Linguagens e Expressões Regulares

Linguagens Formais e Autómatos

Francisco Coelho
fc@di.uevora.pt

Departamento de Informática
Escola de Ciências e Tecnologia
Universidade de Évora

UNIVERSIDADE DE ÉVORA

UNIVERSIDADE DE ÉVORA

Alfabetos, Palavras e Linguagens

Expressões Regulares

Alfabetos

UNIVERSIDADE DE ÉVORA

Um alfabeto (Σ, T) é um conjunto finito, a cujos elementos chamamos letras ou símbolos (representados por $a, b, c, d, \ldots$).

- $\{a, b, c, \ldots, x, y, z\}$
- $\{0, 1\}$
- $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- $\{0, 1, \ldots, 9\} \cup \{+, -, \times, \div, (,)\}$
- as *palavras reservadas* de uma linguagem de programação (`while`, `if`, *etc*)

UNIVERSIDADE DE ÉVORA

Uma palavra *sobre o alfabeto* Σ é uma sequência finita de letras de Σ (representadas por p, q, u, v, w, x, y, z).

A palavra vazia tem zero símbolos e representa-se por λ (também por ϵ ou ε).

> *Formalmente, uma palavra* w*, de tamanho* n*, sobre o alfabeto* Σ*, é uma função de assinatura* $w : \{1, \ldots, n\} \to \Sigma$*:*

i	1	2	3	4	5	6	7	8	9	10	11	12
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
$w(i)$	u	n	i	v	e	r	s	i	d	a	d	e

Operações com palavras – comprimento

UNIVERSIDADE DE ÉVORA

- comprimento: $|\lambda| = 0$ e, se $n \geq 1$, $|x_1 \cdots x_n| = n$.
- Se w é uma palavra sobre Σ e $a \in \Sigma$ então wa é uma palavra sobre Σ.
- A função comprimento de uma palavra também pode ser definida por

  base $|\lambda| = 0$

  passo $|wa| = |w| + 1$ (w é uma palavra sobre Σ e $a \in \Sigma$)

Conjunto de todas as palavras

UNIVERSIDADE DE ÉVORA

O fecho do alfabeto Σ, representado por Σ^*, é o conjunto de **todas** as palavras sobre Σ e define-se por:

base $\lambda \in \Sigma^*$

passo se $w \in \Sigma^*$ e $a \in \Sigma$ então $wa \in \Sigma^*$

fecho $w \in \Sigma^*$ se e só se pode ser obtida por um número finito de aplicações do passo a partir de λ

O fecho garante, por exemplo, que em Σ^* não existem palavras infinitas nem palavras formadas por símbolos de outros alfabetos.

Excluindo λ obtemos

$$\Sigma^+ = \bigcup_{n>0} \Sigma^n = \Sigma\Sigma^*$$

UNIVERSIDADE DE ÉVORA

A concatenação (ou produto) de duas palavras $u, v \in \Sigma^*$, escrita $u \cdot v$ ou uv é uma operação binária em Σ^* definida por:

base Se $|v| = 0$ então $(v = \lambda)$ e $u \cdot v = u$

passo Se $|v| = n > 0$ então (existe uma palavra $w \in \Sigma^*$ com $|w| = n - 1$ e uma letra $a \in \Sigma$ tais que) $v = wa$ e nesse caso $u \cdot v = (u \cdot w)\, a$

As potências de uma palavra $u \in \Sigma^*$ são

$$u^0 = \lambda$$
$$u^1 = u$$
$$u^2 = uu$$
$$u^3 = u^2 u = uuu$$
$$\vdots$$
$$u^n = u^{n-1}u = u \cdots u \text{ concatenada } n \text{ vezes}$$

Operações com palavras – inversão

UNIVERSIDADE DE ÉVORA

A inversa de $u \in \Sigma^*$, escrita u^R ou u^{-1}, é uma operação unária em Σ^* definida por:

base se $|u| = 0$ então $u = \lambda$ e $u^R = \lambda$

passo se $|u| = n > 0$ então $u = wa$ com $|w| = n - 1$ e $u^R = aw^R$

Isto é, se $u = abc \cdots xyz$ *então* $u^R = zyx \cdots cba$

Subpalavra, prefixo e sufixo

UNIVERSIDADE DE ÉVORA

Seja $x = uvw \in \Sigma^*$ uma palavra. Então:

