I.2.1. Formal Languages (10.1)

I.2.2. Grammars and Derivations (10.2)

I.2.3. Chomsky Hierarchy (12.1)

I.2.4. Modularity and BNF notation (10.3)
  I.2.4.1. A simple modular grammar (10.3.1.)
  I.2.4.2. The import construct (10.3.2.)

I.2.5. Derivation Trees for Context-Free Grammars (14.1)
The Import Construct and Modular Grammars (10.3.2)

Definition

Let $H$ be a grammar

<table>
<thead>
<tr>
<th>Grammar</th>
<th>$H$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import</td>
<td>$G$</td>
</tr>
<tr>
<td>Terminals</td>
<td>$T_H$</td>
</tr>
<tr>
<td>Nonterminals</td>
<td>$N_H$</td>
</tr>
<tr>
<td>Start symbol</td>
<td>$S_H$</td>
</tr>
<tr>
<td>Productions</td>
<td>$P_H$</td>
</tr>
</tbody>
</table>

which imports grammar $G$
Let the grammar imported by $H$ be defined as follows:

<table>
<thead>
<tr>
<th>Grammar</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>grammar</td>
<td>$G$</td>
</tr>
<tr>
<td>terminals</td>
<td>$T_G$</td>
</tr>
<tr>
<td>nonterminals</td>
<td>$N_G$</td>
</tr>
<tr>
<td>start symbol</td>
<td>$S_G$</td>
</tr>
<tr>
<td>productions</td>
<td>$P_G$</td>
</tr>
</tbody>
</table>
Then $H$ denotes the grammar $F$ which
- has start symbol $S_H$
- and as nonterminals/terminals/productions the union of the nonterminals/terminals/productions of $G$ and $H$.

This grammar $F$ is called the **flattened form** of $H$. 
Definition (Importing Grammars, Cont)

The flattened form $F$ of $H$ is therefore defined as follows:

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>grammar</td>
<td>$F$</td>
</tr>
<tr>
<td>terminals</td>
<td>$T_G \cup T_H$</td>
</tr>
<tr>
<td>nonterminals</td>
<td>$N_G \cup N_H$</td>
</tr>
<tr>
<td>start symbol</td>
<td>$S_H$</td>
</tr>
<tr>
<td>productions</td>
<td>$P_G \cup P_H$</td>
</tr>
</tbody>
</table>
I.2.1. Formal Languages (10.1)

I.2.2. Grammars and Derivations (10.2)

I.2.3. Chomsky Hierarchy (12.1)

I.2.4. Modularity and BNF notation (10.3)
   I.2.4.1. A simple modular grammar (10.3.1.)
   I.2.4.2. The import construct (10.3.2.)

I.2.5. Derivation Trees for Context-Free Grammars (14.1)
Notations for Derivation Trees

We introduce the following notations:

- If $D$ is a derivation tree, then
  - $\text{label}(D)$ is the label of the root of $D$.
  - $\text{children}(D) = [D_1, \ldots, D_n]$ means that the children (immediate subtrees of $D$) are $D_1, \ldots, D_n$ (read from left to right).
  - $\text{frontier}(D)$ denotes the frontier of $D$.

- If $D_1, \ldots, D_n$ are derivation trees, $A$ a nonterminal, let $D := \text{tree}(A, D_1, \ldots, D_n)$ be the derivation tree s.t.
  - $\text{label}(D) = A$,
  - $\text{children}(D) = [D_1, \ldots, D_n]$.

- A special derivation tree is the tree with only one node, namely the root and no subtrees for this node. It is called the trivial derivation tree.
Inductive Definition of Derivations

We can define the set of derivation trees as well inductively as follows:

**Definition**

We define the set of derivation trees $D$ for a CFG $G = (T, N, S, P)$ inductively together with

- $\text{label}(D) \in T \cup N \cup \{\epsilon\}$,
- $\text{children}(D)$, a list of derivation trees,
- $\text{frontier}(D) \in (T \cup N)^*$.

as follows:

- If $A \in T \cup N \cup \{\epsilon\}$, then
  - $D := \text{tree}(A)$ is a derivation tree, (the **trivial derivation tree**),
  - $\text{label}(D) := A$,
  - $\text{children}(D) = [ ]$,
  - $\text{frontier}(D) := A$. 
We can define the set of derivation trees as well inductively as follows:

**Definition (Cont)**

- If \( A \in N, D_1, \ldots, D_n \) are derivation trees,

\[
A \rightarrow \text{label}(D_1) \cdot \text{label}(D_2) \cdot \cdots \cdot \text{label}(D_n) \text{ is a production}
\]

and \( n = 1 \lor \forall i. \text{label}(D_i) \neq \epsilon \), then

- \( D := \text{tree}(A, D_1, \ldots, D_n) \text{ is a derivation tree} \),
- \( \text{label}(D) := A \),
- \( \text{children}(D) := [D_1, \ldots, D_n] \),
- \( \text{frontier}(D) := \text{frontier}(D_1) \cdot \text{frontier}(D_2) \cdot \cdots \cdot \text{frontier}(D_n) \).
Size of a Derivation Tree

We want show that for every derivation tree we can find a corresponding derivation and vice versa. For this we need a measure of the size of derivation tree.

