2)$$

or

$$A \to C_1 C_2 \cdots C_n, \quad (3)$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\cdots \quad \cdots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots & \quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$

$$D_1 \to C_2 D_2,$$

$$\cdots \quad \cdots$$

$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots &\quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots &\quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots &\quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots & \quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$A \to C_1 D_1,$$
$$D_1 \to C_2 D_2,$$
$$\ldots \quad \ldots$$
$$D_{n-2} \to C_{n-1} C_n.$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \ldots &\quad \ldots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \dots$ and put into $\widehat{P}$ the productions

$$
\begin{aligned}
A &\to C_1 D_1,\\
D_1 &\to C_2 D_2,\\
\dots &\quad \dots\\
D_{n-2} &\to C_{n-1} C_n.
\end{aligned}
$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ∎

This part of the algorithm removes all terminals from productions whose right side has length greater than one, replacing them with newly introduced variables. At the end of this step we have a grammar G_1 all of whose productions have the form

$$A \to a, \tag{2}$$

or

$$A \to C_1 C_2 \cdots C_n, \tag{3}$$

where $C_i \in V_1$.

It is an easy consequence of Theorem 6.1 that $L(G_1) = L(G)$.

Step 2. In the second step, we introduce additional variables to reduce the length of the right sides of the productions where necessary. First we put all productions of the form (2) as well as all the productions of the form (3) with $n = 2$ into $\widehat{P}$. For $n > 2$ we introduce new variables $D_1, D_2, \ldots$ and put into $\widehat{P}$ the productions

$$\begin{aligned} A &\to C_1 D_1, \\ D_1 &\to C_2 D_2, \\ \cdots &\quad \cdots \\ D_{n-2} &\to C_{n-1} C_n. \end{aligned}$$

Obviously, the resulting grammar $\widehat{G}$ is in Chomsky normal form. Repeated applications of Theorem 6.1 will show that $L(G_1) = L(\widehat{G})$, so $L(\widehat{G}) = L(G)$.

This somewhat informal argument can easily be made more precise. We shall leave this to the reader. ■

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$

$$A \to aab,$$

$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$

$$A \to B_aB_aB_b,$$

$$B \to AB_c,$$

$$B_a \to a,$$

$$B_b \to b,$$

$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8

Convert the grammar with productions

$$S \to ABa,$$
$$A \to aab,$$
$$B \to Ac$$

to Chomsky normal form.

As required by the construction of Theorem 6.6, the grammar does not have any λ-productions or any unit-productions.

In Step 1, we introduce new variables B_a, B_b, B_c and use the algorithm to get

$$S \to ABB_a,$$
$$A \to B_aB_aB_b,$$
$$B \to AB_c,$$
$$B_a \to a,$$
$$B_b \to b,$$
$$B_c \to c.$$

Example 6.8 (continuation)

In the second step, we introduce additional variables to get the first two productions into normal form and we get the final result

$$\begin{aligned} S &\to AD_1, \\ D_1 &\to BB_a, \\ A &\to B_aD_2, \\ D_2 &\to B_aB_b, \\ B &\to AB_c, \\ B_a &\to a, \\ B_b &\to b, \\ B_c &\to c. \end{aligned}$$

Example 6.8 (continuation)

In the second step, we introduce additional variables to get the first two productions into normal form and we get the final result

$$
\begin{aligned}
S &\to AD_1, \\
D_1 &\to BB_a, \\
A &\to B_aD_2, \\
D_2 &\to B_aB_b, \\
B &\to AB_c, \\
B_a &\to a, \\
B_b &\to b, \\
B_c &\to c.
\end{aligned}
$$

Example 6.8 (continuation)

In the second step, we introduce additional variables to get the first two productions into normal form and we get the final result

$$\begin{aligned} S &\to AD_1, \\ D_1 &\to BB_a, \\ A &\to B_aD_2, \\ D_2 &\to B_aB_b, \\ B &\to AB_c, \\ B_a &\to a, \\ B_b &\to b, \\ B_c &\to c. \end{aligned}$$

Example 6.8 (continuation)

In the second step, we introduce additional variables to get the first two productions into normal form and we get the final result

$$
\begin{aligned}
S &\to AD_1, \\
D_1 &\to BB_a, \\
A &\to B_aD_2, \\
D_2 &\to B_aB_b, \\
B &\to AB_c, \\
B_a &\to a, \\
B_b &\to b, \\
B_c &\to c.
\end{aligned}
$$

Example 6.8 (continuation)

In the second step, we introduce additional variables to get the first two productions into normal form and we get the final result

$$\begin{aligned} S &\to AD_1, \\ D_1 &\to BB_a, \\ A &\to B_a D_2, \\ D_2 &\to B_a B_b, \\ B &\to AB_c, \\ B_a &\to a, \\ B_b &\to b, \\ B_c &\to c. \end{aligned}$$

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

6.2 Two Important Normal Forms

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

Greibach Normal Form

Another useful grammatical form is *Greibach normal form*. Here we put restrictions not on the length of the right sides of a production, but on the positions in which terminals and variables can appear. Arguments justifying Greibach normal form are a little complicated and not very transparent. Similarly, constructing a grammar in Greibach normal form equivalent to a given context-free grammar is tedious. We therefore deal with this matter very briefly. Nevertheless, Greibach normal form has many theoretical and practical consequences.

Definition 6.5

A context-free grammar is said to be in *Greibach normal form* if all productions have the form

$$A \to ax,$$

where $a \in T$ and $x \in V^*$.

If we compare this with Definition 5.4, the we see that the form $A \to ax$ is common to both Greibach normal form and s-grammars, but Greibach normal form does not carry the restriction that the pair (A, a) occur at most once. This additional freedom gives the Greibach normal form a generality not possessed by s-grammars.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

If a grammar is not in Greibach normal form, we may be able to rewrite it in this form with some of the techniques encountered above. Here are two simple examples.

Example 6.9

The grammar

$$S \to AB,$$
$$A \to aA|bB|b,$$
$$B \to b$$

is not in Greibach normal form. However, using the substitution given by Theorem 6.1, we immediately get the equivalent grammar

$$S \to aAB|bBB|bB,$$
$$A \to aA|bB|b,$$
$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

Example 6.10

Convert the grammar

$$S \to abSb|aa$$

into Greibach normal form.

Here we can use a device similar to the one introduced in the construction of Chomsky normal form. We introduce new variables A and B that are essentially synonyms for a and b, respectively. Substituting for the terminals with their associated variables leads to the equivalent grammar

$$S \to aBSB|aA,$$

$$A \to a,$$

$$B \to b,$$

which is in Greibach normal form.

6.2 Two Important Normal Forms

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

In general, though, neither the conversion of a given grammar to Greibach normal form nor the proof that this can always be done is a simple matter. We introduce Greibach normal form here because it will simplify the technical discussion of an important result in the next lectures. However, from a conceptual viewpoint, Greibach normal form plays no further role in our discussion, so we only quote the following general result without proof.

Theorem 6.7

For every context-free grammar G with $\lambda \notin L(G)$, there exists an equivalent grammar $\widehat{G}$ in Greibach normal form.

Thank You for attention!