- u é um prefixo de x
- v é uma subpalavra de x
- w é um sufixo de x

Outra forma de enunciar estas relações é:

- v é uma subpalavra de x se existem palavras u, w tais que $x = uvw$;
- nesse caso, se $|u| = 0$ então v também é um prefixo e
- se $|w| = 0$ então v também é um sufixo

UNIVERSIDADE DE ÉVORA

Teorema (Inversão da concatenação)

Para quaisquer palavras x, y

$$(xy)^R = y^R x^R$$

Demonstração.

Se $x = \lambda$ o resultado é evidente ($xy = \lambda y = y$).
Se $x = x_1 \cdots x_n$ e $y = y_1 \cdots y_m$,

$$\begin{aligned}(xy)^R &= (x_1 \cdots x_n y_1 \cdots y_m)^R \\ &= y_m \cdots y_1 x_n \cdots x_1 \\ &= (y_m \cdots y_1)(x_n \cdots x_1) \\ &= y^R x^R\end{aligned}$$

□

Propriedades do Fecho

Se Σ e Γ forem alfabetos:

- $\Sigma^* = \bigcup_{n \geq 0} \Sigma^n$
- $\Sigma \subseteq \Sigma^*$;
- $\emptyset^* = \{\lambda\}$;
- se $\Sigma \subset \Gamma$ então $\Sigma^* \subset \Gamma^*$;
- se $\Sigma \neq \emptyset$ então Σ^* é infinito;

Propriedades da Concatenação

UNIVERSIDADE DE ÉVORA

Sejam $x, y, z \in \Sigma^*$.

- (associativa) $x\,(yz) = (xy)\,z$;
- (elemento neutro) $\lambda x = x\lambda = x$;
- (**não** comutativa) $xy \neq yx$, em geral;
- (aditiva) $|xy| = |x| + |y|$;
- (unicidade) cada palavra só pode ser escrita de uma *única* forma como concatenação de símbolos de Σ;

UNIVERSIDADE DE ÉVORA

Sejam a, b, u, v palavras de Σ^*.

1. subpalavra: $a < b$ se existem $u, v \neq \lambda$ tais que $b = uav$;
2. prefixo: $a <_P b$ se existe $v \neq \lambda$ tal que $b = av$;
3. sufixo: $a <_S b$ se existe $u \neq \lambda$ tal que $b = ua$;
4. lexicográfica (supondo que $\Sigma = \{s_1 < s_2 < \cdots < s_n\}$): $a <_L b$ se $a <_P b$ ou, sendo $a = cs_iy$ e $b = cs_jz$, $s_i < s_j$;
5. mista: $a <_M b$ se $|a| < |b|$ ou $a <_L b$;

Aqui $u, v \neq \lambda$ porque esta é a ordem estrita ($<$). Os casos $u, v = \lambda$ são considerados na ordem lata ($\leq$).

As relações anteriores com igualdade: $x \leq y$ se $x = y$ ou $x < y$ e analogamente para $\leq_P, \leq_S, \leq_L, \leq_M$.

Sucessor

UNIVERSIDADE DE ÉVORA

Seja $\Sigma = \{s_1 < s_2 < \cdots < s_n\}$ um alfabeto ordenado.

A função sucessor (em Σ^*), $\sigma : \Sigma^* \to \Sigma^*$, fica definida por:

1. $\sigma(\lambda) = s_1$;
2. $\sigma(xs_i) = xs_{i+1}$ se $i < n$;
3. $\sigma(xs_n) = \sigma(x)\, s_1$

O sucessor enumera Σ^*:

$$\begin{array}{cccccccc} 0 & 1 & 2 & \ldots & n & n+1 & n+2 & \ldots \\ \lambda & s_1 & s_2 & \ldots & s_n & s_1 s_1 & s_1 s_2 & \ldots \end{array}$$

Fazendo $0 < 1$, fica

$$\begin{array}{rccccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & \ldots \\ \{0,1\}^* : & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \ldots \\ & \lambda & 0 & 1 & 00 & 01 & 10 & 11 & \ldots \end{array}$$

Usando a ordem $<_M$ enumeramos as palavras.