Definition

The size $\text{size}(D)$ of a derivation tree $D$ is defined by induction on the definition of trees:

- If $D = \text{tree}(x)$, then $\text{size}(D) := 1$.
- If $D = \text{tree}(A, D_1, \ldots, D_n)$, then
  $$\text{size}(D) := 1 + \text{size}(D_1) + \cdots + \text{size}(D_n).$$

We define as well the height $\text{height}(D)$ of a derivation tree:

- If $D = \text{tree}(x)$, then $\text{height}(D) := 1$.
- If $D = \text{tree}(A, D_1, \ldots, D_n)$, then
  $$\text{height}(D) := 1 + \max\{\text{size}(D_1), \ldots, \text{size}(D_n)\}.$$
Proof of Theorem I.2.5.1.

Theorem I.2.5.1. follows by Lemma I.2.5.2. below. We first introduce the notion of a derivation forest, which generalises derivation trees.
For proving the equivalence of derivation trees and derivations, we need to deal with derivations $w \Rightarrow^* w'$ where $w$ is a string. Such derivations correspond to derivation forests, as defined as follows:

**Definition**

Let $G = (T, N, S, P)$ be a CFG, $w, w' \in (T \cup N)^*$, $w \neq \epsilon$. Let $w = x_1, \ldots, x_l$, $x_i \in T \cup N$.

A **derivation forest** with root $w$ and frontier $w'$ is a list of derivations $[D_1, \ldots, D_l]$ s.t.

- $\text{label}(D_i) = x_i$,
- $\text{frontier}(D_1).\text{frontier}(D_2).\cdots.\text{frontier}(D_l) = w'$.

Furthermore $\text{size}([D_1, \ldots, D_l]) := \text{size}(D_1) + \cdots + \text{size}(D_l)$. 
Lemma I.2.5.2. (Derivation Trees and Language Generation)

Let \( G = (T, N, S, P) \) be a CFG, \( A \in T \), \( w, w' \in (T \cup N)^* \). Then the following are equivalent

1. There exist a derivation forest \( D \) with root \( w \) and frontier \( w' \).
2. \( w \Rightarrow^* w' \).

In case \( w' \in T^* \), the derivation sequence \( w \Rightarrow^* w' \) can both be chosen as a left-most and as a right-most derivation sequence.
Proof (1) ⇒ (2)

- The proof is by induction on \( \text{size}(D) \).
- We will do it in such a way that we get, in case all leaves are terminal symbols, a left-most derivation.
- A right most derivation can be obtained by choosing instead of the left-most the right most non-trivial derivation tree.
- Let \( w = x_1 \cdots x_n, D = [D_1, \ldots, D_n] \).
- If all \( D_i \) are trivial, then \( w' = w, w \Rightarrow^* w' \).
- So assume at least one \( D_i \) is non-trivial. Let \( D_k \) be the left-most non-trivial derivation tree, i.e. \( D_1, \ldots, D_{k-1} \) are trivial, \( D_k \) is non-trivial.
- Let \( D_k = \text{tree}(A, D'_1, \ldots, D'_l), \text{label}(D'_i) = x'_i, x_k = A \).
Proof (1) ⇒ (2)

Then

\[ A \rightarrow x'_1 \cdots x'_i \]

is a production, and we have that with

\[ w_1 := x_1 \cdots x_{k-1} x'_1 x'_2 \cdots x'_i x_{k+1} x_{k+2} \cdots x_n \]

that

\[ w = x_1 \cdots x_{k-1} A x_{k+1} \cdots x_n \Rightarrow w_1 \]

is a one-step derivation.

In case \( w' \in T^* \), we have \( x_i \in T \), and this one-step derivation was left-most.

