Introducción a la Lógica y la Computación

Mariana Badano Héctor Gramaglia
Pedro Sánchez Terraf Mauricio Tellechea
Guido Ivetta

FaMAF, 12 de noviembre de 2021

UNC Universidad Nacional de Córdoba 1613 - 2013 400 AÑOS

Contenidos estimados para hoy

1. Autómatas para lenguajes libres de contexto
   - Ejemplo de computación para L_{01}
2. Lenguajes contextuales
   - Lema de bombeo para libres de contexto
   - Ejemplo de lenguaje no libre de contexto
   - Gramáticas contextuales
3. Gramáticas irrestrictas
   - Computabilidad
   - Jerarquía de Chomsky

Los lenguajes libres de contexto son exactamente los aceptados por los **autómatas a pila** (“*push-down automata*”, **PDA**).

$$L_{01} = \{ 0^n 1^n : n \in \mathbb{N}_0 \}$$

requiere memoria no acotada.

Un vistazo a los PDA

Los lenguajes libres de contexto son exactamente los aceptados por los **autómatas a pila** ("*push-down automata*", **PDA**).

Los PDA tienen toda la estructura de un NFA pero además pueden guardar información en un pila (LIFO) sin límite de memoria.

Un vistazo a los PDA

Los lenguajes libres de contexto son exactamente los aceptados por los **autómatas a pila** (“*push-down automata*”, **PDA**).

Los PDA tienen toda la estructura de un NFA pero además pueden guardar información en un pila (LIFO) sin límite de memoria. O sea, tienen memoria (además de “un core”) pero no es RAM.

Un vistazo a los PDA

Los lenguajes libres de contexto son exactamente los aceptados por los **autómatas a pila** ("*push-down automata*", **PDA**).

Los PDA tienen toda la estructura de un NFA pero además pueden guardar información en un pila (LIFO) sin límite de memoria. O sea, tienen memoria (además de "un core") pero no es RAM.

A diferencia de los autómatas finitos, el rol de los PDA es comparativamente secundario (a nivel práctico y en cuanto a uso en demostraciones).

Un vistazo a los PDA

Los lenguajes libres de contexto son exactamente los aceptados por los **autómatas a pila** ("*push-down automata*", **PDA**).

Los PDA tienen toda la estructura de un NFA pero además pueden guardar información en un pila (LIFO) sin límite de memoria. O sea, tienen memoria (además de "un core") pero no es RAM.

A diferencia de los autómatas finitos, el rol de los PDA es comparativamente secundario (a nivel práctico y en cuanto a uso en demostraciones).

Consumir palabras

Los PDA dan un modelo de computación "aceptador" (*acceptor*), que va consumiendo una palabra y finalmente la acepta o no.

Un vistazo a los PDA

Los lenguajes libres de contexto son exactamente los aceptados por los **autómatas a pila** ("*push-down automata*", **PDA**).

Los PDA tienen toda la estructura de un NFA pero además pueden guardar información en un pila (LIFO) sin límite de memoria. O sea, tienen memoria (además de "un core") pero no es RAM.

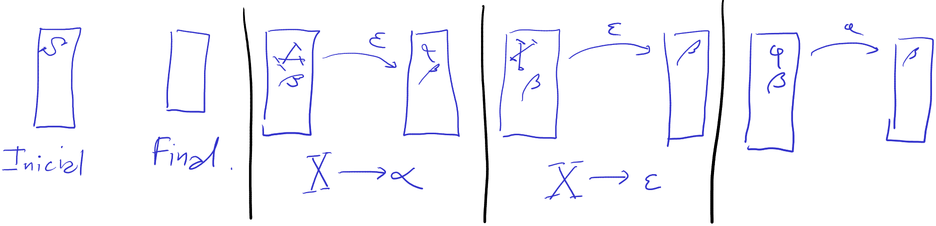
A diferencia de los autómatas finitos, el rol de los PDA es comparativamente secundario (a nivel práctico y en cuanto a uso en demostraciones).