Teorema (Enumeração)

Seja $x_0, x_1, \ldots$ a enumeração de Σ^* e $x \in \Sigma^*$.

1. $x <_M \sigma(x)$;
2. *se* $x \neq \lambda$ *então* $x = \sigma(y)$ *para algum* y;
3. *para qualquer* $i \geq 0$, $\sigma(x_i) = x_{i+1}$;

Uma linguagem é um conjunto de palavras (sobre um certo alfabeto Σ). Isto é, A é uma linguagem (sobre Σ) se

$$A \subseteq \Sigma^*$$

Se A for uma linguagem, $|A|$ é o número de palavras em A;

- $\{0, 1\}$, $\{0^0, 0^1, 0^2, \ldots\}$;
- as palavras do português;
- os programas de uma linguagem de programação;

Diagrama de uma linguagem

UNIVERSIDADE DE ÉVORA

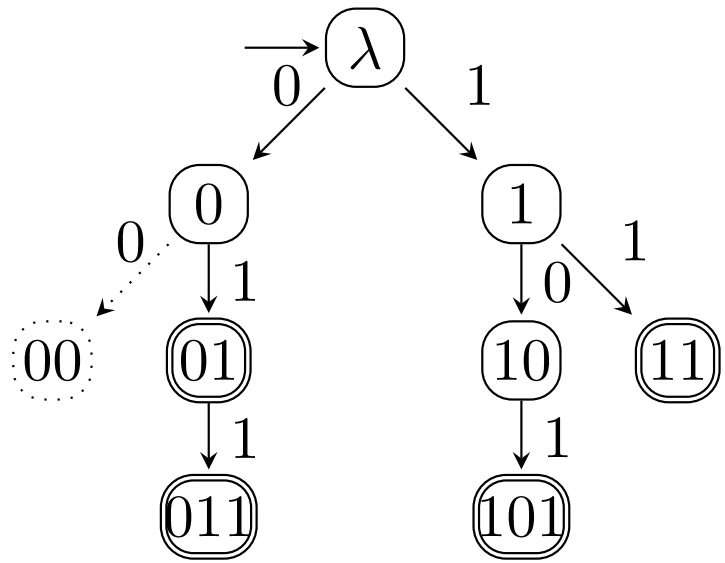
Podemos associar uma árvore a uma linguagem:

As palavras da linguagem

$$\{01, 011, 11, 101\}$$

são assinaladas com um círculo duplo.

Cada vértice corresponde a uma palavra e cada aresta a uma letra.



Operações com linguagens

UNIVERSIDADE DE ÉVORA

Sejam A, B linguagens sobre Σ.

1. união: $A \cup B = \{x \ : \ x \in A \vee x \in B\}$;
2. intersecção: $A \cap B = \{x \ : \ x \in A \wedge x \in B\}$;
3. subtracção: $A \setminus B = \{x \ : \ x \in A \wedge x \notin B\}$;
4. complemento: $\overline{A} = \Sigma^* \setminus A$;
5. concatenação: $AB = \{ab \ : \ a \in A \wedge b \in B\}$;
6. potência: $A^0 = \{\lambda\}$; $A^{n+1} = AA^n$;
7. fecho(s): $A^* = A^0 \cup A^1 \cup A^2 \cup \cdots$; $A^+ = AA^*$;
8. inversão: $A^R = \{a^R \ : \ a \in A\}$;

