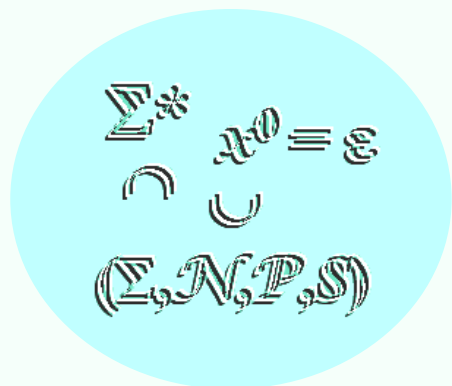
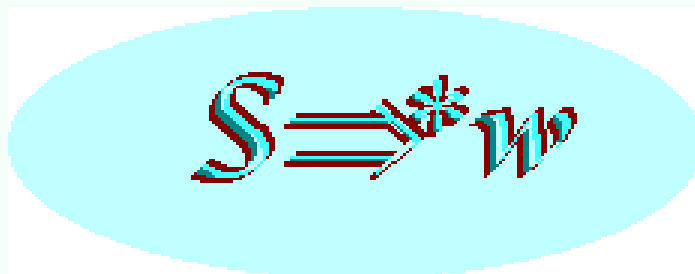


$S \Rightarrow^* w$

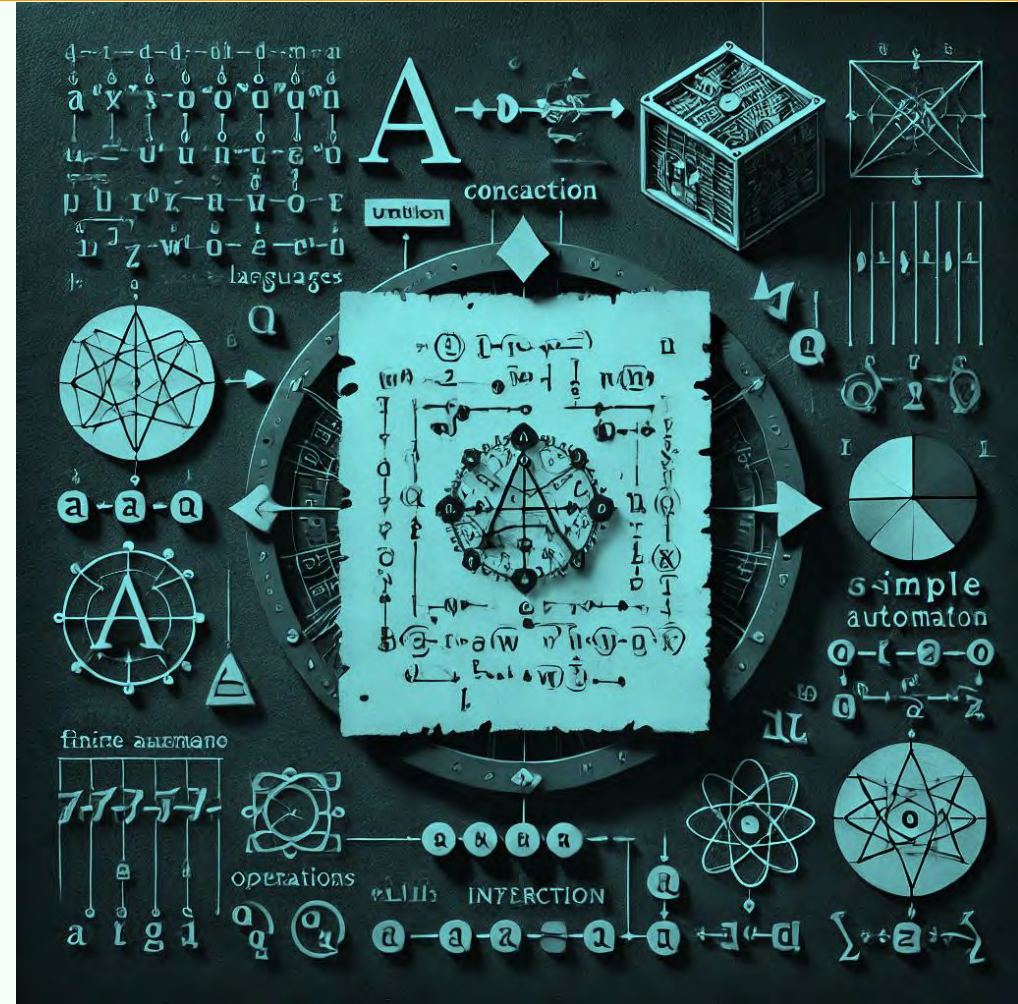
Alfabetos,



lenguajes



solo 7 primeras paginas



y gramáticas

Alfabetos y cadenas (1)

- **Alfabeto:** Un alfabeto Σ es un conjunto *finito y no vacío* de símbolos.
- **Cadena sobre un alfabeto Σ :** Es una sucesión de símbolos tomados de Σ .
- **Cadena vacía:** Cadena sin elementos, se representa por ε
- **Longitud de una cadena c :** Es el número de caracteres que forman c , se denota por $|c|$ o por $long(c)$
$$|\varepsilon| = long(\varepsilon) = 0$$
- **Concatenación de cadenas:** La concatenación de la cadena x con la cadena y es la cadena que resulta al añadir a la cadena x la cadena y , se denota por xy .

Alfabetos y cadenas (2)

- Propiedades de la concatenación:
 1. Es una operación cerrada
 2. Es asociativa
 3. La cadena vacía es el elemento identidad
 4. La concatenación no es conmutativa
 5. Verifica las leyes de cancelación izquierda y derecha
- Para representar la concatenación repetida se usa una notación de tipo exponencial

$$x^n = \underbrace{xx \cdots x}_{n \text{ veces}} \quad x^{n+1} = xx^n = x^n x \quad x^n x^m = x^{n+m} \quad x^1 = x \quad x^0 = \varepsilon$$

