1. Grammar:

It is a finite set of formal rules for generating correct sentences or meaningful correct sentences. A grammar is a set of rules which are used to construct a language (combine words to generate sentences).

Constitute of Grammar:

Grammar is basically composed of two basic elements:

1. **Terminal Symbols**: Terminal symbols are those which are the components of the sentences generated using a grammar and are represented using small case letter like a, b, c etc.
2. **Non-Terminal Symbols:** Non-Terminal Symbols are those symbols which take part in the generation of the sentence but are not the component of the sentence. Non-Terminal Symbols are also called **Auxiliary Symbols** and **Variables**. These symbols are represented using a capital letter like A, B, C, etc.

Formal Definition of Grammar:

Definition: **A grammar** is a quadruple (∑, V, S, P), where:

1. ∑ is a finite nonempty set called the **terminal alphabet**. The elements of ∑ are called the **terminals**.
2. V is a finite nonempty set disjoint from ∑. The elements of V are called the **nonterminals** or **variables**.
3. $S \in V$ is a distinguished nonterminal called the **start symbol**.
4. P is a finite set of **productions** (or **rules**) of the form

$$\alpha \to \beta$$

where $\alpha \in (\sum \cup V)^* V (\sum \cup V)^*$ and $\beta \in (\sum \cup V)^*$, i.e. α is a string of terminals and nonterminals containing at least one nonterminal and β is a string of terminals and nonterminals.

Example 1: Let G1 = ({0, 1}, {S, T,O, I}, S, P), where P contains the following productions:

$$S \to OT$$
$$S \to OI$$
$$T \to SI$$
$$O \to 0$$
$$I \to 1$$

The grammar G1 can be used to describe the set $\{0^n1^n | n \geq 1\}$.

Example 2: An article can be the word a or the:

$$A \rightarrow a$$
$$A \rightarrow the$$

• A noun can be the word dog, cat or rat:

$$N \rightarrow dog, \quad N \rightarrow cat, \quad N \rightarrow rat$$

A noun phrase is an article followed by a noun:

$$P \rightarrow AN$$

An verb can be the word loves, hates or eats:

$$V \rightarrow loves, \quad V \rightarrow hates, \quad V \rightarrow eats$$

A sentence can be a noun phrase, followed by a verb, followed by another noun phrase:

$$S \rightarrow PVP$$

Taken all together, a grammar G1 for a small subset of unpunctuated English:

$S \rightarrow PVP$	$A \rightarrow a$
$P \rightarrow AN$	$A \rightarrow the$
$V \rightarrow loves$	$N \rightarrow dog$
$V \rightarrow hates$	$N \rightarrow cat$
$V \rightarrow eats$	$N \rightarrow rat$

Each production says how to modify strings by substitution

• $x \rightarrow y$ says, substring x may be replaced by y.

2. The Language of the Grammar:

If G(V, T, P, S) is a CFG, then the language of G is $L(G) = \{w \text{ in } T^* \mid S \overset{*}{\Rightarrow}_G w\}$ i.e., the set of strings over T derivable from the start symbol. If G is a CFG, then L(G) a context-free language.

3. Derivation:

A derivation is a sequence of rewriting operations that starts with the string $\sigma = S$ and then repeats the following until σ contains only terminals.

A ***left-most derivation***$(\underset{lm}{\Rightarrow})$ is one in which the left-most non-terminal is always chosen as the next non-terminal to expand(Always replace the left-most variable by one of its rule-bodies).

A right-most derivation$(\underset{rm}{\Rightarrow})$ is one in which the right-most non-terminal is always chosen as the next non-terminal to expand(Always replace the rightmost variable by one of its rule-bodies).

$$E \rightarrow E+T, \quad E \rightarrow T, \quad T \rightarrow id$$

Derivations for id + id:

LEFTMOST	RIGHTMOST
$E \Rightarrow E+T$	$E \Rightarrow E+T$
$\Rightarrow T+T$	$\Rightarrow E+\mathbf{id}$
$\Rightarrow \mathbf{id}+T$	$\Rightarrow T+\mathbf{id}$
$\Rightarrow \mathbf{id}+\mathbf{id}$	$\Rightarrow \mathbf{id}+\mathbf{id}$

$\Rightarrow^*$is the transitive closure of $\Rightarrow$. If $\alpha \Rightarrow^* \beta$ holds, then α can be derived to β. The sequence $\alpha \Rightarrow \cdots \gamma \cdots \Rightarrow \beta$ is called the derivation of α to β. In the same sense $\Rightarrow^*$LM is the transitive closure of $\Rightarrow LM$.

This transitive closure can also be expressed as tree. Whenever a production is applied on a nonterminal, its node expands in the tree. Every symbol on the right-hand-side of the production becomes a child of this node. The advantage is that the order in which productions are applied does not matter and always result in the same tree. Such a tree is called a *derivation tree*.

