# Lenguajes No Regulares

Introducción a la Logica y la Computación (3era Parte)

*Docentes: Badano, Bustos, Costamagna, Tellechea, Zigaran*

Año 2024

# Lenguajes No-Regulares

- Hasta aquí, todos los lenguajes vistos resultaron ser lenguajes regulares.

# Lenguajes No-Regulares

- Hasta aquí, todos los lenguajes vistos resultaron ser lenguajes regulares.
- Por lo tanto, surge la pregunta sobre si verdaderamente existen lenguajes que no lo sean.

# Lenguajes No-Regulares

- Hasta aquí, todos los lenguajes vistos resultaron ser lenguajes regulares.
- Por lo tanto, surge la pregunta sobre si verdaderamente existen lenguajes que no lo sean.
- La respuesta es si, claro que existen y hay infinidad de ellos.

# Lenguajes No-Regulares

- Hasta aquí, todos los lenguajes vistos resultaron ser lenguajes regulares.
- Por lo tanto, surge la pregunta sobre si verdaderamente existen lenguajes que no lo sean.
- La respuesta es si, claro que existen y hay infinidad de ellos.

Por ejemplo, el lenguaje de todas las cadenas que son secuencia de $a$'s seguido de una secuencia de $b$'s de igual longitud no es regular:

$$L_1 = \{a^m b^m : m \geq 0\} \notin LR^{\Sigma}.$$

# Lenguajes No-Regulares

- Hasta aquí, todos los lenguajes vistos resultaron ser lenguajes regulares.
- Por lo tanto, surge la pregunta sobre si verdaderamente existen lenguajes que no lo sean.
- La respuesta es si, claro que existen y hay infinidad de ellos.

Por ejemplo, el lenguaje de todas las cadenas que son secuencia de $a$'s seguido de una secuencia de $b$'s de igual longitud no es regular:

$$L_1 = \{a^m b^m : m \geq 0\} \notin LR^{\Sigma}.$$

**Una intuición posible detrás de este resultado**, es que no puede existir un AF que compute $L_1$, pues los AF son máquinas puramente locales (sin memoria). No guardan ningún registro de lo que ya computaron de la cadena. Y claramente, para poder computar $L_1$ necesitamos de algún mecanismo para saber cuantas a's procesamos previamente para poder compararla más tarde con la cantidad de $b$'s que se procesarán.

# Más lenguajes No-Regulares

El lenguaje de las cadenas capicuas en el alfabeto no es regular:

$$L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^{\Sigma}.$$

# Más lenguajes No-Regulares

El lenguaje de las cadenas capicuas en el alfabeto no es regular:

$$L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^{\Sigma}.$$

El lenguaje de todas las reflexiones en el alfabeto no es regular:

$$L_3 = \{\alpha\alpha : \alpha \in \Sigma^*\} \notin LR^{\Sigma}.$$

# Más lenguajes No-Regulares

El lenguaje de las cadenas capicuas en el alfabeto no es regular:

$$L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^\Sigma.$$

El lenguaje de todas las reflexiones en el alfabeto no es regular:

$$L_3 = \{\alpha\alpha : \alpha \in \Sigma^*\} \notin LR^\Sigma.$$

Ninguno de estos tres lenguajes puede tener un AF que lo acepte.

# Más lenguajes No-Regulares

El lenguaje de las cadenas capicuas en el alfabeto no es regular:

$$L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^\Sigma.$$

El lenguaje de todas las reflexiones en el alfabeto no es regular:

$$L_3 = \{\alpha\alpha : \alpha \in \Sigma^*\} \notin LR^\Sigma.$$

Ninguno de estos tres lenguajes puede tener un AF que lo acepte.

$L_1$ y $L_2$ son lenguajes independientes de contexto, mientras que $L_3$ es sensible al contexto.

# Técnica Formal para Probar No-Regularidad

- Más alla de la intuición, necesitamos una herramienta para probar con **rigurosidad matemática** que dicho autómoma no existe, es decir, una técnica que pruebe formalmente que un lenguaje no es regular.

# Técnica Formal para Probar No-Regularidad

- Más alla de la intuición, necesitamos una herramienta para probar con **rigurosidad matemática** que dicho autómoma no existe, es decir, una técnica que pruebe formalmente que un lenguaje no es regular.
- Para ello, utilizaremos el **"Pumping Lema"** que es una propiedad que todo lenguaje regular cumple.