Consumir palabras

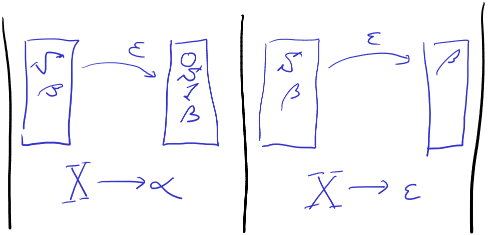
Los PDA dan un modelo de computación "aceptador" (*acceptor*), que va consumiendo una palabra y finalmente la acepta o no.

Ejemplo: $G_{01} := (\{S\}, \{0, 1\}, \{S \to \epsilon;\ S \to 0\,S\,1\}, S)$.

$L_{01} = \{0^n 1^n : n \in \mathbb{N}_0\}$

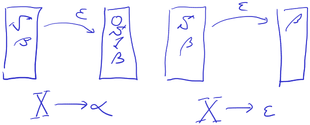


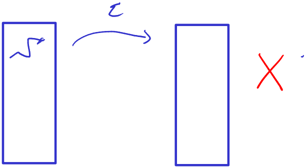
Para L_{01}:

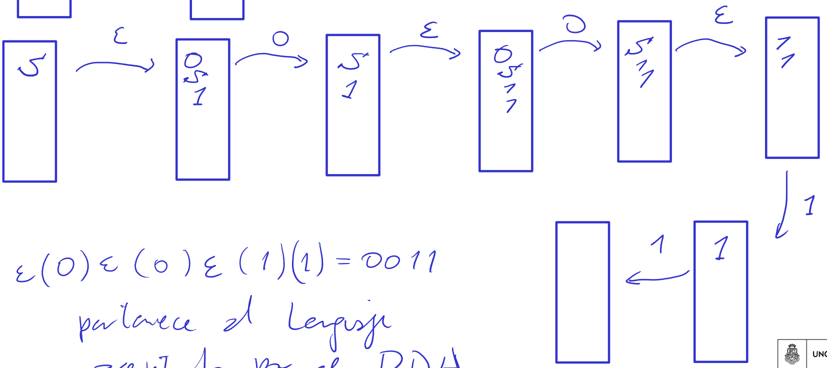


PDA para L_{01}

Consumir $0011 \in L_{01}$







$\varepsilon(0)\varepsilon(0)\varepsilon(1)(1) = 0011$

pertenece al Lenguaje aceptado por el PDA

Viejo *Pumping Lemma*

Lema de bombeo

Sea L **regular**. Toda palabra $\alpha \in L$ larga ($\geqslant k$) tiene un prefijo $\beta\gamma$ corto ($\leqslant k$) con un sufijo ($\gamma \neq \epsilon$) "repetible" (sin salirse de L).

Viejo *Pumping Lemma*

Lema de bombeo

Sea L **regular**. Toda palabra $\alpha \in L$ larga ($\geqslant k$) tiene un prefijo $\beta\gamma$ corto ($\leqslant k$) con un sufijo ($\gamma \neq \epsilon$) “repetible” (sin salirse de L).

su contrarrecíproca,

Si para todo $k \in \mathbb{N}$ existe $\alpha \in L$ con $|\alpha| \geqslant k$ tal que para todas β, γ, δ tales que $\alpha = \beta\gamma\delta$, $\gamma \neq \epsilon$ y $|\beta\gamma| \leqslant k$, existe $n > 0$ tal que $\beta\gamma^n\delta \notin L$, entonces L no es regular.

MUCHO TEXTO!!!

Viejo *Pumping Lemma*

Lema de bombeo

Sea L **regular**. Toda palabra $\alpha \in L$ larga ($\geqslant k$) tiene un prefijo $\beta\gamma$ corto ($\leqslant k$) con un sufijo ($\gamma \neq \epsilon$) "repetible" (sin salirse de L).

su contrarrecíproca,

Si para todo $k \in \mathbb{N}$ existe $\alpha \in L$ con $|\alpha| \geqslant k$ tal que para todas β, γ, δ tales que $\alpha = \beta\gamma\delta$, $\gamma \neq \epsilon$ y $|\beta\gamma| \leqslant k$, existe $n > 0$ tal que $\beta\gamma^n\delta \notin L$, entonces L no es regular.

y la versión estratégica:

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en tres pedazos $\beta\gamma\delta$ (pero debe cumplir que $|\beta\gamma| \leqslant k$ y que $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\beta\gamma^n\delta \notin L$.