Lenguajes

- **Lenguaje universal sobre Σ :** Conjunto de todas las posibles cadenas formadas sobre Σ (incluida ε), se denota por Σ^* .
- **Lenguaje formal sobre un alfabeto Σ :** Conjunto de cadenas sobre Σ , cualquier subconjunto de Σ^* .
- **Lenguaje vacío:** Es aquel lenguaje que no contiene ninguna cadena, ni siquiera la vacía. Se representa por Φ .
- **Ejemplos:** Dado el alfabeto $\Sigma = \{a, b\}$
 - lenguaje de las cadenas con exactamente una b , $L_1 = \{b, ab, a^3ba^2, \dots\}$
 - lenguaje de las cadenas de longitud menor o igual que uno, $L_2 = \{\varepsilon, a, b\}$
 - lenguaje vacío, $L_3 = \Phi$
 - lenguaje que solo tiene la cadena vacía, $L_4 = \{\varepsilon\}$

Operaciones sobre lenguajes (1)

- **Unión de lenguajes:** La unión $L_1 \cup L_2$ de dos lenguajes L_1 y L_2 es el lenguaje formado por las cadenas que pertenecen bien a L_1 bien a L_2 . La unión de lenguajes constituye un *monoide*. <https://es.wikipedia.org/wiki/Monoide>

$$L_1 \cup L_2 = \{x: x \in L_1 \vee x \in L_2\}$$

- **Intersección de lenguajes:** La intersección $L_1 \cap L_2$ de dos lenguajes L_1 y L_2 es el lenguaje formado por las cadenas que pertenecen simultáneamente a L_1 y a L_2 .

$$L_1 \cap L_2 = \{x: x \in L_1 \wedge x \in L_2\}$$

- **Diferencia de lenguajes:** La diferencia $L_1 - L_2$ de dos lenguajes L_1 y L_2 es el lenguaje formado por las cadenas que pertenecen a L_1 y no pertenece a L_2 .

$$L_1 - L_2 = \{x: x \in L_1 \wedge x \notin L_2\}$$

Operaciones sobre lenguajes (2)

- **Complementación de un lenguaje:** La complementación \bar{L} de un lenguaje L sobre un alfabeto Σ es el lenguaje formado por las cadenas que pertenecen al lenguaje universal sobre Σ (Σ^*) pero que no pertenecen a L . Dicho de otro modo, está formado por las cadenas que pertenecen a $\Sigma^* - L$.

$$\bar{L} = \{x: x \in \Sigma^* \wedge x \notin L\} = \{x: x \in \Sigma^* - L\}$$

- **Concatenación:** La concatenación $L_1 \cdot L_2$ de dos lenguajes L_1 y L_2 es el lenguaje formado por cadenas construidas mediante la concatenación de una cadena de L_1 con una cadena de L_2 . La concatenación de lenguajes constituye un *monoide*.

$$L_1 \cdot L_2 = \{xy: x \in L_1 \wedge y \in L_2\}$$

También se define la **exponenciación** como simplificación notacional de la concatenación, con las siguientes propiedades:

- $L^n = L \cdot L^{n-1} = L^{n-1} \cdot L$
- $L^n \cdot L^m = L^{n+m}$
- $L^0 = \{\varepsilon\}$

Operaciones sobre lenguajes (3)

- **Iteración, cierre o clausura de un lenguaje:** El cierre de un lenguaje L (o *cerradura de Kleene*) se representa por L^* y se define como:

$$L^* = \{\varepsilon\} \cup L \cup L^2 \cup L^3 \cup \dots = \bigcup_{i=0}^{\infty} L^i$$

- **Clausura positiva:** Se representa por L^+ y se define como:

$$L^+ = L \cup L^2 \cup L^3 \cup \dots = \bigcup_{i=1}^{\infty} L^i$$

- **Ejemplos:** Si $L = \{a, b, \dots, z, A, B, \dots, Z\}$ y $D = \{0, 1, \dots, 9\}$, entonces:

$L \cdot D$ es el lenguaje $\{a0, a1, \dots, a9, b0, \dots, b9, \dots, Z8, Z9\}$

$L \cdot (L \cup D)^*$ es el lenguaje de las palabras que comienzan por una letra y pueden tener tanto letras como dígitos, es decir, el formato típico de los identificadores de los lenguajes de programación.

- ¿lenguaje de las palabras de longitud par?
- ¿lenguaje de las palabras de longitud impar?
- ¿lenguaje de los números decimales múltiplos de cinco?
- ¿lenguaje de los números binarios múltiplos de tres?
- ¿lenguaje de los números decimales en punto flotante?

Gramáticas (1)

- El estudio de los lenguajes se puede hacer desde tres puntos de vista:
 1. el de la **interpretación**: tiene que ver con la semántica de los lenguajes intenta formalizar el significado de las sentencias de un lenguaje.
 2. la **generación** de lenguajes: consiste en encontrar mecanismos que permitan enumerar las cadenas que pertenecen a un lenguaje. Estos mecanismos son las *gramáticas*.
 3. el del **reconocimiento** del lenguaje: está muy ligado a la teoría de autómatas y es « ...*el estudio de algoritmos o estructuras de máquinas que permiten, dado un lenguaje L y una cadena x , determinar si $x \in L$ o $x \notin L$* ».

Gramáticas (2)

- **Gramática formal:** Es una cuádrupla $(\Sigma, \mathcal{N}, \mathcal{P}, S)$, donde:
 - Σ es un alfabeto (*conjunto finito y no vacío*) de **símbolos terminales** (o **tokens/tókenes**).
 - \mathcal{N} es un alfabeto cuyos elementos se llaman **símbolos no terminales**
 - \mathcal{P} es un alfabeto de **producciones** de la forma:
$$\{u \rightarrow v: u = xAy \in (\Sigma \cup \mathcal{N})^+ \wedge A \in \mathcal{N} \wedge v \in (\Sigma \cup \mathcal{N})^*\}$$
 - $S \in \mathcal{N}$ es un símbolo especial llamado **axioma** o **símbolo inicial**.

