Classification of grammars into 4 groups, depending on the form of the productions:

- **Grammar of type 0:** no limitations
  1. **1:** context-sensitive
  2. **2:** context-free
  3. **3:** regular (or right linear)

**Definition:** grammar of type 0

Productions have the most general form \( \alpha \rightarrow \beta \), with \( \alpha \in V^* \cdot V_N \cdot V^* \) and \( \beta \in V^* \)

A language generated by a grammar of type 0 is called of type 0.

**Example:** grammar for \( L = \{ 0^m1^m | m, n \in \mathbb{N} \land m < n \} \)

\[ G = (\{ S \}, \{ 0, 1 \}, \{ P \}, S) \]

with \( P \):
1. \( S \rightarrow 0S1 \) generates \( 0\ldots 0S1\ldots 1 \) \( \frac{k \text{ times}}{k \text{ times}} \)
2. \( 0S \rightarrow S \) removes a 0 on the left of \( S \)
3. \( S \rightarrow 1 \) generates the final 1

**Derivation of 001111:**

\[ S \xrightarrow{1} 0S1 \xrightarrow{4} 00S11 \xrightarrow{1} 000S11 \xrightarrow{2} 00S11 \xrightarrow{3} 0111 \]

**Derivation of 111:**

\[ S \xrightarrow{4} 0S1 \xrightarrow{4} 00S11 \xrightarrow{2} 0S11 \xrightarrow{3} S11 \xrightarrow{2} 111 \]

\[ \boxed{1} \]
Note: 1) the application of productions could go on forever (e.g. rule 1 in the previous example)

2) grammars of type 0 allow for derivations that shorten the sentential form (see for instance the previous derivations)

Definition: grammar of type 1 (context-sensitive)

Productions have the form \( \alpha \to \beta \), with

\( \alpha \in V^* \cdot V_N \cdot V^* \), \( \beta \in V^+ \), and \( |\alpha| \leq |\beta| \)

These productions cannot shorten the length of the sentential form to which they are applied.

A language generated by a grammar of type 1 is called of type 1 or context-sensitive.

Example: grammar for \( L = \{a^n b^n c^n | n \geq 1\} \)

\[ G = (\{S, A, B, C\}, \{a, b, c\}, \{\}, S) \]

with \( P : \)

1) \( S \to aSBC \) \( \{ \) generate \( a \ldots aBCBC \ldots BC \)
2) \( S \to aBC \) \( \{ \) move the C's to the end
3) \( CB \to BC \) \( \{ \) generate the terminals from left to right
4) \( aB \to ab \) \( \{ \) Note; we cannot simply have \( B \to b \) and \( C \to c \)
5) \( bB \to bb \) \( \{ \) because this would generate many words not in \( L \)
6) \( bC \to bc \)
7) \( cC \to cc \)
Example (con’t)

Derivation of $aaabbbccc$:

1. $S \rightarrow aSBC$  
2. $aSBC \rightarrow aSBCBC$  
3. $aSBCBC \rightarrow aaaaBCBCBC$  
4. $aaaaBCBCBC \rightarrow aaaaBBBCBC$  
5. $aaabBBBCCC \rightarrow aaabbbBCCC$  
6. $aaabbbBCCC \rightarrow aaabbbccc$

Note: not each sequence of direct derivations leads to a sentence in $L(G)$

1. $S = \rightarrow aSBC = \rightarrow aSBCBC = \rightarrow aaaaBCBCBC = \rightarrow aaaaBBBCBC$
2. $aaaaBBBCBC \rightarrow aaabBCBCCC \rightarrow aaabBCBCCC$

we cannot apply any other production

Definition: grammar of type 2 (context-free)

Productions have the form $A \rightarrow \beta$, with $A \in V_n$ and $\beta \in V^+$

These productions are productions of type 1, with the additional requirement that on the left there is a single nonterminal

A language generated by a grammar of type 2 is called of type 2 or context-free

Example: grammar for $L = \{0^n1^m | m, n \in \mathbb{N} \land n < m \}$

$G = \{S, A, B \}, \{0, 1\}, \{S\}$

with $P$:  
1. $S \rightarrow A1B$
2. $A \rightarrow 0A11B$ generates $0^{l-1}1^{l+2}$ for $l \geq 1$
3. $B \rightarrow 11B$ generates $1^{k+1}$ for $k \geq 1$
Example: grammar for \( L = \{1^m 0^n | m, n \in \mathbb{N} \land m < n \} \)

\[ G = (\{S, A, B, 0, 1, \#, \} \cup \{\varepsilon\}, \{\#, \}, S, \{\varepsilon\}) \]

with \( B \): 1) \( S \rightarrow \varepsilon \varepsilon 0 \varepsilon S 0 A \)
2) \( A \rightarrow 1A 1A \)

\[ \text{generates } 0 \cdot 0^1 \cdot 1^0 \cdot 0^1 \cdot \ldots \cdot 0^1 \cdot 1^0 \cdot \varepsilon \text{ (lines 1, 3, 5, 7, 9)} \]
\[ \text{generates } \frac{\ldots \{\#\} \ldots \} \text{ (lines 2, 4, 6, 8, 10)} \]
\[ \text{possibly } \varepsilon = 0 \text{ or } \varepsilon \neq 0 \text{ but } k \geq 2 \text{ and } k \text{ even if not both} \]

**Definition:** grammar of type 3 (regular, right-linear)

Productions have the form \( A \rightarrow \varepsilon \), with \( A \in V_N \)

\( \varepsilon \in V_T \cup (V_T \cdot V_N) \)

(i.e. \( A \rightarrow aB \) or \( A \rightarrow a \) with \( A, B \in \) \( V_N \) and \( a \in V_T \))

A language generated by a grammar of type 3 is called of type 3 or regular

**Example:** \( \{a^m b^n | m < n \} \) is of type 3, since it is generated by the grammar \( S \rightarrow aS \varepsilon \)

**Note:** a grammar of type 3 is called linear because on the right-hand side of a production there is at most one nonterminal; it is called right-linear because the nonterminal is on the right of the terminal

**Exercise:** show that grammars of type 3 generate the class of regular languages that do not contain \( \varepsilon \)
Solution to the exercise:

Given a grammar $G = (V_N, V_T, P, S)$ we construct an NFA $A_G = (V_N \cup \{F\}, V_T, S, S, \{F\})$

with $B \in S(A, a)$ iff $A \rightarrow aB$

$F \in S(A, a)$ iff $A \rightarrow a$

Note that $A_G$ is constructed in such a way that $w \in L(A_G)$ if and only if $w \in L(G)$

Example: If $G = (\{S\}, \{a,b\}, \{S \rightarrow aS | b\}, S)$ then $A_G$ looks as follows

Conversely, given an NFA $A = (Q, \Sigma, S, q_0, F)$ we construct a grammar $G_A = (Q, \Sigma, P, q_0)$

with $p \rightarrow aq \in P$ iff $q \in S(p, a)$

$p \rightarrow a \in P$ iff $q \in S(p, a)$ and $q \in F$

Note that $G_A$ is constructed in such a way that $w \in L(G_A)$ if and only if $w \in L(A)$

Example: If $A = (\{q_0, q_1\}, \{a, b\}, S, q_0, \{q_0\})$ with $S(q_0, a) = \{q_0\}$, $S(q_0, b) = \{q_1\}$, and $S(q_1, a) = S(q_1, b) = \emptyset$ then $A_G$ has the productions $q_0 \rightarrow aq_0 | bq_1 b$

Notes: We are never able to produce a sentence out of this sentential form