UNIVERSIDADE DE ÉVORA

Alfabetos, Palavras e Linguagens

Expressões Regulares

UNIVERSIDADE DE ÉVORA

Seja Σ um alfabeto. Uma linguagem regular sobre Σ define-se por:

base $\emptyset$, $\{\lambda\}$ e $\{s\}$, para cada $s \in \Sigma$, são linguagens regulares;

passo se A e B forem linguagens regulares, $A \cup B$, AB e A^* são linguagens regulares;

fecho X é uma linguagem regular se e só se pode ser construído através de um número finito de aplicações do passo a partir de conjuntos da base;

- $\{001, 110\} = \{0\}\{0\}\{1\} \cup \{1\}\{1\}\{0\}$;
- qualquer conjunto finito é uma linguagem regular;

Seja Σ um alfabeto.

Uma expressão regular sobre Σ define-se por:

- base $\emptyset$, λ e s, para cada $s \in \Sigma$, são expressões regulares;
- passo se a e b forem expressões regulares, $a \cup b$, ab e a^* são expressões regulares;
- fecho x é uma expressão regular se e só se pode ser construído através de um número finito de aplicações do passo a partir das expressões da base;

Também usamos $x^+ = xx^*$.

Precedência

UNIVERSIDADE DE ÉVORA

A escrita completa de uma expressão regular pode ser confusa:

$$(0^* \cup (10\,(0^*)))\,1\,(0^*)\,(((0\,(1^*)) \cup (1^*)))$$

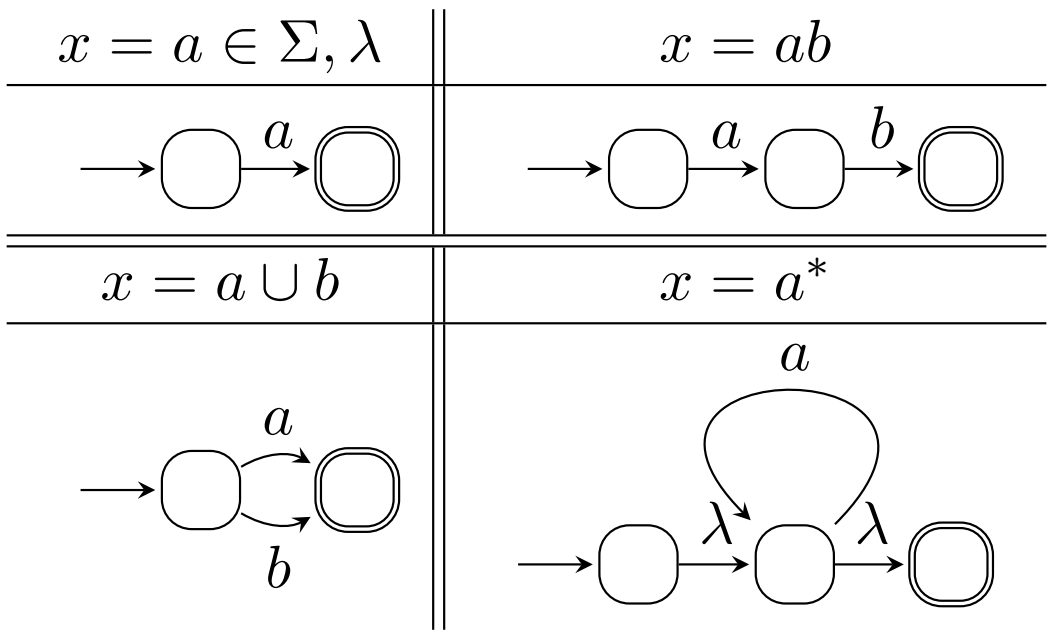
Para simplificar a escrita usamos as regras seguintes:

1. * *tem precedência sobre* $\cdot$ *e* $\cup$*;*
2. $\cdot$ *tem precedência sobre* $\cup$*;*