$S \Rightarrow^* w$

$$G_i = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{\underline{S}, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

Ejemplo de gramática

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{\underline{S}, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

$$\underline{S} \xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a a \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b$$

Ejemplo de derivación
más a la izquierda

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

$$\begin{aligned} \underline{S} &\xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a a \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b \\ &\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \end{aligned}$$

Ejemplo de derivación
alternativa

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{\underline{S}, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

$$\begin{aligned} \underline{S} &\xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a a \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b \\ &\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \\ &\xRightarrow{6} a A b \xRightarrow{3} a A \underline{B} c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \end{aligned}$$

Ejemplo de derivación
más a la derecha

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{\underline{S}, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

Left most derivation
lmd

$$\underline{S} \xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a a \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b$$

$$\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b$$

$$\xRightarrow{6} a A b \xRightarrow{3} a A \underline{B} c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b$$

$S \Rightarrow^* aacb$
derivación más
a la izda

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{\underline{S}, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

Left most derivation
lmd

$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más
a la izda

$$\begin{aligned} \underline{S} &\xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a \underline{a} \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b \\ &\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \\ &\xRightarrow{6} a A b \xRightarrow{3} a A \underline{B} c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \end{aligned}$$

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2 \ 3 \ 4 \\ B \rightarrow aA / \epsilon & 5 \ 6 \end{cases}$$

Left most derivation
l.m.d

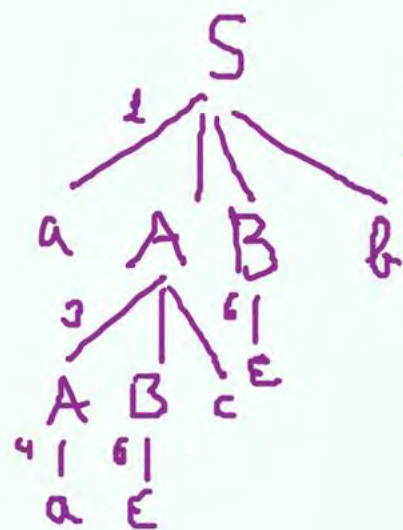
$$S \Rightarrow_1^* aacb$$

$$S \Rightarrow_2^* aacb$$

derivación más
a la izda

$$\underline{S} \xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a a \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b$$

$$\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b$$



$$\xRightarrow{6} a A b \xRightarrow{3} a A B c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b$$

Ejemplo de árbol de derivación,
árbol de análisis sintáctico
o árbol de sintaxis abstracta

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb \\ A \rightarrow bB / ABc / a \\ B \rightarrow aA / \epsilon \end{cases}$$

1

2 3 4

5 6

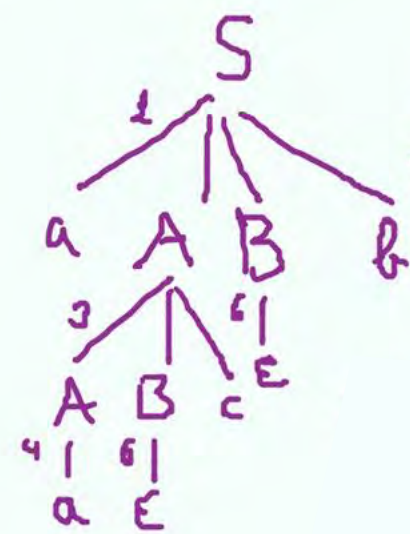
$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más a la izda

Left most derivation
l.m.d

$$\begin{aligned} S &\xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c B b \xRightarrow{4} a \underline{a} \underline{B} c B b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b \\ &\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \end{aligned}$$



$$\begin{aligned} &\xRightarrow{6} a A c b \xRightarrow{3} a A B c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \\ &\xRightarrow{6} a A c b \xRightarrow{3} a A B c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \end{aligned}$$

$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\} \quad G_1 \equiv G_2 \Leftrightarrow L(G_1) = L(G_2)$$

Definición de lenguaje generado por una gramática

Definición de equivalencia de gramáticas

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2, 3, 4 \\ B \rightarrow aA / \epsilon & 5, 6 \end{cases}$$

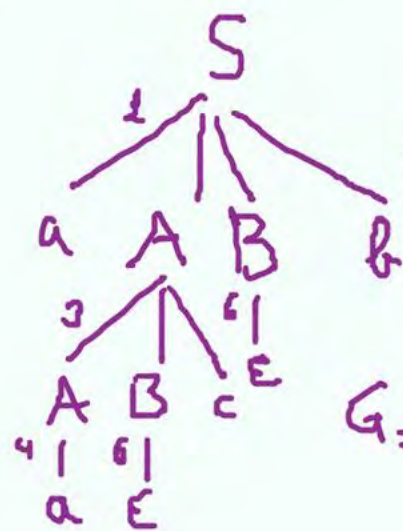
Left most derivation
l.m.d

$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más
a la izda

$$\begin{aligned} S &\xRightarrow{1} aABb \xRightarrow{3} aABcBb \xRightarrow{4} aaBcBb \xRightarrow{6} aacBb \xRightarrow{6} aacb \\ &\xRightarrow{6} aAcBb \xRightarrow{6} aAcbb \xRightarrow{4} aacb \end{aligned}$$



$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\}$$

$$G_1 \equiv G_2 \Leftrightarrow L(G_1) = L(G_2)$$

$$G_2 = (\Sigma, N_2, P_2, S)$$

$$P_2 = \begin{cases} S \rightarrow aCaCb & 1 \\ S \rightarrow aAb & 2 \\ C \rightarrow b / A & 3, 4 \\ B \rightarrow bB & 5 \\ A \rightarrow ABc / a & 6, 7 \end{cases}$$

Producciones de una gramática
equivalente a la que se ha venido
usando

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2, 3, 4 \\ B \rightarrow aA / \epsilon & 5, 6 \end{cases}$$

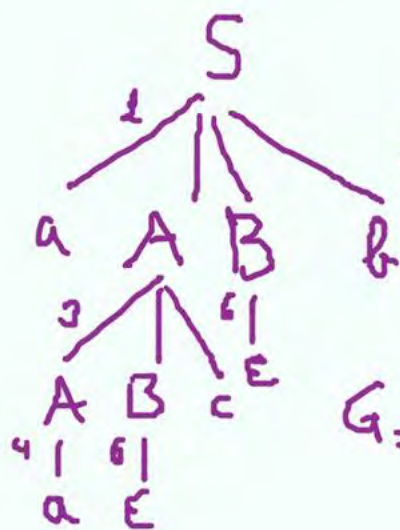
Left most derivation
l.m.d

$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más
a la izda

$$\begin{aligned} S &\xRightarrow{1} aABb \xRightarrow{3} aABcBb \xRightarrow{4} aaBcBb \xRightarrow{6} aacBb \xRightarrow{6} aacb \\ &\xRightarrow{6} aAcBb \xRightarrow{6} aAc b \xRightarrow{4} aacb \end{aligned}$$