Nuevo *Pumping Lemma*

Lema de bombeo para lenguajes libres de contexto

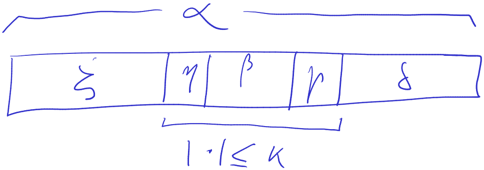
Sea L **libre de contexto**. Toda palabra $\alpha \in L$ larga ($\geqslant k$) tiene un segmento $\eta\beta\gamma$ corto ($\leqslant k$) con extremos ($\eta \neq \epsilon$ ó $\gamma \neq \epsilon$) “repetibles” (sin salirse de L).

Nuevo *Pumping Lemma*

Lema de bombeo para lenguajes libres de contexto

Sea L **libre de contexto**. Toda palabra $\alpha \in L$ larga ($\geqslant k$) tiene un segmento $\eta\beta\gamma$ corto ($\leqslant k$) con extremos ($\eta \neq \epsilon$ ó $\gamma \neq \epsilon$) “repetibles” (sin salirse de L).

Si para todo $k \in \mathbb{N}$ existe $\alpha \in L$ con $|\alpha| \geqslant k$ tal que para todos $\zeta, \eta, \beta, \gamma, \delta$ tales que $\alpha = \zeta\eta\beta\gamma\delta$, $|\eta\beta\gamma| \leqslant k$ y $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$, existe $n > 0$ tal que $\zeta\eta^n\beta\gamma^n\delta \notin L$, entonces L no es libre de contexto.



Nuevo *Pumping Lemma*

Lema de bombeo para lenguajes libres de contexto

Sea L **libre de contexto**. Toda palabra $\alpha \in L$ larga ($\geqslant k$) tiene un segmento $\eta\beta\gamma$ corto ($\leqslant k$) con extremos ($\eta \neq \epsilon$ ó $\gamma \neq \epsilon$) "repetibles" (sin salirse de L).

Si para todo $k \in \mathbb{N}$ existe $\alpha \in L$ con $|\alpha| \geqslant k$ tal que para todos $\zeta, \eta, \beta, \gamma, \delta$ tales que $\alpha = \zeta\eta\beta\gamma\delta$, $|\eta\beta\gamma| \leqslant k$ y $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$, existe $n > 0$ tal que $\zeta\eta^n\beta\gamma^n\delta \notin L$, entonces L no es libre de contexto.

Versión estratégica

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\eta\beta\gamma\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

Ejemplo de lenguaje no libre de contexto

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\eta\beta\gamma\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

Ejemplo de lenguaje no libre de contexto

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\eta\beta\gamma\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$ no es libre de contexto.

Estrategia

Ejemplo de lenguaje no libre de contexto

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\eta\beta\gamma\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$ no es libre de contexto.

Estrategia

- Adversario elige k.

Ejemplo de lenguaje no libre de contexto

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\underbrace{\eta\beta\gamma}\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$ no es libre de contexto.

Estrategia

- Adversario elige k.
- Nosotros proponemos $a^k b^k c^k \in L$.

$$\overbrace{a \cdots a}^{k}\ \overbrace{b \cdots b}^{k}\ \overbrace{c \cdots c}^{k}$$

$$\zeta \quad \eta \quad \beta \quad \gamma \quad \delta$$

$\eta\beta\gamma \in \{a,b\}^* \cup \{b,c\}^*$

$\eta^2\beta\gamma^2$ repite a's y/o b's y no c's o al revés.

