Grammars

You should read the page: `https://en.wikipedia.org/wiki/Chomsky_hierarchy`

Grammars are defined in Definition 1.1 on page 21 of the sixth edition of Linz, and the language $L(G)$ *generated* by a grammar G is given in Definition 1.2 on page 22.

Definition of a Grammar. A *grammar* G has the following parts.

1. An alphabet Σ. The members of Σ are called *terminals.*
2. An alphabet V which is disjoint from Σ. The members of V are called *variables.* One member of V must be designated the *start symbol.* Members of Σ and V are both called *grammar symbols.*
3. A finite set of *replacement rules,* also called *productions.* Each rule has a *left hand side* and a *right hand side*, both of which are strings of grammar symbols. The left hand side may not be the empty string. A replacement rule is usually written as the left hand side, followed by an arrow, followed by the right hand side.

Language Generated by a Grammar. It is conventional to use capital Roman letters for the variables, S for the start symbol, either numerals or lower case Roman letters near the beginning of the alphabet for terminals, and lower case Roman letters near the end of the alphabet for strings. But these are not rules, just conventions.

Sentential Forms. Sentential forms of a grammar G are defined inductively. Let S be the start symbol of G.

1. S is a sentential form.
2. If $\ell \to r$ is a production of G, and if $x\ell y$ is a sentential form of G, then xry is also a sentential form of G. We write $x\ell y \Rightarrow xry$ in this case, and say that $x\ell y$ *derives* xry in one step.

$L(G)$ is defined to be all sentential forms of G which are strings over Σ. A sequence of sentential forms beginning at the start symbol and ending at a string $w \in L$ is called a *derivation* of w.

Example. Let G be the grammar where $\Sigma = \{a, b\}$, $V = \{S, A, B, C\}$, S is the start symbol, and the productions are as listed below.
1. $S \to aA$
2. $S \to bB$
3. $A \to bB$
4. $A \to aS$
5. $B \to aA$
6. $B \to bC$
7. $B \to \lambda$
8. $C \to aC$

9. $C \to bC$

Here is a derivation of $w = abaab$.

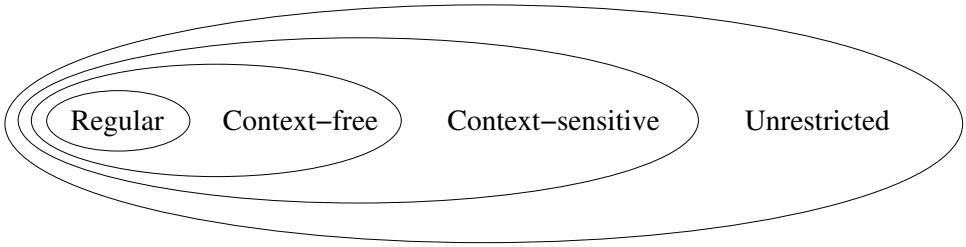
$S \Rightarrow aA \Rightarrow abB \Rightarrow abaA \Rightarrow abaaS \Rightarrow abaabB \Rightarrow abaab$

We sometimes write indices over the arrows to indicate which production is used at each step of a derivation:

$S \overset{1}{\Rightarrow} aA \overset{2}{\Rightarrow} abB \overset{5}{\Rightarrow} abaA \overset{4}{\Rightarrow} abaaS \overset{1}{\Rightarrow} abaabB \overset{7}{\Rightarrow} abaab$

Classes of Grammars

The following Euler diagram shows the grammar classes of the Chomsky hierarchy.



Chomsky Hierarchy of Grammars

Types of grammars are defined by the properties of the left hand and right hand sides.

1. In a *right linear* grammar, the left hand side of each production is a variable, and the right hand side is either a terminal followed by a variable or the empty string. The example grammar given above is right-linear.

2. In a *left linear* grammar, the left hand side of each production is a variable, and the right hand side is either a variable followed by a terminal or the empty string.

3. A grammar is *regular* if it is either left linear or right linear. A language is *regular* if it is generated by a regular grammar.

4. In a *context-free* grammar, the left hand side is a variable. (The right hand side may be any string of grammar symbols.) A language is *context-free* if it is generated by a regular grammar.

5. In a *context-sensitive* grammar, the left hand side cannot be the empty string, and the length of the left hand side must be less than or equal to the length of the right hand side. There is an exception to this rule: the production $S \to \lambda$ is permitted if S does not appear on the right hand side of any production. A language is *context-sensitive* if it is generated by a context-sensitive grammar.

6. in an *unrestricted* grammar, the left-hand side can be a non-empty string of grammar symbols, and the right hand side can be any string of grammar symbols. A language is *recursively enumerable* if it is generated by an unrestricted grammar.