$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\}$$

$$G_2 = (\Sigma, N_2, P_2, S)$$

$$P_2 = \begin{cases} S \rightarrow aCaCb & 1 \\ S \rightarrow aAb & 2 \\ C \rightarrow B / A & 3, 4 \\ B \rightarrow bB & 5 \\ A \rightarrow ABc / a & 6, 7 \end{cases}$$

Algunos tipos de producciones especiales

$$B \rightarrow \epsilon \text{ ÉPSILON}$$

Producción épsilon

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2, 3, 4 \\ B \rightarrow aA / \epsilon & 5, 6 \end{cases}$$

1

2 3 4
5 6

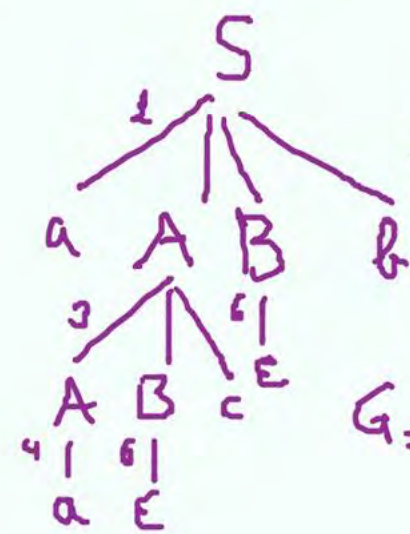
$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más a la izda

Left most derivation
l.m.d

$$\begin{aligned} S &\xRightarrow{1} a \underline{A} \underline{B} b \xRightarrow{3} a \underline{A} \underline{B} c b \xRightarrow{4} a \underline{a} \underline{B} c b \xRightarrow{6} a a c \underline{B} b \xRightarrow{6} a a c b \\ &\xRightarrow{6} a A c \underline{B} b \xRightarrow{6} a A c b \xRightarrow{4} a a c b \end{aligned}$$



$$\xRightarrow{6} a A c b \xRightarrow{3} a A B c b \xRightarrow{6} a A c b \xRightarrow{4} a a c b$$

$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\}$$

$$G_1 \equiv G_2 \Leftrightarrow L(G_1) = L(G_2)$$

Algunos tipos de producciones especiales

$$G_2 = (\Sigma, N_2, P_2, S)$$

$$P_2 = \begin{cases} S \rightarrow aCaCb & 1 \\ S \rightarrow aAb & 2 \\ C \rightarrow B / A & 3, 4 \\ B \rightarrow bB & 5 \\ A \rightarrow ABc / a & 6, 7 \end{cases}$$

$$B \rightarrow \epsilon \text{ ÉPSILON}$$

$$C \rightarrow B \text{ NO GENERATIVAS}$$

Producción no generativa

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB \mid ABc \mid a & 2, 3, 4 \\ B \rightarrow aA \mid \epsilon & 5, 6 \end{cases}$$

1

2 3 4

5 6

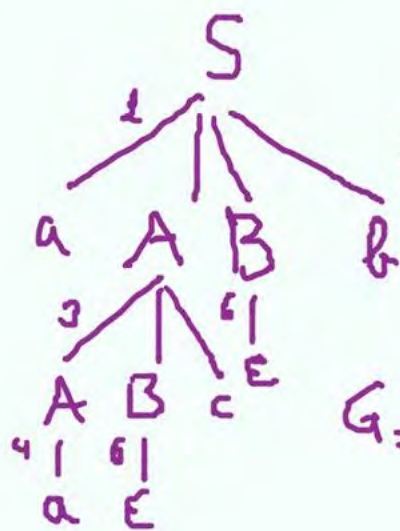
$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más a la izda

Left most derivation
l.m.d

$$\begin{aligned} S &\xRightarrow{1} aABb \xRightarrow{3} aABcBb \xRightarrow{4} aaBcBb \xRightarrow{6} aacBb \xRightarrow{6} aacb \\ &\xRightarrow{6} aAcBb \xRightarrow{6} aAcbb \xRightarrow{4} aacb \end{aligned}$$



$$\xRightarrow{6} aAcbb \xRightarrow{3} aABcbb \xRightarrow{6} aAcbb \xRightarrow{4} aacb$$

$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\}$$

$$G_1 \equiv G_2 \Leftrightarrow L(G_1) = L(G_2)$$

Algunos tipos de producciones especiales

$$G_2 = (\Sigma, N_2, P_2, S)$$

$$P_2 = \begin{cases} S \rightarrow aCaCb & 1 \\ S \rightarrow aAb & 2 \\ C \rightarrow B \mid A & 3, 4 \\ B \rightarrow bB & 5 \\ A \rightarrow ABc \mid a & 6, 7 \end{cases}$$

$$B \rightarrow \epsilon \text{ ÉPSILON}$$

$$C \rightarrow B \text{ NO GENERATIVAS}$$

$$A \rightarrow ABc$$

RECURSIVA A IZDA
Producción recursiva a izdas

$$G_1 = (\Sigma, N, P_1, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2, 3, 4 \\ B \rightarrow aA / \epsilon & 5, 6 \end{cases}$$

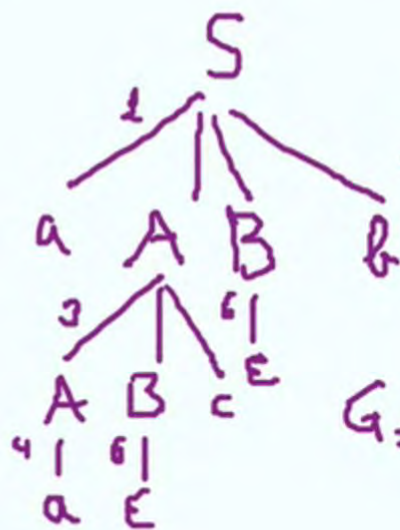
Left most derivation
lmd

$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más
a la izda

$$\begin{aligned} S &\xRightarrow{1} aABb \xRightarrow{3} aABcBb \xRightarrow{4} aaBcBb \xRightarrow{6} aacBb \xRightarrow{6} aacb \\ &\xRightarrow{6} aAcBb \xRightarrow{6} aAcbb \xRightarrow{4} aacb \end{aligned}$$