Ejemplo de lenguaje no libre de contexto

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\eta\beta\gamma\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$ no es libre de contexto.

Estrategia

- Adversario elige k.
- Nosotros proponemos $a^k b^k c^k \in L$.
- Adversario descompone $a^k b^k c^k = \zeta\,\eta\,\beta\,\gamma\,\delta$

Ejemplo de lenguaje no libre de contexto

1. Adversario elige $k \in \mathbb{N}$.
2. Nosotros debemos dar $\alpha \in L$ con $|\alpha| \geqslant k$.
3. Adversario descompone α en cinco pedazos $\zeta\eta\beta\gamma\delta$ (pero debe cumplir que $|\eta\beta\gamma| \leqslant k$ y que $\eta \neq \epsilon$ ó $\gamma \neq \epsilon$).
4. Nosotros damos un $n > 0$ y armamos $\zeta\eta^n\beta\gamma^n\delta \notin L$.

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$ no es libre de contexto.

Estrategia

$\overbrace{a \cdots a}^{k}\overbrace{b \cdots b}^{k}\overbrace{c \cdots c}^{k}$

$\zeta \quad \eta \quad \beta \quad \gamma \quad \delta$

- Adversario elige k.
- Nosotros proponemos $a^k b^k c^k \in L$.
- Adversario descompone $a^k b^k c^k = \zeta\,\eta\,\beta\,\gamma\,\delta$
- Nosotros jugamos $n = 2$ y ganamos: $\alpha' = \zeta\,\eta^2\,\beta\,\gamma^2\,\delta \notin L$.

siempre se repite dos letras pero no las 3.

P. ej. como el diagrama, $|\alpha'|_c < |\alpha'|_b$ ó $|\alpha'|_c$

Cambio de notación

¡Advertencia!

A continuación, el adjetivo **contextual** reemplazará la insensata frase "sensible al contexto" que figura en los apuntes.

Gramáticas contextuales (CSG)

Lenguaje contextual

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$.

Gramáticas contextuales (CSG)

Lenguaje contextual

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}.$

$$S \longrightarrow aSBC \mid aBC$$ ⟶ producciones tipo CFG

$$\left.\begin{array}{rcl} CB & \longrightarrow & BC \\ aB & \longrightarrow & ab \\ bB & \longrightarrow & bb \\ bC & \longrightarrow & bc \\ cC & \longrightarrow & cc \end{array}\right\} \text{no.}$$

Gramáticas contextuales (CSG)

Lenguaje contextual

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}.$

$$
\begin{array}{rcl}
S & \longrightarrow & aSBC \mid aBC \\
CB & \longrightarrow & BC \\
aB & \longrightarrow & ab \\
bB & \longrightarrow & bb \\
bC & \longrightarrow & bc \\
cC & \longrightarrow & cc
\end{array}
$$

Derivemos aabbcc

$S \Rightarrow aSBC \Rightarrow aaBCBC$

$\Downarrow$

$\not\Leftarrow aabcBC \Leftarrow aabCBC$

$\Downarrow$

$aabbcc \Leftarrow aabbcC \Leftarrow aabbCC \Leftarrow aabBCC$

UNC Universidad Nacional de Córdoba 1613 - 2013 400 AÑOS

Gramáticas contextuales (CSG)

Lenguaje contextual

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}.$

$$
\begin{array}{lcl}
S & \longrightarrow & aSBC \mid \overbrace{aBC}^{\gamma} \\
\boxed{CB \longrightarrow BC} & & \\
aB & \longrightarrow & ab \\
bB & \longrightarrow & bb \\
bC & \longrightarrow & bc \\
cC & \longrightarrow & cc
\end{array}
$$

⟶ Producción CF: $\alpha, \beta = \varepsilon$
$A = S$, $\gamma :=$ lado der.