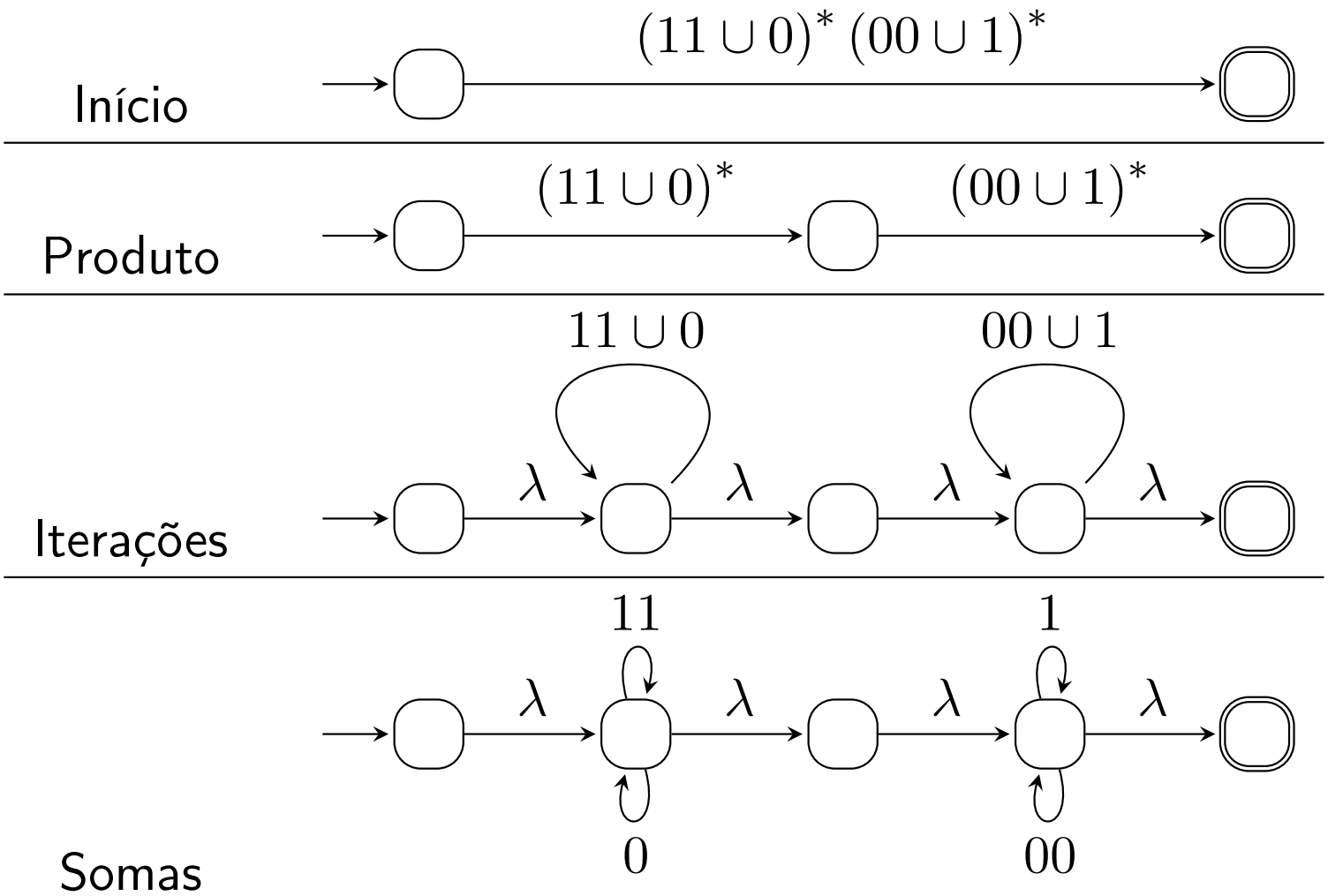
A expresão anterior fica reduzida a

$$(0^* \cup 100^*)\,10^*\,(01^* \cup 1^*)$$

O diagrama da expressão regular x, $\mathcal{G}(x)$, é um digrafo definido por:



Exemplo


Início
$(11 \cup 0)^* (00 \cup 1)^*$
Produto
$(11 \cup 0)^*$
$(00 \cup 1)^*$
Iterações
$11 \cup 0$
$00 \cup 1$
λ
λ
λ
λ
Somas
11
1
λ
λ
λ
λ
0
00