$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\}$$

$$G_1 \equiv G_2 \Leftrightarrow L(G_1) = L(G_2)$$

Algunos tipos de producciones especiales

$$G_2 = (\Sigma, N_2, P_2, S)$$

$$P_2 = \begin{cases} S \rightarrow aCaCb & 1 \\ S \rightarrow aAb & 2 \\ C \rightarrow B / A & 3, 4 \\ B \rightarrow bB & 5 \\ A \rightarrow ABc / a & 6, 7 \end{cases}$$

$$B \rightarrow \epsilon \text{ ÉPSILON}$$

$$C \rightarrow B \text{ NO GENERATIVAS}$$

$$A \rightarrow ABc$$

RECURSIVA A IZDA

$$A \rightarrow cBcA$$

Producción recursiva a derchas

$S \Rightarrow^* w$

$$G_1 = (\Sigma, N, P_1, S)$$

$$\Sigma = \{a, b, c\}$$

$$N = \{S, A, B\}$$

$$P_1 = \begin{cases} S \rightarrow aABb & 1 \\ A \rightarrow bB / ABc / a & 2, 3, 4 \\ B \rightarrow aA / \epsilon & 5, 6 \end{cases}$$

1

2 3 4
5 6

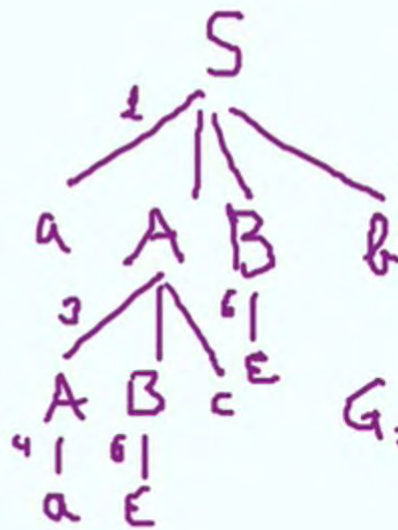
$$S \Rightarrow^* aacb$$

$$S \Rightarrow^* aacb$$

derivación más a la izda

Left most derivation
lmd

$$\underline{S} \xrightarrow{1} a \underline{A} \underline{B} b \xrightarrow{3} a \underline{A} \underline{B} c B b \xrightarrow{4} a \underline{a} \underline{B} c B b \xrightarrow{6} a a c \underline{B} b \xrightarrow{6} a a c b \xrightarrow{6} a A c b \xrightarrow{4} a a c b$$



$$\xrightarrow{6} a A c b \xrightarrow{3} a A B c b \xrightarrow{6} a A c b \xrightarrow{4} a a c b$$

$$L(G_1) = \{w \in \Sigma^* : S \Rightarrow^* w\} \quad G_1 \equiv G_2 \Leftrightarrow L(G_1) = L(G_2)$$

$$G_2 = (\Sigma, N_2, P_2, S)$$

Recursividad en varios pasos

$$A \rightarrow cBcA$$

$$\begin{aligned} A &\rightarrow BCa & 1 \\ B &\rightarrow \epsilon & 2 \\ C &\rightarrow BAB & 3 \end{aligned}$$

$$P_2 = \begin{cases} S \rightarrow aCaCb & 1 \\ S \rightarrow aAb & 2 \\ C \rightarrow B / A & 3, 4 \\ B \rightarrow bB & 5 \\ A \rightarrow ABc / a & 6, 7 \end{cases}$$

$$B \rightarrow \epsilon \text{ ÉPSILON}$$

$$C \rightarrow B \text{ NO GENERATIVAS}$$

$$A \rightarrow ABc$$

RECURSIVA A LA DCHA

Gramáticas (3)

- **Producción:** Una producción (o **regla de reescritura**) es un par ordenado de cadenas sobre un alfabeto $((x,y): x, y \in \Sigma^*)$. Se representa por $x \rightarrow y$. x es la parte izquierda o **antecedente** de la producción e y es la parte derecha o **consecuente**.
- **Derivación directa:** Sea $x \rightarrow y$ una producción y $v, w \in \Sigma^*$. Se dice que w *deriva directamente* de v , y se escribe $v \Rightarrow w$ si y sólo si existen $z, u \in \Sigma^*$ tales que $v = zxu$, $w = zyu$ y $x \rightarrow y$.
- **Derivación (en uno o más pasos):** w deriva de v y se escribe $v \Rightarrow^+ w$ cuando existen $u_0, u_1, \dots, u_n \in \Sigma^*$ tales que:

$$\begin{aligned} v &= u_0 \\ u_0 &\Rightarrow u_1 \\ u_1 &\Rightarrow u_2 \\ &\vdots \\ u_{n-1} &\Rightarrow u_n \\ u_n &= w \end{aligned}$$

En el contexto de las gramáticas el alfabeto va a estar formado por la unión de los alfabetos de terminales y no terminales. Además, el antecedente de una producción nunca va a poder ser épsilon y ha de contener al menos un no terminal.

A la secuencia u_0, u_1, \dots, u_n se la llama **cadena de derivación** de longitud n .

Gramáticas (4)

- **Derivación más a la izquierda:** las producciones utilizadas en la derivación se aplican siempre a los símbolos más a la izquierda.
- **Derivación más a la derecha:** las producciones se aplican a los símbolos más a la derecha.
- **Forma sentencial** de una gramática: Dada una gramática $(\Sigma, \mathcal{N}, \mathcal{P}, S)$. Una cadena $\alpha \in (\Sigma \cup \mathcal{N})^*$ es una *forma sentencial* de esa gramática si existe una derivación que produce α a partir del axioma S , es decir si $S \Rightarrow^* \alpha$.
- **Frase o sentencia** de una gramática: Es una forma sentencial α que sólo contiene símbolos terminales ($\alpha \in \Sigma^*$).
- **Lenguaje generado por una gramática:** El *lenguaje generado por una gramática* G se representa por $L(G)$ y se define como el conjunto de todas las sentencias de la gramática G .

$$L(G) = \{x \in \Sigma^* : S \Rightarrow^+ x\}$$

Gramáticas (5)

- Dos gramáticas G_1 y G_2 se dice que son **gramáticas equivalentes**, y se representa por $G_1 \equiv G_2$, si generan el mismo lenguaje $L(G_1) = L(G_2)$.
- Una gramática G se dice que es una **gramática recursiva en un cierto símbolo no terminal** U , cuando existe una forma sentencial de U que contiene a U .

$$U \Rightarrow^+ xUy \text{ con } x, y \in (\Sigma \cup \mathcal{N})^*$$

- La **gramática** será **recursiva** si es recursiva para algún no terminal.
 - Si $x = \varepsilon$ se dice que es una **gramática recursiva a izquierdas**.
 - Si $y = \varepsilon$ se dice que es una **gramática recursiva a derechas**.