NO es CS

⟶ No es CF pero sí CS.

$$aB \longrightarrow ab$$
$$\alpha A \beta \longrightarrow \alpha \gamma \beta$$
$$a B \varepsilon \longrightarrow a b \varepsilon$$

Definición

Una **gramática contextual** tiene producciones de la forma $\alpha\, A\, \beta \longrightarrow \alpha\, \gamma\, \beta$ con $\gamma \neq \epsilon$, y $S \longrightarrow \epsilon$ si S no ocurre en ningún lado derecho.

α, β pueden ser ε.

UNC Universidad Nacional de Córdoba 400 AÑOS

Gramáticas contextuales (CSG)

Lenguaje contextual

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}$.

$$
\begin{array}{rcl}
S & \longrightarrow & aSBC \mid aBC \\
\color{red}{CB} & \longrightarrow & \color{red}{BC} \\
aB & \longrightarrow & ab \\
bB & \longrightarrow & bb \\
bC & \longrightarrow & bc \\
cC & \longrightarrow & cc
\end{array}
$$

Definición

Una **gramática contextual** tiene producciones de la forma $\alpha\, A\, \beta \longrightarrow \alpha\, \gamma\, \beta$ con $\gamma \neq \epsilon$, y $S \longrightarrow \epsilon$ si S no ocurre en ningún lado derecho.

Gramáticas contextuales (CSG)

Lenguaje contextual

$L_{abc} := \{a^n b^n c^n : n \in \mathbb{N}, n \geqslant 1\}.$

$$\begin{aligned}
S &\longrightarrow aSBC \mid aBC \\
CB &\longrightarrow BC \\
aB &\longrightarrow ab \\
bB &\longrightarrow bb \\
bC &\longrightarrow bc \\
cC &\longrightarrow cc
\end{aligned}$$

Sirve para $AB \longrightarrow CD$

Truco de Révész

$$\begin{aligned}
\overset{\alpha A \beta}{CB} &\longrightarrow \overset{\alpha \gamma \beta}{CZ} \\
CZ &\longrightarrow WZ \\
WZ &\longrightarrow WC \\
WC &\longrightarrow BC
\end{aligned}$$

Definición

Una **gramática contextual** tiene producciones de la forma $\alpha\, A\, \beta \longrightarrow \alpha\, \gamma\, \beta$ con $\gamma \neq \epsilon$, y $S \longrightarrow \epsilon$ si S no ocurre en ningún lado derecho.

UNC Universidad Nacional de Córdoba 400 AÑOS

NSPACE($O(n)$)

La familia de los lenguajes contextuales es enormemente más vasta que la de los libres de contexto.

NSPACE($O(n)$)

La familia de los lenguajes contextuales es enormemente más vasta que la de los libres de contexto.

- Equivalen problemas resolubles/decidibles usando “un core + RAM proporcional a la longitud de la entrada”.

Sin embargo, casi todas las CSG admiten una forma muy simple

NSPACE($O(n)$)

La familia de los lenguajes contextuales es enormemente más vasta que la de los libres de contexto.

- Equivalen problemas resolubles/decidibles usando "un core + RAM proporcional a la longitud de la entrada".

Sin embargo, casi todas las CSG admiten una forma muy simple:

Teorema (Forma normal de Kuroda)

Toda CSG que no derive ϵ es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD$, $A \longrightarrow BC$ ó $A \longrightarrow a$.

L_{abc} en forma normal

Teorema (Forma normal de Kuroda)

Toda CSG que no derive ϵ es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD,\ A \longrightarrow BC$ *ó* $A \longrightarrow a$.

L_{abc} en forma normal

Teorema (Forma normal de Kuroda)

Toda CSG que no derive ϵ es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD$, $A \longrightarrow BC$ ó $A \longrightarrow a$.

$$
\begin{array}{rcl}
S & \longrightarrow & aSBC \mid aBC \\
CB & \longrightarrow & BC \\
aB & \longrightarrow & ab \\
bB & \longrightarrow & bb \\
bC & \longrightarrow & bc \\
cC & \longrightarrow & cc
\end{array}
$$

UNC Universidad Nacional de Córdoba 1613 - 2013 400 AÑOS

L_{abc} en forma normal

Teorema (Forma normal de Kuroda)