# Técnica Formal para Probar No-Regularidad

- Más alla de la intuición, necesitamos una herramienta para probar con **rigurosidad matemática** que dicho autómoma no existe, es decir, una técnica que pruebe formalmente que un lenguaje no es regular.
- Para ello, utilizaremos el **"Pumping Lema"** que es una propiedad que todo lenguaje regular cumple.
- Por lo tanto, si probamos que un lenguaje no satisface la propiedad, entonces ese lenguaje no puede ser regular.

# Pumping Lema (o Lema de Bombeo)

La propiedad dice que toda cadena de un lenguaje regular tiene una subcadena que puede repetirse una cantidad arbitraria de veces (bombeo) y esta nueva cadena "bombeada" pertenece al lenguaje.

# Pumping Lema (o Lema de Bombeo)

La propiedad dice que toda cadena de un lenguaje regular tiene una subcadena que puede repetirse una cantidad arbitraria de veces (bombeo) y esta nueva cadena "bombeada" pertenece al lenguaje.

## Lema (Pumping Lema)

*Para todo $L \in LR^{\Sigma}$, existe $n \in \mathbb{N}$, llamada constante de bombeo de $L$, tal que, si $\alpha \in L$ con $|\alpha| \geq n$, entonces $\alpha$ se puede descomponer como $\alpha = \alpha_1\alpha_2\alpha_3$ donde:*

*1)* $|\alpha_1\alpha_2| \leq n$
*2)* $\alpha_2 \neq \epsilon$
*3)* $(\alpha_1\alpha_2^i\alpha_3) \in L$, *para todo* $i \in \mathbb{N}$

# Pumping Lema (o Lema de Bombeo)

La propiedad dice que toda cadena de un lenguaje regular tiene una subcadena que puede repetirse una cantidad arbitraria de veces (bombeo) y esta nueva cadena "bombeada" pertenece al lenguaje.

## Lema (Pumping Lema)

*Para todo $L \in LR^{\Sigma}$, existe $n \in \mathbb{N}$, llamada constante de bombeo de $L$, tal que, si $\alpha \in L$ con $|\alpha| \geq n$, entonces $\alpha$ se puede descomponer como $\alpha = \alpha_1\alpha_2\alpha_3$ donde:*

*1)* $|\alpha_1\alpha_2| \leq n$
*2)* $\alpha_2 \neq \epsilon$
*3)* $(\alpha_1\alpha_2^i\alpha_3) \in L$, *para todo* $i \in \mathbb{N}$

Demo. Si $L \in LR^{\Sigma}$, entonces existe un AFD $M = (\Sigma, Q, q_0, F, \delta)$ tal que $L(M) = L$ y sea $n = |Q|$.

# Pumping Lema (o Lema de Bombeo)

La propiedad dice que toda cadena de un lenguaje regular tiene una subcadena que puede repetirse una cantidad arbitraria de veces (bombeo) y esta nueva cadena "bombeada" pertenece al lenguaje.

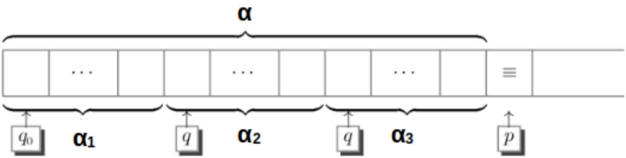
## Lema (Pumping Lema)

*Para todo $L \in LR^{\Sigma}$, existe $n \in \mathbb{N}$, llamada constante de bombeo de $L$, tal que, si $\alpha \in L$ con $|\alpha| \geq n$, entonces $\alpha$ se puede descomponer como $\alpha = \alpha_1\alpha_2\alpha_3$ donde:*

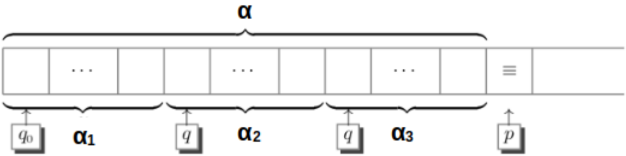
*1)* $|\alpha_1\alpha_2| \leq n$
*2)* $\alpha_2 \neq \epsilon$
*3)* $(\alpha_1\alpha_2^i\alpha_3) \in L$*, para todo* $i \in \mathbb{N}$

Demo. Si $L \in LR^{\Sigma}$, entonces existe un AFD $M = (\Sigma, Q, q_0, F, \delta)$ tal que $L(M) = L$ y sea $n = |Q|$.
Si $\alpha \in L$ y $|\alpha| \geq n$, entonces al procesar la cadena $\alpha$ existe al menos un estado $q \in Q$ que es visitado dos veces (o sea, se repite).