Árbol de derivación

- Son una forma de representar las derivaciones de algunas gramáticas (las independientes del contexto).
- Se utilizan para representar el análisis sintáctico de los programas fuente, sirviendo de base para la generación de código.
- En los árboles de derivación (o de análisis sintáctico o árboles de sintaxis concreta):
 - el axioma se representa en la raíz del árbol;
 - los nodos hojas del árbol son símbolos terminales de la gramática; y
 - las derivaciones se representan creando tantos sucesores del símbolo no terminal de la izquierda de la producción como símbolos (terminales y no terminales) aparezcan en la parte derecha de las producciones.

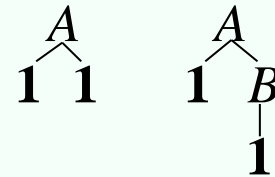
Ambigüedad

La ambigüedad puede surgir a varios niveles: en sentencias, lenguajes, y gramáticas. A la hora de utilizar eficientemente los lenguajes y gramáticas, es conveniente que no exista ambigüedad, pues provoca que el análisis de las sentencias no sea determinista.

- **Sentencia:** una sentencia es ambigua si tiene más de árbol de derivación.

Ej: En el caso de la gramática $G = (\{1\}, \{A, B\}, \{A \rightarrow 1B, A \rightarrow 11, B \rightarrow 1\}, A)$

la sentencia **11** puede ser obtenida con dos árboles:



- **Gramática:** una gramática es ambigua si tiene al menos una sentencia ambigua.
- **Lenguaje:** un lenguaje es ambiguo si existe una gramática ambigua que lo genera. Si todas las gramáticas que generan un lenguaje son ambiguas, el lenguaje es **inherentemente ambiguo** ($L = \{a^i b^j c^k : i=j \text{ o } j=k\}$).

Notación gramatical. Notación BNF.

- **Notación BNF:** es una forma de escribir las producciones. Las reglas son:

1. Los símbolos terminales se suelen poner en negrita, y los no terminales entre ángulos < >
2. El nexos para la producción es ::=
3. La agrupación de reglas mediante |
4. Cero o una vez mediante corchetes []
5. Cero o más veces mediante llaves { } $\{ \}_n^{n+k}$
6. Paréntesis para delimitar casos de duda en el uso de los operadores meta-lingüísticos

https://en.wikipedia.org/wiki/Backus%E2%80%93Naur_form

Una gramática de un subconjunto del castellano en notación BNF

$\langle \text{Oración} \rangle ::= \langle \text{Fnominal} \rangle \langle \text{Fverbal} \rangle$
| $\langle \text{Fnominal} \rangle \langle \text{Fverbal} \rangle \langle \text{Complemento} \rangle$

$\langle \text{Fnominal} \rangle ::= \langle \text{Sustantivo} \rangle$ | $\langle \text{PrNom} \rangle$ | $\langle \text{Artículo} \rangle \langle \text{Sustantivo} \rangle$
| $\langle \text{Artículo} \rangle \langle \text{Sustantivo} \rangle \langle \text{Adjetivo} \rangle$
| $\langle \text{Fnominal} \rangle$ **de** $\langle \text{Fnominal} \rangle$

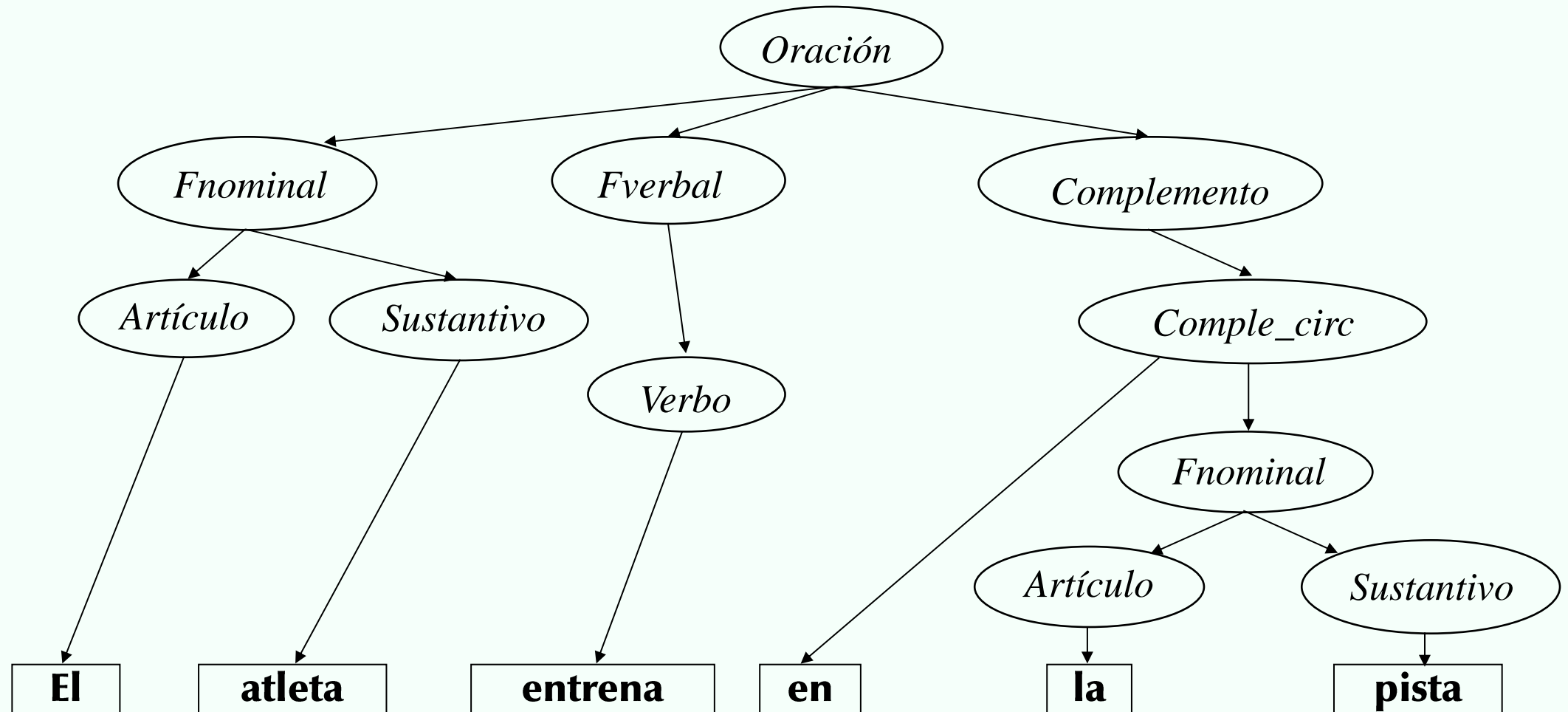
$\langle \text{Fverbal} \rangle ::= \langle \text{Verb} \rangle$ | $\langle \text{Verb} \rangle \langle \text{Adverbio} \rangle$

