

# Expresiones Regulares

Alan Reyes-Figueroa

Teoría de la Computación

(Aula 03b) 12.julio.2023

Expresiones regulares  
Árboles sintácticos  
Notación Polaca Reversa

# Expresiones regulares

Una *expresión regular* es una representación de un lenguaje. (no de cualquier lenguaje)

Los lenguajes que son representables mediante expresiones regulares se llaman *lenguajes regulares*.

Dado un alfabeto  $\Sigma$ , para representar un lenguaje regular usamos los símbolos en  $\Sigma$ , y ciertos operadores especiales:

# Expresiones regulares

- $ab$  ó  $a \cdot b$ , para la concatenación
- $a \cup b$ , ó  $a \mid b$  ó  $a + b$ , para la unión
- $a^*$  para la cerradura de Kleene
- $a^+$  para la cerradura positiva
- $(, ), [, ]$ , para definir agrupaciones y jerarquías

También se usan otros símbolos como  
abreviaturas:  $[a_1, a_2, \dots, a_n]$

$[a_1 - a_n]$

# Expresiones regulares

Las expresiones regulares en un alfabeto  $\Sigma$  se construyen siguiendo las

1.  $\varepsilon$  y cualquier elemento de  $\Sigma$  es una expresión regular.
2. Si  $\alpha$  y  $\beta$  son expresiones regulares, también lo es  $\alpha\beta$ .
3. Si  $\alpha$  y  $\beta$  son expresiones regulares, también lo es  $\alpha \mid \beta$ .
4. Si  $\alpha$  es expresión regular, también lo son  $\alpha^*$  y  $\alpha^+$ .
5. Sólo las reglas 1-4 generan expresiones regulares.

# Precedencia de operadores

1.  $()$  operadores de agrupación se aplican primero.
2.  $*$  se aplica antes que  $|$  y que  $.$
3.  $.$  Concatenación precede a  $|$ .
4.  $+$  se aplica al final.

Ejemplos:

$ab^*$  es equivalente a  $a(b^*)$

$ab|c$  es equivalente a  $(ab)|c$

$a|b^*$  es equivalente a  $a|(b^*)$

# Expresiones regulares

Ejemplo: En  $\Sigma = \{0,1\}$

$a = (0|1)^*0$  es expresión regular.

Representa todas las cadenas terminadas en 0.

Si  $L$  es el lenguaje representado por la expresión  $a$ ,  
en  $L$  tenemos las cadenas

0	00	000	100	0000	...
	10	010	110	0010	

# Expresiones regulares

Ejemplo: En  $\Sigma = \{a,b\}$

$\beta = b^*(abb^*)(a|\epsilon)$  es expresión regular.

Representa a las cadenas que comienzan con un número cualquiera de b's, luego tiene ab, luego tiene cualquier número de b's luego terminan en a ó en  $\epsilon$ .

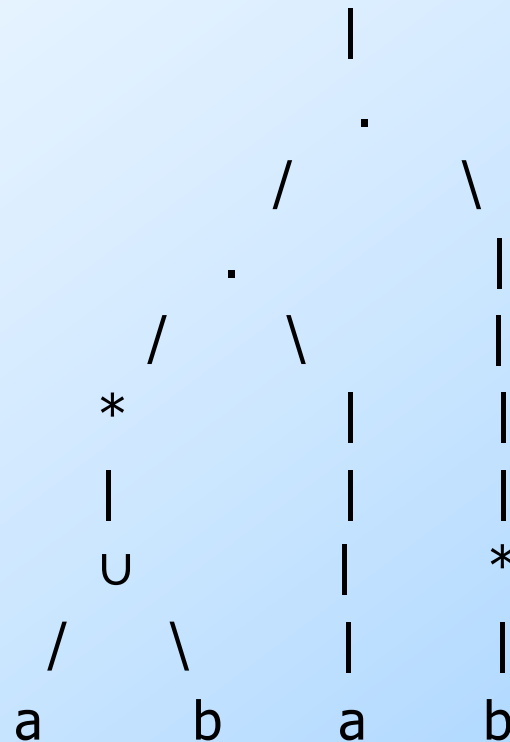
En el lenguaje L representado por  $\beta$  tenemos

ab	bab	baba	bbab
abb	aba	abbb	babb

# Árboles sintácticos

Cada expresión regular tiene asociado un **árbol sintáctico** o (*parse tree*), el cual representa de forma visual la estructura de la regexp

(a|b)\*ab\*





# Notaciones

Notación infix: La que usamos regularmente. Los operadores binarios van en medio de los términos.

$$( a \mid b )^* a b b^*$$

Notación postfix: También se llama **notación polaca revertida** (RPN), o **notación de Lukasiewicz**. Los operadores van después.

$$a b \mid * a \bullet b \bullet b * \bullet$$

Notación prefix: También se llama **notación polaca** (PN). Los operadores preceden a los términos.

$$\bullet \bullet \bullet * \mid a b b * b$$

# Ejercicios

Para cada una de las expresiones regulares siguientes, construir:

- El árbol sintáctico
- La expresión en notación postfix
- El AFD que representa la expresión
- La tabla de transiciones