# Fundamentele Informatica 1 (I&E)

najaar 2015

`http://www.liacs.leidenuniv.nl/~vlietrvan1/fi1ie/`

**Rudy van Vliet**
kamer 124 Snellius, tel. 071-527 5777
rvvliet(at)liacs(dot)nl

college 6, 13 november 2015

4. Context-Free Languages

4.1. Using Grammar Rules to Define a Language

4.2. Context-Free Grammars: Definitions and More Examples

# 4. Context-Free Languages

| | | | |
|---|---|---|---|
| reg. languages | FA | reg. grammar | reg. expression |
| determ. cf. languages | DPDA | | |
| cf. languages | PDA | cf. grammar | |
| re. languages | TM | unrestr. grammar | |

## 4.1. Using Grammar Rules to Define a Language

**Example 4.1.** The language *AnBn*

$\mathit{AnBn} = \{a^i b^i \mid i \geq 0\}$

or

**1.** $\Lambda \in \mathit{AnBn}$.
**2.** For every $S \in \mathit{AnBn}$, also $aSb \in \mathit{AnBn}$.

**Example 4.2.** The language *Expr*

**1.** $a \in$ *Expr*.
**2.** For every $x$ and $y$ in *Expr*, also $x + y$ and $x * y$ are in *Expr*.
**3.** For every $x \in$ *Expr*, also $(x) \in$ *Expr*.

$a + (a * a)$

$a + a * a$

ambiguity

**Example 4.3.** Palindromes and Nonpalindromes

*Pal*

*NonPal*

$x = abbbbaaba$

**Example 4.3.** Palindromes and Nonpalindromes

*NonPal*

$x = abbbbaaba$

**1.** For every $A \in \{a, b\}^*$, $aAb$ and $bAa$ are elements of *NonPal*.
**2.** For every $S$ in *NonPal*, $aSa$ and $bSb$ are in *NonPal*
(and $aSb$ and $bSa$ . . . ).

**Example 4.4.** English and Programming Language Syntax

English:

$<$ declarative sentence $> \rightarrow$
$\quad <$ subject phrase $> <$ verb phrase $> <$ object $>$

"haste makes waste"
"the ends justify the means"
"we must extend our notion"

**Example 4.4.** English and Programming Language Syntax

Programming language:

$$
\begin{aligned}
<\text{statement}> \;\rightarrow\; \ldots \;&|\; <\text{if-statement}> \;|\; <\text{for-statement}> \\
&|\; <\text{compound-statement}>
\end{aligned}
$$

$$<\text{compound-statement}> \rightarrow \{ <\text{statement-sequence}> \}$$

$$
\begin{aligned}
<\text{statement-sequence}> \rightarrow \Lambda \;|\; \\
<\text{statement}> \; <\text{statement-sequence}>
\end{aligned}
$$

‘Complete’ programming language. . .

# 4.2. Context-Free Grammars: Definitions and More Examples

**Definition 4.6.** Context-Free Grammars

A *context-free grammar* (CFG) is a 4-tuple $G = (V, \Sigma, S, P)$, where

$V$ and $\Sigma$ are disjoint finite sets,
$S \in V$,
and $P$ is a finite set of formulas of the form $A \to \alpha$,
where $A \in V$ and $\alpha \in (V \cup \Sigma)^*$.

Elements of $\Sigma$ are called *terminal symbols*, or *terminals*, and elements of $V$ are *variables*, or *nonterminals*, $S$ is the *start* variable and elements of $P$ are *grammar rules* or *productions*.

Notation:

$A \to \alpha$

$\alpha \Rightarrow \beta$, $\qquad \alpha \Rightarrow^n \beta$, $\qquad \alpha \Rightarrow^* \beta$

$\alpha \Rightarrow_G \beta$, $\qquad \alpha \Rightarrow_G^n \beta$, $\qquad \alpha \Rightarrow_G^* \beta$

Term ‘context-free’:

If $\alpha \Rightarrow \beta$, then $\alpha = \ldots$ and $\beta = \ldots$

**Definition 4.7.** The Language Generated by a CFG

If $G = (V, \Sigma, S, P)$ is a CFG,
the language generated by $G$ is

$$L(G) = \{x \in \Sigma^* \mid S \Rightarrow_G^* x\}.$$

A language $L$ is a *context-free language* (CFL)
if there is a CFG $G$ with $L = L(G)$.

**Example 4.8.** The Language *AEqB*

$$AEqB = \{x \in \{a, b\}^* \mid n_a(x) = n_b(x)\}$$

A slide from lecture 4:

**Example 3.4.** Strings in $\{a, b\}^*$ in Which Both the Number of $a$'s and the Number of $b$'s are Even

$$(aa + bb + (ab + ba)(aa + bb)^*(ab + ba))^*$$

**Theorem 4.9.**

If $L_1$ and $L_2$ are context-free languages over an alphabet $\Sigma$, then

$$L_1 \cup L_2, \quad L_1 L_2 \quad \text{and } L_1^*$$

are also CFLs.

**Proof. . .**

**Example 4.10.**

The Language $\{a^ib^jc^k \mid j \neq i + k\}$