$\langle \text{Complemento} \rangle ::= \langle \text{ComDir} \rangle$ | $\langle \text{ComIn} \rangle$ | $\langle \text{ComCir} \rangle$ | $\langle \text{ComDir} \rangle$

$\langle \text{ComDir} \rangle ::= \langle \text{Fnominal} \rangle$

$\langle \text{ComIn} \rangle ::=$ **a** $\langle \text{Fnominal} \rangle$ | **para** $\langle \text{Fnominal} \rangle$ |

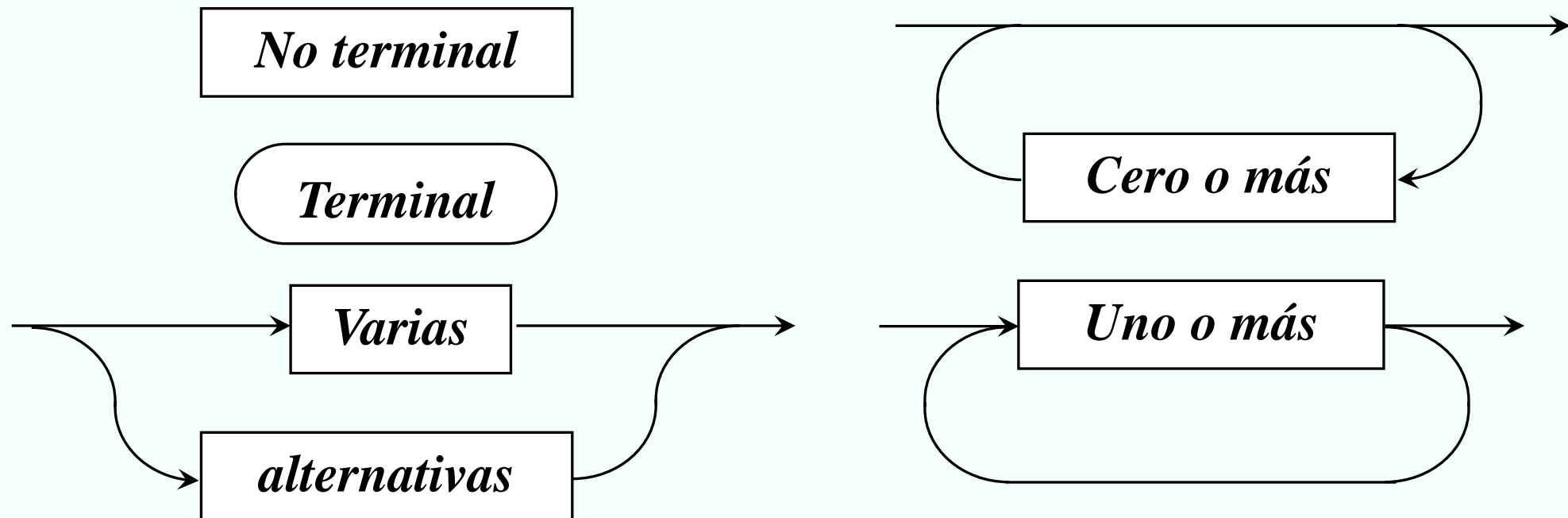
$\langle \text{ComCir} \rangle ::=$ **en** $\langle \text{Fnominal} \rangle$ | **desde** $\langle \text{Fnominal} \rangle$
| **cuando** $\langle \text{Fnominal} \rangle$ |



Notación gramatical. Diagramas sintácticos

- Los **diagramas sintácticos** (**diagramas de trenes**, o **diagramas de Conway**), son construcciones gráficas que permiten presentar la misma información sintáctica que la notación BNF pero de una manera más fácil de interpretar para las personas.

https://en.wikipedia.org/wiki/Syntax_diagram



Clasificación de gramáticas (1)

- Gramáticas de tipo 0 o gramáticas de Chomsky, con reglas de producción de la forma:

$$u \rightarrow v \text{ con } u = xAy \in (\Sigma \cup \mathcal{N})^+ \wedge A \in \mathcal{N} \wedge x, y, v \in (\Sigma \cup \mathcal{N})^*$$

- El conjunto de lenguajes de tipo 0 coincide con el de todos los lenguajes gramaticales posibles.
- Puede demostrarse que todo lenguaje generado por una gramática de Chomsky puede generarse también por unas gramáticas más restrictivas llamadas **gramáticas con estructura de frase**, cuyas reglas de producción son de la forma:

$$xAy \rightarrow xvy \text{ con } x, y, v \in (\Sigma \cup \mathcal{N})^* \wedge A \in \mathcal{N}$$

Clasificación de gramáticas (2)

- Gramáticas de tipo 1 o sensibles al contexto, con reglas de producción de la forma:

$$xAy \rightarrow xvy \text{ con } x, y \in (\Sigma \cup \mathcal{N})^* \wedge A \in \mathcal{N} \wedge v \in (\Sigma \cup \mathcal{N})^+$$

- En los lenguajes generados por estas gramáticas el significado de las «palabras» depende de su posición en la frase.
- A la x e y se les llama **contexto** (es decir, A sólo puede transformarse en v si va precedido de x y al mismo tiempo seguido de y).
- No tiene *reglas compresoras*, aunque se tolera la regla $S \rightarrow \varepsilon$.
- Las gramáticas de mayor categoría que se suelen utilizar (la mayor parte de los lenguajes de ordenador pertenecen a este grupo, aunque gran parte de las reglas de las gramáticas que los generan pueden reducirse a las de tipo 2).
- La longitud de las formas sentenciales a partir de S es siempre no decreciente.