UNIVERSIDADE DE ÉVORA

Teorema (Arestas λ)

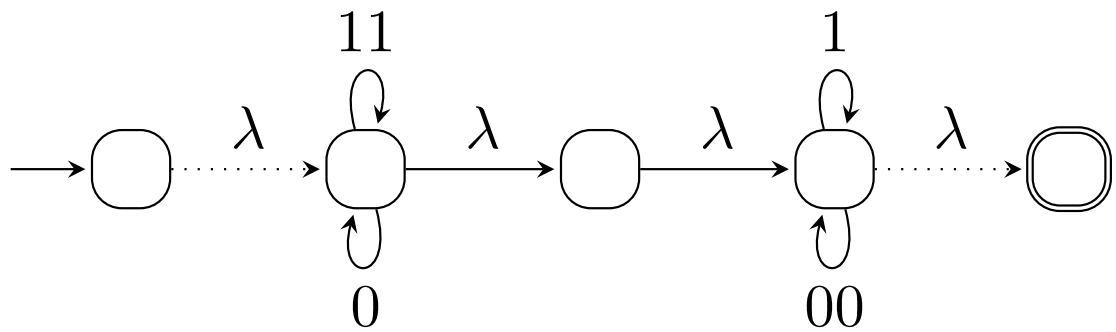
A aresta $\alpha \xrightarrow{\lambda} \beta$ *pode ser eliminada se*

- α *não é final e não saem mais arestas de* α *ou*
- β *não é inicial e não entram mais arestas em* β*;*

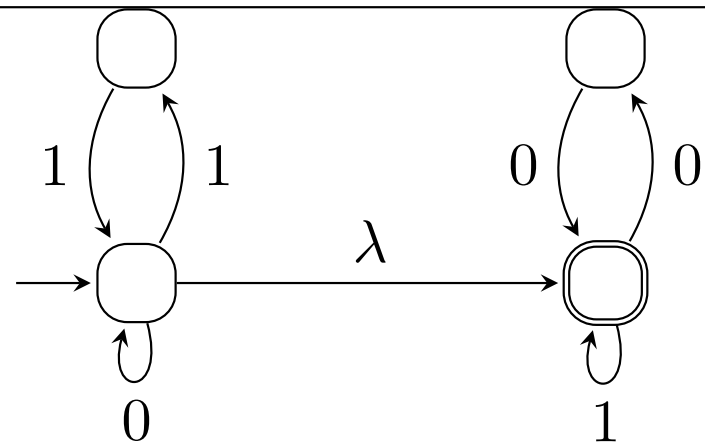
Dito de outra forma, estas arestas não podem ser eliminadas se

- α é final ou saem mais arestas de α e
- β é inicial ou entram mais arestas em β

Exemplo



$(11 \cup 0)^* (00 \cup 1)^*$



Eliminam-se as arestas-λ do princípio e do fim e usa-se $\lambda\lambda = \lambda$.

A linguagem representada por uma expressão regular é:

$$\begin{aligned}
\mathcal{L}(\emptyset) &= \emptyset \\
\mathcal{L}(\lambda) &= \{\lambda\} \\
\mathcal{L}(s) &= \{s\} \quad s \in \Sigma \\
\mathcal{L}(a \cup b) &= \mathcal{L}(a) \cup \mathcal{L}(b) \\
\mathcal{L}(ab) &= \mathcal{L}(a)\,\mathcal{L}(b) \\
\mathcal{L}(a^*) &= \mathcal{L}(a)^*
\end{aligned}$$

UNIVERSIDADE DE ÉVORA

1. qualquer expressão regular representa uma linguagem regular;
2. qualquer linguagem regular é representada por uma expressão regular;

Inteiros em binário, sem sinal e sem 0 à esquerda:

$$\begin{aligned}\{0,1,10,11,100,101,\ldots\} &= \{0\} \cup \{1\}\{\lambda,0,1,00,01,000,\ldots\} \\ &= \{0\} \cup \{1\}\{0,1\}^* \\ &= \mathcal{L}\left(0 \cup 1\left(0 \cup 1\right)^*\right)\end{aligned}$$