Example 3: Recall the CFG for equal 0's and 1's:

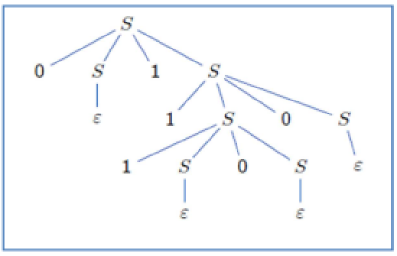
$$S \rightarrow 0S1S \mid 1S0S \mid \epsilon$$

The derivation for 011100

$$S \Rightarrow 0\underline{S}1S \Rightarrow 01\underline{S} \Rightarrow 011\underline{S}0S \Rightarrow 0111\underline{S}0S0S$$

$$\Rightarrow 01110\underline{S}0S \Rightarrow 011100\underline{S} \Rightarrow 011100$$

Here is derivation tree for 011100



Example 4:

The grammar: $S \rightarrow aABe, \quad A \rightarrow Abc \mid b, \quad B \rightarrow d$

The right most derivation for **abbcde** is as follow:

$$S \underset{rm}{\Rightarrow} aA\underline{B}e \underset{rm}{\Rightarrow} a\underline{A}de \underset{rm}{\Rightarrow} a\underline{A}bcde \underset{rm}{\Rightarrow} abbcde$$

Exercise 1:

Consider the following grammar G:

$$S \rightarrow XY$$
$$X \rightarrow aX \mid bX \mid a$$
$$Y \rightarrow Y\,a \mid Y\,b \mid a$$

(a) Give a leftmost derivation of abaabb.

(b) Build the derivation tree for the derivation in part (1).

(c) What is L(G)?

4. Right- or Left-Linear Grammar:

Linear Grammar: A grammar in which each production contains at most one nonterminal in its right-hand side of any production.

Right-linear grammar (Definition): G = (V, T, S, P) is said to be right-linear if all productions are of the form: $A \rightarrow xB$, $A \rightarrow x$, where A, B∈V and x ∈ T*.

Left-linear grammar (Definition): G = (V, T, S, P) is said to be left-linear if all productions are of the form: $A \rightarrow Bx$, $A \rightarrow x$, where A, B ∈ V and x ∈ T*.

Example 1:

Find L(G) where G = ({S, S1, S2}, {a, b}, S, P) with

$S \rightarrow S1ab$,

$S1 \rightarrow S1ab \mid S2$,

$S2 \rightarrow a$.

Answer: This is a left-linear grammar.

$S \Rightarrow S1ab \Rightarrow S1abab \Rightarrow S2abab \Rightarrow aabab$.

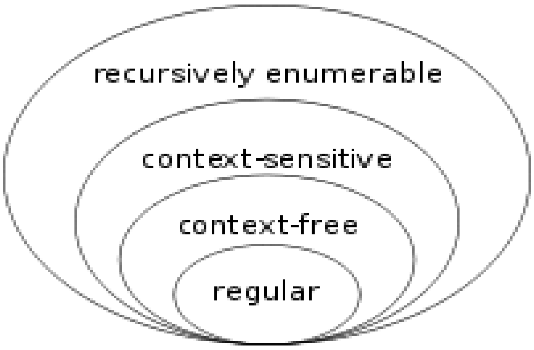
Then

L(G) = {aabw |w ∈ (ab)*}.

5. Hierarchy of Grammars (Chomsky Hierarchy):

The Chomsky hierarchy classifies grammars according to syntactic restrictions on rules as following. Let G = (∑, V, S, P) be a grammar.

1. G is called a **Type-0** grammar or an **unrestricted** grammar.
2. G is called a **Type-1** or **context-sensitive** grammar.
3. G is called a **Type-2** or **context-free** grammar.
4. G is called a **Type-3** or **regular** grammar.



1 An Unrestricted Grammar:

A set of production rules of the form $\alpha \to \beta$ where α and β are arbitrary strings of terminal and non-terminal symbols. The rules of these grammars do not have the restriction above, their left-hand sides may contain any string of terminal and /or non-terminal symbols, provided there is at least one non-terminal symbol.

Example 2:

L = {w ∈ {a, b, c}$^+$: number of a's, b's and c's is the same}

S → ABCS
S → ABC
AB → BA
BC → CB
AC → CA
BA → AB
CA → AC
CB → BC
A → a
B → b
C → c

Exercise 1: what is the language of the following grammar?

S → aBSc
S → aBc
Ba → aB
Bc → bc
Bb → bb

Exercise 2: Let G be the grammar <N, ∑, P, S>, where N = {S}, ∑ = {a, b}, and P are S→ϵ , S →aSbS.

a. Find all the strings that are directly derivable from SaS in G.
b. Find all the derivations in G that start at S and end at ab.
c. Find all the sentential forms (sequences) of G of length 4 at most.

Exercise 3: Find all the derivations of length 3 at most that start at S in the grammar <N, ∑, P, S> whose production rules are:

$S \rightarrow AS$
$aS \rightarrow bb$
$A \rightarrow aa$

2 A Context-Sensitive Grammar (CSG):

A production rules of the grammar have the form $\alpha \rightarrow \beta$ and $|\beta| \geq |\alpha|$, i.e. no production rule is length-decreasing.

A language L is context-sensitive if it is generated by some context-sensitive grammar.