Clasificación de gramáticas (3)

- Gramáticas de tipo 2 o independientes del contexto, con reglas de la forma:

$$A \rightarrow v \text{ con } A \in \mathcal{N} \wedge v \in (\Sigma \cup \mathcal{N})^*$$

- Se vuelven a introducir leyes compresoras, pero es fácil demostrar que se puede obtener una gramática equivalente que no las tenga, obteniéndose una definición algo más restrictiva:

$$A \rightarrow v \text{ con } A \in \mathcal{N} \wedge v \in (\Sigma \cup \mathcal{N})^+$$

además, es posible que se tenga la regla $S \rightarrow \varepsilon$

- En los lenguajes generados por las gramáticas de este tipo el significado de las «palabras» es independiente de su posición.
- Una última característica de este tipo de gramáticas es que las derivaciones obtenidas al utilizarlas se pueden representar utilizando árboles.

https://en.wikipedia.org/wiki/Context-free_grammar

Clasificación de gramáticas (4)

- Gramáticas de tipo 3 o **gramáticas regulares**, con reglas de la forma:
 $A \rightarrow aB \wedge A \rightarrow b$ o de la forma $A \rightarrow Ba \wedge A \rightarrow b$
con $A, B \in \mathcal{N} \wedge a, b \in \Sigma$
- A las gramáticas regulares del primer tipo se las llama **gramáticas regulares a derechas**, a las del segundo tipo **gramáticas regulares a izquierdas**, en realidad son totalmente equivalentes. Si ε pertenece al lenguaje, se tolera la regla $S \rightarrow \varepsilon$.
- Existe una generalización de este tipo de gramáticas llamadas **gramáticas lineales** con reglas de la forma:
 $A \rightarrow wB \wedge A \rightarrow v$ o de la forma $A \rightarrow Bw \wedge A \rightarrow v$
con $A, B \in \mathcal{N} \wedge w, v \in \Sigma^*$
son totalmente equivalentes a las gramáticas regulares normales, pero en muchos casos su notación es más adecuada.

Máquinas y autómatas

○ Máquinas de Turing

- Dispositivo de cómputo universal consistente en una cinta de E/S de tamaño infinito y una cabeza de lectura y escritura capaz de realizar las siguientes operaciones: leer un símbolo, escribir un símbolo, cambiar de estados, moverse sobre la cinta a izquierda o a derecha.

○ Autómata linealmente acotado

- Máquina de Turing con una cinta de tamaño limitado.

https://en.wikipedia.org/wiki/Linear_bounded_automaton

○ Autómata de pila

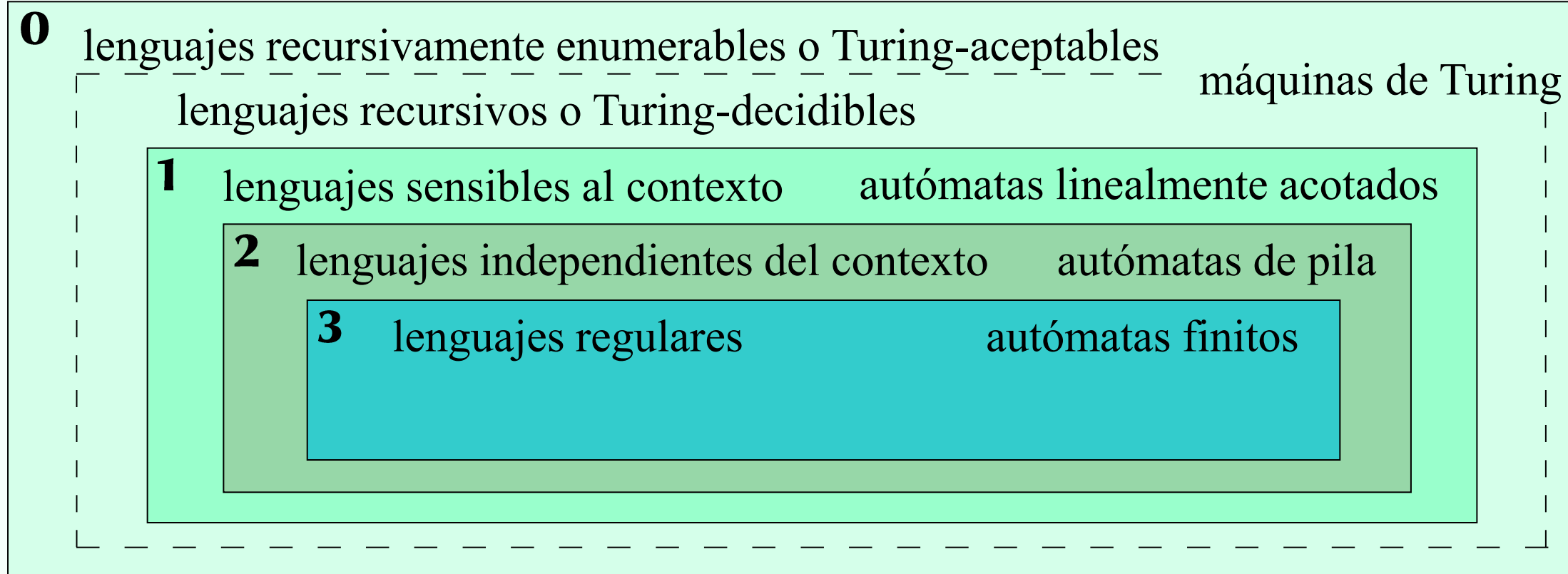
- Se puede ver como una MT que sólo puede leer la cinta en un sentido, aunque puede utilizar una pila para almacenar lo que ha leído.

○ Autómata finito

- Se puede ver como una MT que sólo puede leer la cinta en un sentido y no tiene ningún tipo de dispositivo de almacenamiento (salvo el propio estado interno en el que se encuentra).

Relación entre autómatas y lenguajes

https://en.wikipedia.org/wiki/Turing_machine



https://en.wikipedia.org/wiki/Chomsky_hierarchy