Toda CSG que no derive ϵ es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD$, $A \longrightarrow BC$ ó $A \longrightarrow a$.

$$
\begin{array}{rcl}
S & \longrightarrow & aSBC \mid aBC \\
CB & \longrightarrow & BC \\
aB & \longrightarrow & ab \quad etc\ldots \\
bB & \longrightarrow & bb \\
bC & \longrightarrow & bc \\
cC & \longrightarrow & cc
\end{array}
$$

L_{abc} en forma normal

Teorema (Forma normal de Kuroda)

Toda CSG que no derive ϵ es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD$, $A \longrightarrow BC$ ó $A \longrightarrow a$.

$$
\begin{aligned}
S &\longrightarrow aSBC \mid aBC \\
CB &\longrightarrow BC \\
aB &\longrightarrow ab \quad etc \ldots \\
bB &\longrightarrow bb \\
bC &\longrightarrow bc \\
cC &\longrightarrow cc
\end{aligned}
$$

$$
\begin{aligned}
S &\longrightarrow AD \mid AE \\
A &\longrightarrow a \\
D &\longrightarrow SE \\
E &\longrightarrow BC
\end{aligned}
$$

$$S \longrightarrow \underbrace{a}_{A}\underbrace{SBC}_{D}$$

$$D \Longrightarrow \underbrace{S}_{S}\underbrace{BC}_{E}$$

$$E \longrightarrow BC$$

L_{abc} en forma normal

Teorema (Forma normal de Kuroda)

Toda CSG que no derive ϵ es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD$, $A \longrightarrow BC$ ó $A \longrightarrow a$.

$$\begin{array}{rcl} S & \longrightarrow & aSBC \mid aBC \\ CB & \longrightarrow & BC \\ aB & \longrightarrow & ab \quad etc\ldots \\ bB & \longrightarrow & bb \\ bC & \longrightarrow & bc \\ cC & \longrightarrow & cc \end{array}$$

$$\begin{array}{rcl} S & \longrightarrow & AD \mid AE \\ A & \longrightarrow & a \\ D & \longrightarrow & SE \\ E & \longrightarrow & BC \end{array}$$

La maldita palabra vacía

¿Qué pasaría si permitiéramos las producciones $A \longrightarrow \epsilon$ en la forma normal de Kuroda?

Gramáticas irrestrictas

Definición

Una gramática es **irrestricta** o de **tipo 0** si tiene producciones arbitrarias: $\alpha \longrightarrow \beta$ con $\alpha \neq \epsilon$.

$L(G) = \{\alpha \in T^* : S \stackrel{*}{\Longrightarrow} \alpha\}$

Gramáticas irrestrictas

Definición

Una gramática es **irrestricta** o de **tipo 0** si tiene producciones arbitrarias:
$\alpha \longrightarrow \beta$ con $\alpha \neq \epsilon$.

Teorema (Kuroda para gramáticas tipo 0)

Toda gramática es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD,\ A \longrightarrow BC,\ A \longrightarrow a$ *ó* $A \longrightarrow \epsilon.$

Gramáticas irrestrictas

Definición

Una gramática es **irrestricta** o de **tipo 0** si tiene producciones arbitrarias: $\alpha \longrightarrow \beta$ con $\alpha \neq \epsilon$.

Teorema (Kuroda para gramáticas tipo 0)

Toda gramática es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD$, $A \longrightarrow BC$, $A \longrightarrow a$ *ó* $A \longrightarrow \epsilon$*.*

El salto de CSG a gramáticas tipo 0 es más grande que el de CFG a CSG

Gramáticas irrestrictas

Definición

Una gramática es **irrestricta** o de **tipo 0** si tiene producciones arbitrarias: $\alpha \longrightarrow \beta$ con $\alpha \neq \epsilon$.

Teorema (Kuroda para gramáticas tipo 0)

Toda gramática es (débilmente) equivalente a una con producciones de las formas $AB \longrightarrow CD,\ A \longrightarrow BC,\ A \longrightarrow a$ *ó* $A \longrightarrow \epsilon$.