Context-Sensitive grammars may have more than one symbol on the left-hand-side of their grammar rules, provided that at least one of them is a non-terminal and the number of symbols on the left-hand-side does not exceed the number of symbols on the right-hand-side.

Example3: The following grammar is context-sensitive (CSG).

$S \rightarrow aBCT|aBC$
$T \rightarrow ABCT|ABC$
$BA \rightarrow AB$
$CA \rightarrow AC$
$CB \rightarrow BC$
$aA \rightarrow aa$,
$aB \rightarrow ab$
$bB \rightarrow bb$,
$bC \rightarrow bc$
$cC \rightarrow cc$

Example 4: The following grammar is context-sensitive.

$S \rightarrow aTb \mid ab$
$aT \rightarrow aaTb \mid ac$.

What is the language of the grammar?

$\{ab\} \cup \{a^{n+1}cb^{n+1} \mid n \geq 0\}$. This language is context-free, it has the grammar

$S \rightarrow aTb \mid ab$, and $T \rightarrow aTb \mid c$. Any context-free language is context sensitive.

3 A Context-Free Grammar (CFG):

A production rules of the grammar have the form $\alpha \rightarrow \beta$, each production in P satisfies:

$|\alpha|=1$; i.e., α is a single nonterminal.

A language generated from a context-free grammar is called a context-free language. Any context-free language is context sensitive.

The grammars are called context free because – since all rules only have a nonterminal on the left-hand side – one can always replace that nonterminal symbol with what is on the right-hand side of the rule.

Example 5: $\{a^n b^n c^n \mid n \geq 0\}$ is context-sensitive but not context-free.

Here is a **CSG**.

$S \rightarrow \varepsilon \mid abc \mid aTBc$
$T \rightarrow abC \mid aTBC$
$CB \rightarrow BC$
$B \rightarrow b.$
$C \rightarrow c.$

Derive aaabbbccc.

$S \Rightarrow aTBc \Rightarrow aaTBCBc \Rightarrow aaabCBCBc \Rightarrow aaabBCCBc \Rightarrow aaabbCCBc \Rightarrow aaabbCBCc \Rightarrow aaabbBCCc \Rightarrow aaabbbCCc \Rightarrow aaabbbCcc \Rightarrow aaabbbccc.$

Example 6:

Let $L(G1) = \{0^n 1^n \mid n \geq 0\}$ and $L(G2) = \{0^n \# 1^n \mid n \geq 0\}$. Given two CFLs, it is easy to construct a CFG for their **union**, e.g., combining CFGs for L(G1) and L(G2):

$S \rightarrow S_1 \mid S_2$
$S_1 \rightarrow 0S_1 1 \mid \epsilon$
$S_2 \rightarrow 0S_2 1 \mid \#$

Example 7:

$S \rightarrow abS$

$S \rightarrow a$

$L(G)=(ab)^*a$

4 Regular Grammar:

G is a *Type-3* or *right-linear* or *regular grammar* if each production has one of the following three forms: $A \rightarrow cB$, $A \rightarrow c$, $A \rightarrow \epsilon$; where A, B are non-terminals (with B = A allowed) and c is a terminal.

The **regular languages** are subset of the context-free languages.

Such a grammar restricts its rules to a single nonterminal on the left-hand side. The a right-hand side consisting of a single terminal, possibly followed (or preceded, but not both in the same grammar) by a single nonterminal.

Regular languages can be considered as special types of **context free languages**, i.e. all regular languages are CF languages but not all CF languages are regular.

Example 8:

The following grammar is unrestricted.

$S \rightarrow TbC$

$Tb \rightarrow c$

$cC \rightarrow Sc \mid \varepsilon$

This grammar is not context-sensitive, not context-free, and not regular. But can transform it into $S \rightarrow Sc \mid \varepsilon$. So, the language of the grammar is regular.

Regular grammar generates regular languages as in following examples:

Example 9:

$S \rightarrow Aab$

$A \rightarrow Aab \mid B$

$B \rightarrow a$

$L(G)=aab(ab)^*$

Example 10:

The CFG ({S}, {a, b}, S, P) with P consisting of the following productions:

$S \rightarrow aSb$

$S \rightarrow \epsilon$

The grammar is not regular because of the ***b*** on the right of S.

It generates the language $a^n b^n$ where $n \geq 0$. This is not a regular language but it can be generated by a context free grammar is therefore a context free language.

Exercise 1:

$G = (\{S\}, \{0, 1\}, \{S \rightarrow 0S1 \mid \epsilon\}, S)$

- Is ϵ in L(G)?
- Is 01 in L(G)?
- Is 0011 in L(G)?
- Is $0^n 1^n$ in L(G)?

What language is defined by the following G?

$S \rightarrow \epsilon$

$S \rightarrow 0S1$

What language is defined by the following G?

$S \rightarrow \epsilon$

$S \rightarrow 0S0$

$S \rightarrow 1S1$

Exercise 2:

What is language generated by this grammar G given by the productions

$S \rightarrow 0S0 \mid 0B0$

$B \rightarrow 1B \mid 1$