We have that

\[ [D_1, \ldots, D_{k-1}, D'_1, \ldots, D'_i, D_{k+1}, D_{k+2}, \ldots, D_n] \]

is a derivation forest with root \( w_1 \) and frontier \( w' \), and has size \( \text{size}(D) - 1 \).
Proof (1) ⇒ (2)

- By IH there exist a derivation $w_1 \Rightarrow^* w'$, which in case of $w' \in T^*$ can be chosen as a left-most derivation sequence.
- Therefore $w \Rightarrow w_1 \Rightarrow^* w'$ is a derivation, which in case of $w' \in T^*$ can be chosen as a left-most derivation sequence.
Proof is by induction on the length of the derivation \( w \Rightarrow^* w' \).

Let \( w = x_1, \ldots, x_n \).

In case the length is 0, \( w' = w \), and we can choose \( D = [\text{tree}(x_1), \ldots, \text{tree}(x_n)] \).

Otherwise, assume that \( x_k = A \) is a non-terminal, \( A \rightarrow y_1 \cdots y_l \) (with \( y_i \in T \cup N \) or \( l = 1 \land y_1 = \epsilon \)). Let

\[
w_1 := x_1 \cdots x_{k-1} Ax_{k+1} \cdots x_n \Rightarrow x_1 \cdots x_{k-1} y_1 y_2 \cdots y_l x_{k+1} \cdots x_n
\]

Assume that the derivation is

\[
w = x_1 \cdots x_{k-1} Ax_{k+1} \cdots x_n \Rightarrow w_1 \Rightarrow^* w'
\]

By IH there exist a derivation forest

\[
[D_1, \ldots, D_{k-1}, D'_1, \ldots, D'_l, D_{k+1}, \ldots, D_n]
\]

with root \( w_1 \) and frontier \( w' \).
Now

\[ [D_1, \ldots, D_{k-1}, (A, D'_1, \ldots, D'_l), D_{k+1}, \ldots, D_n] \]

is a forest with root \( w \) and frontier \( w' \).
Lemma I.2.5.4. Uniqueness of Derivation (Trees)

The proof of Theorem I.2.5.3. is done by proving the following more general

**Lemma**

Let $G = (T, N, S, P)$ be a CFG, $w \in (T \cup N)^+$, $w' \in T^*$. 

1. Assume there are two different derivation forests with root $w$ and frontier $w'$. Then there exist two different left-most and two different right-most derivations of $w \Rightarrow^* w'$.

2. Assume there are two different left-most derivations or two different right-most derivations of $w \Rightarrow^* w'$. Then there exist two different derivation forests of with root $w$ and frontier $w'$. 
Proof of the Lemma I.2.5.4. (1)

- We prove only that there exist two different left-most derivations.
- Let \([D_1, \ldots, D_n]\) and \([D'_1, \ldots, D'_n]\) be two different derivation forests with root \(w\) and frontier \(w'\).
- Induction on the length of the first derivation forest.
- If all \(D_i\) are trivial, then \(w \in T^*, w = w'\), but then \(D'_i = D_i\), which is not possible.
- Let \(D_k\) be the first non-trivial derivation tree, i.e. \(D_1, \ldots, D_{k-1}\) are trivial.
- Let \(D_k = \text{tree}(A, D'_1, \ldots, D'_l), \text{label}(D'_i) = y_i, A \rightarrow y_1 \cdots y_l\) a production.
- Let \(w_1 := x_1 \cdots x_{k-1}y_1 \cdots y_lx_{k+1} \cdots x_n\).
- We have \(D'_i\) are trivial for \(i < k\), \(D_k\) must be non-trivial (since it has as label a nonterminal).
Proof of the Lemma I.2.5.4. (1)

Case 1: $D'_k$ has the same production at the root, i.e.

$D'_k = \text{tree}(A, D''', \ldots, D''')$, \text{label}(D''') = y_i.$

Then

$[D_1, \ldots, D_{k-1}, D'_1, \ldots, D''', D_{k+1}, \ldots, D_n]$

and

$[D'_1, \ldots, D'_{k-1}, D'''', \ldots, D''', D'_{k+1}, \ldots, D'_n]$

must be different.

By IH there exist two different left-most productions $w_1 \Rightarrow w'$.

Therefore we obtain two different left-most productions $w \Rightarrow w_1 \Rightarrow w'$. 
Case 2: $D'_k$ has a different production at the root, i.e. 
$D'_k = \text{tree}(A, D''', \ldots, D''')$, $\text{label}(D''') = y'_1 \cdot y'_1 \cdots y'_m \neq y_1 \cdots y_m$.

But then the first steps in the derivations constructed in the lemma are different, and we obtain two different left-most derivations.
Proof of the Lemma 1.2.5.4. (2)

This proof is similar, at the first place where the two derivations differ we construct two different derivation forests.