El salto de CSG a gramáticas tipo 0 es más grande que el de CFG a CSG:

- Para cualquier lenguaje L listable por cualquier método computacional conocido (cantidad de cores arbitraria, con acceso a memoria y tiempo de cómputo ilimitados) existe G de tipo 0 tal que $L(G) = L$.

Computabilidad

Sea $L \subseteq \Sigma^*$.

Computabilidad

Sea $L \subseteq \Sigma^*$.

Definición

- L es **computable** (**decidible**, **recursivo**) si hay un algoritmo que responde correctamente a la pregunta "$\alpha \in L$" para cada α.

Computabilidad

Sea $L \subseteq \Sigma^*$.

Definición

- L es **computable** (**decidible**, **recursivo**) si hay un algoritmo que responde correctamente a la pregunta "$\alpha \in L$" para cada α.
- L es **listable** (**recursivamente enumerable**) si hay un algoritmo que eventualmente lista todos las cadenas en L.

Computabilidad

Sea $L \subseteq \Sigma^*$.

Definición

- L es **computable** (**decidible**, **recursivo**) si hay un algoritmo que responde correctamente a la pregunta "$\alpha \in L$" para cada α.
- L es **listable** (**recursivamente enumerable**) si hay un algoritmo que eventualmente lista todos las cadenas en L.

el primero.

El único requerimiento sobre los algoritmos en ~~ambos~~ ítemes es que *terminen*, sin limitación sobre uso de memoria y tiempo de cómputo.

Computabilidad

Sea $L \subseteq \Sigma^*$.

Definición

- L es **computable** (**decidible**, **recursivo**) si hay un algoritmo que responde correctamente a la pregunta "$\alpha \in L$" para cada α.
- L es **listable** (**recursivamente enumerable**) si hay un algoritmo que eventualmente lista todos las cadenas en L.

El único requerimiento sobre los algoritmos en ambos ítemes es que *terminen*, sin limitación sobre uso de memoria y tiempo de cómputo.

Teorema

- *Hay conjuntos listables que no son recursivos.*
- *Los lenguajes de las gramáticas irrestrictas son exactamente los recursivamente enumerables.*

Jerarquía de Chomsky

Una clasificación de gramáticas y sus lenguajes.

Jerarquía de Chomsky

Una clasificación de gramáticas y sus lenguajes.

- **Tipo 3**: Regulares. $A \longrightarrow a\,B,\ A \longrightarrow \epsilon$.

Jerarquía de Chomsky

Una clasificación de gramáticas y sus lenguajes.

- **Tipo 3**: Regulares. $A \longrightarrow a\,B,\ A \longrightarrow \epsilon$.
- **Tipo 2**: Libres de contexto. $A \longrightarrow \alpha$.

Jerarquía de Chomsky

Una clasificación de gramáticas y sus lenguajes.

- **Tipo 3**: Regulares. $A \longrightarrow a\,B,\; A \longrightarrow \epsilon$.
- **Tipo 2**: Libres de contexto. $A \longrightarrow \alpha$.
- **Tipo 1**: Contextuales. $\alpha\,A\,\beta \longrightarrow \alpha\,\gamma\,\beta$ ($\gamma \neq \emptyset$),
  $S \longrightarrow \epsilon$ (si no está del lado derecho).

Una clasificación de gramáticas y sus lenguajes.

- **Tipo 3**: Regulares. $A \longrightarrow a\,B,\ A \longrightarrow \epsilon$.
- **Tipo 2**: Libres de contexto. $A \longrightarrow \alpha$.
- **Tipo 1**: Contextuales. $\alpha\,A\,\beta \longrightarrow \alpha\,\gamma\,\beta$ ($\gamma \neq \emptyset$),
  $S \longrightarrow \epsilon$ (si no está del lado derecho).
- **Tipo 0**: Irrestrictas $\alpha \longrightarrow \gamma$ ($\alpha \neq \epsilon$).

Fin de la Tercera Parte