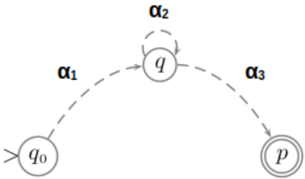
Por lo tanto, notar que $\alpha$ se puede descomponer de la siguiente manera:



Por lo tanto, notar que $\alpha$ se puede descomponer de la siguiente manera:




Luego, la subcadena $\alpha_2$ puede ser repetida o bombeada una cantidad cualquiera de veces, y aún asi, el procesamiento de la cadena bombeada arribará al estado $p \in F$ , por lo tanto, pertenece al lenguaje $L$:




$$(\alpha_1 \alpha_2^i \alpha_3) \in L, \text{ para todo } i \geq 0$$

# Pasos para Probar No-Regularidad

Para demostrar formalmente que un lenguaje $L$ no es regular, utilizando el Pumping Lema, procedemos siempre de la siguiente manera:

# Pasos para Probar No-Regularidad

Para demostrar formalmente que un lenguaje $L$ no es regular, utilizando el Pumping Lema, procedemos siempre de la siguiente manera:

1) Suponer que $L \in LR^{\Sigma}$ y denotar con $n$ a la cte de bombeo de $L$.

# Pasos para Probar No-Regularidad

Para demostrar formalmente que un lenguaje $L$ no es regular, utilizando el Pumping Lema, procedemos siempre de la siguiente manera:

1) Suponer que $L \in LR^{\Sigma}$ y denotar con $n$ a la cte de bombeo de $L$.

2) Tomar una $\alpha \in L$, con $|\alpha| \geq n$, arbitraria pero "conveniente" que esté definida en términos de $n$.

# Pasos para Probar No-Regularidad

Para demostrar formalmente que un lenguaje $L$ no es regular, utilizando el Pumping Lema, procedemos siempre de la siguiente manera:

1) Suponer que $L \in LR^{\Sigma}$ y denotar con $n$ a la cte de bombeo de $L$.

2) Tomar una $\alpha \in L$, con $|\alpha| \geq n$, arbitraria pero "conveniente" que esté definida en términos de $n$.

3) Descomponer $\alpha$ en subcadenas $\alpha_1, \alpha_2, \alpha_3$ como indica el Pumping Lema.

# Pasos para Probar No-Regularidad

Para demostrar formalmente que un lenguaje $L$ no es regular, utilizando el Pumping Lema, procedemos siempre de la siguiente manera:

1) Suponer que $L \in LR^{\Sigma}$ y denotar con $n$ a la cte de bombeo de $L$.

2) Tomar una $\alpha \in L$, con $|\alpha| \geq n$, arbitraria pero "conveniente" que esté definida en términos de $n$.

3) Descomponer $\alpha$ en subcadenas $\alpha_1, \alpha_2, \alpha_3$ como indica el Pumping Lema.

4) Hallar un $i \geq 0$ tal que $(\alpha_1 \alpha_2^i \alpha_3) \notin L$ (Absurdo!!!)

# Pasos para Probar No-Regularidad

Para demostrar formalmente que un lenguaje $L$ no es regular, utilizando el Pumping Lema, procedemos siempre de la siguiente manera:

1) Suponer que $L \in LR^{\Sigma}$ y denotar con $n$ a la cte de bombeo de $L$.

2) Tomar una $\alpha \in L$, con $|\alpha| \geq n$, arbitraria pero "conveniente" que esté definida en términos de $n$.

3) Descomponer $\alpha$ en subcadenas $\alpha_1, \alpha_2, \alpha_3$ como indica el Pumping Lema.

4) Hallar un $i \geq 0$ tal que $(\alpha_1 \alpha_2^i \alpha_3) \notin L$ (Absurdo!!!)

