# Propriedades das expressões regulares (1)

$$\emptyset u = u\emptyset = \emptyset \qquad \lambda u = u\lambda = u$$

$$(uv)w = u(vw)$$

$$u \cup v = v \cup u \qquad u \cup \emptyset = u$$

$$(u \cup v) \cup w = u \cup (v \cup w) \qquad u \cup u = u$$

$$u(v \cup w) = uv \cup uw$$

$$(u \cup v)w = uw \cup vw$$

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

$$u^* = (u^*)^* \qquad u^* = \lambda \cup uu^*$$

$$(uv)^* u = u(vu)^*$$

# Propriedades das expressões regulares (2)

$$
\begin{aligned}
(u \cup v)^* &= (u^* \cup v)^* \\
&= u^*(u \cup v)^* \\
&= (u \cup vu^*)^* \\
&= (u^* v^*)^* \\
&= (u^* v)^* u^* \\
&= u^*(vu^*)^*
\end{aligned}
$$

# Simplificação de expressões regulares (1)

## Exemplo

$$
\begin{aligned}
& 0^*(1 \cup (0^*1^*)^*)00^*(10^*)^*0 \\
= \quad & 0^*(1 \cup (0 \cup 1)^*)00^*(10^*)^*0 \\
= \quad & 0^*(0 \cup 1)^*00^*(10^*)^*0 \\
= \quad & (0 \cup 1)^*00^*(10^*)^*0 \\
= \quad & (0 \cup 1)^*0(0 \cup 1)^*0
\end{aligned}
$$

**Justificação**

$(u^*v^*)^* = (u \cup v)^*$

$v \cup (u \cup v)^* = (u \cup v)^*$
(porquê?)

$u^*(u \cup v)^* = (u \cup v)^*$

$u^*(vu^*)^* = (u \cup v)^*$

Já está numa forma aceitável, mas pode-se continuar

# Simplificação de expressões regulares (2)

## Exemplo (continuando...)

| | Justificação |
|---|---|
| $= (0 \cup 1)^*0(0 \cup 1)^*0$ | |
| | $u \cup v = v \cup u$ |
| $= (1 \cup 0)^*0(0 \cup 1)^*0$ | |
| | $(u \cup v)^* = u^*(vu^*)^*$ |
| $= 1^*(01^*)^*0(0 \cup 1)^*0$ | |
| | $(uv)^*u = u(vu)^*$ |
| $= 1^*0(1^*0)^*(0 \cup 1)^*0$ | |
| | $u \cup v = v \cup u$ |
| $= 1^*0(1^*0)^*(1 \cup 0)^*0$ | |
| | $(u \cup v)^* = (u^*v)^*u^*$ |
| $= 1^*0(1^*0)^*(1^*0)^*1^*0$ | |
| | $u^*u^* = u^*$ (porquê?) |
| $= 1^*0(1^*0)^*1^*0$ | |
| | $(u^*v)^*u^* = (u \cup v)^*$ |
| $= 1^*0(1 \cup 0)^*0$ | |

# Autómatos finitos deterministas

Um autómato finito determinista (AFD) é um tuplo
$M = (Q, \Sigma, \delta, q_0, F)$ onde

$Q$ é um conjunto finito de **estados**

$\Sigma$ é um conjunto finito de símbolos (**alfabeto**)

$\delta$ é a **função de transição**, uma função total de $Q \times \Sigma$ em $Q$

$q_0 \in Q$ é o **estado inicial** do autómato

$F \subseteq Q$ é o conjunto dos **estados de aceitação**

# Configuração e computação

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um AFD

A configuração de um AF é um par $[q, w] \in Q \times \Sigma^*$, onde $q$ é o estado corrente do autómato e $w$ é a parte da palavra ainda por processar

A computação de um AFD $M$ para a palavra $w = a_1 a_2 \dots a_n \in \Sigma^*$ é a sequência de configurações

$$[s_0, a_1 a_2 \dots a_n] \vdash_M [s_1, a_2 \dots a_n] \vdash_M \cdots \vdash_M [s_n, \lambda]$$

com

$$s_0 = q_0 \quad \text{e} \quad s_i = \delta(s_{i-1}, a_i)$$

para $i > 0$

# Função de transição estendida

A função de transição estendida $\hat{\delta} : Q \times \Sigma^* \to Q$ de um AFD é definida por

$$\begin{array}{rcl} \hat{\delta}(q, \lambda) & = & q \\ \hat{\delta}(q, a) & = & \delta(q, a) \\ \hat{\delta}(q, wa) & = & \delta(\hat{\delta}(q, w), a) \end{array}$$

Para um AFD,

$$\hat{\delta}(q_0, w) = q \quad \text{sse} \quad [q_0, w] \vdash_M^* [q, \lambda]$$

# Linguagem reconhecida

Uma palavra $w$ é aceite pelo AFD sse

$$\hat{\delta}(q_0, w) \in F$$

A linguagem reconhecida (ou aceite) por $M$ é o conjunto das palavras aceites por $M$

$$L(M) = \{w \mid \hat{\delta}(q_0, w) \in F\}$$

Dois autómatos finitos são equivalentes se reconhecem a mesma linguagem

# Autómatos finitos não deterministas (1)

Um autómato finito não determinista é um tuplo
$M = (Q, \Sigma, \delta, q_0, F)$ onde

$Q$ é um conjunto finito de **estados**

$\Sigma$ é um conjunto finito de símbolos (**alfabeto**)

$\delta$ é a **função de transição**, uma função total de $Q \times \Sigma$ em $\mathcal{P}(Q)$

$q_0 \in Q$ é o **estado inicial** do autómato

$F \subseteq Q$ é o conjunto dos **estados de aceitação**

Qualquer autómato finito determinista é um autómato finito não determinista.

# Autómatos finitos não deterministas (2)

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um autómato finito não determinista

Uma palavra $w$ é aceite por $M$ se existe uma computação que termina num estado de aceitação depois de terem sido processados todos os símbolos de $w$

$$[q_0, w] \vdash_M^* [q_i, \lambda], \text{ onde } q_i \in F$$

A linguagem reconhecida por $M$ é o conjunto das palavras aceites por $M$

$$L(M) = \left\{ w \;\middle|\; \begin{array}{l} \text{existe uma computação} \\ [q_0, w] \vdash_M^* [q_i, \lambda] \text{ em que } q_i \in F \end{array} \right\}$$