Equivalência

Duas expressões regulares, x e y são equivalentes, $x \equiv y$, se representam a mesma linguagem:

$$\mathcal{L}(x) = \mathcal{L}(y)$$

$$0^*1 \cup \emptyset \equiv 0^*1$$

Usamos "$x = y$" em vez de "$x \equiv y$" para indicar que as expressão regular x e y são equivalentes. Mas é preciso cautela: $0^*1 \cup \emptyset$ e 0^*1 são expressões regulares ***diferentes****, embora* ***equivalentes****.*

Propriedades da soma e produto de expressões regulares

UNIVERSIDADE DE ÉVORA

$$x \cup (y \cup z) = (x \cup y) \cup z \qquad x\,(yz) = (xy)\,z$$

$$x \cup \emptyset = \emptyset \cup x = x \qquad x\lambda = \lambda x = x$$

$$x\emptyset = \emptyset x = \emptyset$$

$$x \cup y = y \cup x$$

$$x \cup x = x$$

$$x\,(y \cup z) = xy \cup xz \qquad (x \cup y)\,z = xz \cup yz$$

Propriedades da iteração de expressões regulares

UNIVERSIDADE DE ÉVORA

$$\emptyset^* = \lambda \qquad \lambda^* = \lambda$$

$$(x^*)^* = x^* \qquad x^* = \lambda \cup xx^*$$

$$x\,(yx)^* = (xy)^*\,x$$

$$\begin{aligned}
(x \cup y)^* &= (x^* \cup y)^* \\
&= x^*\,(x \cup y)^* \\
&= (x \cup yx^*)^* \\
&= (x^*y^*)^* \\
&= (x^*y)^*\,x^* \\
&= x^*\,(yx^*)^*
\end{aligned}$$

Exemplo – simplificação de expressões regulares

UNIVERSIDADE DE ÉVORA

simplificar (tornar menos "profunda" e menos "comprida")

$$a^*(b \cup (a^*b^*)^*)aa^*(ba^*)^*b \stackrel{?}{=} (a \cup b)^*a(a \cup b)^*b$$

$$\begin{aligned}
a^*(b \cup (a^*b^*)^*)aa^*(ba^*)^*b \\
= a^*(b \cup (a \cup b)^*)aa^*(ba^*)^*b \\
= a^*(a \cup b)^*aa^*(ba^*)^*b \\
= (a \cup b)^*aa^*(ba^*)^*b \\
= (a \cup b)^*a(a \cup b)^*b \\
\stackrel{?}{=} b^*a(b \cup a)^*b
\end{aligned}$$

$$(x^*y^*)^* = (x \cup y)^*$$

$y \cup (x \cup y)^* = (x \cup y)^*$ (porquê?)

$$x^*(x \cup y)^* = (x \cup y)^*$$

$$x^*(yx^*)^* = (x \cup y)^*$$

aceitável... continuando:

Exemplo – simplificação de expressões regulares (cont.)

UNIVERSIDADE DE ÉVORA

$$
\begin{aligned}
&= (a \cup b)^* a (a \cup b)^* b && x \cup y = y \cup x \\
&= (b \cup a)^* a (a \cup b)^* b && (x \cup y)^* = x^* (y x^*)^* \\
&= b^* (a b^*)^* a (a \cup b)^* b && (xy)^* x = x (yx)^* \\
&= b^* a (b^* a)^* (a \cup b)^* b && x \cup y = y \cup x \\
&= b^* a (b^* a)^* (b \cup a)^* b && (x \cup y)^* = (x^* y)^* x^* \\
&= b^* a (b^* a)^* (b^* a)^* b^* b && x^* x^* = x^* \text{(porquê?)} \\
&= b^* a (b^* a)^* b^* b && (x^* y)^* x^* = (x \cup y)^* \\
&= b^* a (b \cup a)^* b
\end{aligned}
$$