El absurdo proviene de suponer en (1) que $L \in LR^{\Sigma}$ $\therefore$ $L \notin LR^{\Sigma}$.

# Aplicando el Pumping Lema

Sea $\Sigma = \{a, b\}$.

Veamos que $L_1 = \{a^m b^m : m \geq 0\} \notin LR^{\Sigma}$.

# Aplicando el Pumping Lema

Sea $\Sigma = \{a, b\}$.

Veamos que $L_1 = \{a^m b^m : m \geq 0\} \notin LR^{\Sigma}$.

Sup. $L_1 \in LR^{\Sigma}$ y sea $n$ la cte de bombeo de $L_1$.

# Aplicando el Pumping Lema

Sea $\Sigma = \{a, b\}$.

Veamos que $L_1 = \{a^m b^m : m \geq 0\} \notin LR^{\Sigma}$.

Sup. $L_1 \in LR^{\Sigma}$ y sea $n$ la cte de bombeo de $L_1$.

Tomamos $\alpha = a^n b^n \in L_1$, luego $|\alpha| = 2n \geq n$.

# Aplicando el Pumping Lema

Sea $\Sigma = \{a, b\}$.

Veamos que $L_1 = \{a^m b^m : m \geq 0\} \notin LR^\Sigma$.

Sup. $L_1 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_1$.

Tomamos $\alpha = a^n b^n \in L_1$, luego $|\alpha| = 2n \geq n$.

Por Pumping Lema, $\alpha = \alpha_1\alpha_2\alpha_3$ donde:

$\alpha_1 = a^r$ con $r \geq 0$
$\alpha_2 = a^s$ con $s \geq 1$
$\alpha_3 = a^{n-(s+r)} b^n$

# Aplicando el Pumping Lema

Sea $\Sigma = \{a, b\}$.

Veamos que $L_1 = \{a^m b^m : m \geq 0\} \notin LR^\Sigma$.

Sup. $L_1 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_1$.

Tomamos $\alpha = a^n b^n \in L_1$, luego $|\alpha| = 2n \geq n$.

Por Pumping Lema, $\alpha = \alpha_1\alpha_2\alpha_3$ donde:

$$\begin{aligned}
\alpha_1 &= a^r \text{ con } r \geq 0 \\
\alpha_2 &= a^s \text{ con } s \geq 1 \\
\alpha_3 &= a^{n-(s+r)} b^n
\end{aligned}$$

Para $i = 0$, tenemos que:

$$\begin{aligned}
\alpha_1\alpha_2^0\alpha_1 &= a^r (a^s)^0 a^{n-(s+r)} b^n \\
&= a^r \epsilon a^{n-s-r} b^n \\
&= a^{n-s} b^n \notin L_1, \text{ pues } s \geq 1 \therefore \text{ Absurdo.}
\end{aligned}$$

# Aplicando el Pumping Lema

Sea $\Sigma = \{a, b\}$.

Veamos que $L_1 = \{a^m b^m : m \geq 0\} \notin LR^\Sigma$.

Sup. $L_1 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_1$.

Tomamos $\alpha = a^n b^n \in L_1$, luego $|\alpha| = 2n \geq n$.

Por Pumping Lema, $\alpha = \alpha_1 \alpha_2 \alpha_3$ donde:

$\alpha_1 = a^r$ con $r \geq 0$
$\alpha_2 = a^s$ con $s \geq 1$
$\alpha_3 = a^{n-(s+r)} b^n$

Para $i = 0$, tenemos que:

$$\begin{aligned} \alpha_1 \alpha_2^0 \alpha_1 &= a^r (a^s)^0 a^{n-(s+r)} b^n \\ &= a^r \epsilon a^{n-s-r} b^n \\ &= a^{n-s} b^n \notin L_1, \text{ pues } s \geq 1 \therefore \text{ Absurdo.} \end{aligned}$$

$\therefore L_1 \notin LR^\Sigma$.

Veamos que $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^{\Sigma}$, es decir, el lenguaje de todas las cadenas capicuas del alfabeto no es regular.

Veamos que $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^{\Sigma}$, es decir, el lenguaje de todas las cadenas capicuas del alfabeto no es regular.

Sup. $L_2 \in LR^{\Sigma}$ y sea $n$ la cte de bombeo de $L_2$.

Veamos que $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^\Sigma$, es decir, el lenguaje de todas las cadenas capicuas del alfabeto no es regular.

Sup. $L_2 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_2$.

Tomamos $\alpha = a^n b a^n \in L_2$, luego $|\alpha| = 2n + 1 \geq n$.

Veamos que $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^\Sigma$, es decir, el lenguaje de todas las cadenas capicuas del alfabeto no es regular.

Sup. $L_2 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_2$.

Tomamos $\alpha = a^n b a^n \in L_2$, luego $|\alpha| = 2n + 1 \geq n$.

Por Pumping Lema, $\alpha = \alpha_1 \alpha_2 \alpha_3$ donde:

$\alpha_1 = a^r$ con $r \geq 0$
$\alpha_2 = a^s$ con $s \geq 1$
$\alpha_3 = a^{n-(s+r)} b a^n$

Veamos que $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^\Sigma$, es decir, el lenguaje de todas las cadenas capicuas del alfabeto no es regular.

Sup. $L_2 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_2$.

Tomamos $\alpha = a^n b a^n \in L_2$, luego $|\alpha| = 2n + 1 \geq n$.

Por Pumping Lema, $\alpha = \alpha_1 \alpha_2 \alpha_3$ donde:

$\alpha_1 = a^r$ con $r \geq 0$
$\alpha_2 = a^s$ con $s \geq 1$
$\alpha_3 = a^{n-(s+r)} b a^n$

Para $i = 2$, tenemos que:

$$\begin{aligned}\alpha_1 \alpha_2^2 \alpha_3 &= a^r (a^s)^2 a^{n-(s+r)} b a^n \\ &= a^r a^{2s} a^{n-s-r} b a^n \\ &= a^{n+s} b a^n \notin L_1, \text{ pues } s \geq 1 \therefore \text{Absurdo.}\end{aligned}$$

Veamos que $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\} \notin LR^\Sigma$, es decir, el lenguaje de todas las cadenas capicuas del alfabeto no es regular.

Sup. $L_2 \in LR^\Sigma$ y sea $n$ la cte de bombeo de $L_2$.

Tomamos $\alpha = a^n b a^n \in L_2$, luego $|\alpha| = 2n + 1 \geq n$.

Por Pumping Lema, $\alpha = \alpha_1 \alpha_2 \alpha_3$ donde:

$\alpha_1 = a^r$ con $r \geq 0$
$\alpha_2 = a^s$ con $s \geq 1$
$\alpha_3 = a^{n-(s+r)} b a^n$

Para $i = 2$, tenemos que:

$$\begin{aligned} \alpha_1 \alpha_2^2 \alpha_3 &= a^r (a^s)^2 a^{n-(s+r)} b a^n \\ &= a^r a^{2s} a^{n-s-r} b a^n \\ &= a^{n+s} b a^n \notin L_1, \text{ pues } s \geq 1 \therefore \text{Absurdo.} \end{aligned}$$

$\therefore L_2 \notin LR^\Sigma$.

# Resumen

- Hay lenguajes que no son regulares tales como $L_1 = \{a^m b^m : m \geq 0\}$ y $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\}$.

# Resumen

- Hay lenguajes que no son regulares tales como $L_1 = \{a^m b^m : m \geq 0\}$ y $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\}$.
- El Pumping Lema y razonamiento por el absurdo conforman una técnica para probar formalmente que un lenguaje no es regular.

# Resumen

- Hay lenguajes que no son regulares tales como $L_1 = \{a^m b^m : m \geq 0\}$ y $L_2 = \{\alpha \in \Sigma^* : \alpha = \alpha^R\}$.
- El Pumping Lema y razonamiento por el absurdo conforman una técnica para probar formalmente que un lenguaje no es regular.
- $L_1$ y $L_2$ son lenguajes independientes de contexto y esto se prueba mediante autómatas a pila y/o gramáticas independientes de contexto (modelos que serán estudiados en 4to año).

# Bibliografía

Rodrigo De Castro Korgi.
"Teoria de la Computación". Lenguajes, Autómatas, Gramáticas.
*Capítulo 3: sección 3.1.*

# Fin 3era Parte!