

AUTÔMATOS FINITOS com SAÍDA

➤ AF com Saída

◆ O conceito básico de AF

- ☐ aplicações restritas
- ☐ saída
 - * limitada à lógica binária aceita/rejeita

AUTÔMATOS FINITOS com SAÍDA

➤ AF com Saída

♦ Geração de uma palavra de saída

- *estende* a definição de AF
- *não altera* a capacidade de *reconhecimento*
 - * reconhece a classe de linguagens *regulares*
- *saída*
 - * *não* pode ser *lida*
 - * *não* pode ser usada como *memória auxiliar*
- as *saídas* podem ser associadas
 - * às *transições* - Máquina de Mealy
 - * aos *estados* - Máquina de Moore

AUTÔMATOS FINITOS com SAÍDA

➤ AF com Saída

◆ Saída

- definida sobre um alfabeto especial
 - * *alfabeto de saída*
 - * pode ser igual ao alfabeto de entrada
- **saída**
 - * fita de saída **independente** da fita de **entrada**
- cabeça da fita de saída
 - * move **uma** célula para a **direita**, a **cada símbolo gravado**
- **resultado** do processamento
 - * **estado final** (condição de aceita/rejeita) + info (fita de **saída**)

AUTÔMATOS FINITOS com SAÍDA

➤ Máquina de Mealy

◆ Para cada transição

- * gera uma palavra de saída
- * pode ser vazia

◆ *Definição. Máquina de Mealy*

- 6-upla $M = (\Sigma, Q, \delta, q_0, F, \Delta)$
- Σ alfabeto de símbolos de entrada
- Q conjunto finito de estados
- δ função programa ou de transição
 - * $\delta: Q \times \Sigma \rightarrow Q \times \Delta^*$
 - * função parcial
- q_0 estado inicial tq $q_0 \in Q$
- F conjunto de estados finais tq $F \subseteq Q$
- Δ alfabeto de símbolos de saída

AUTÔMATOS FINITOS com SAÍDA

➤ Máquina de Mealy

♦ Máquina de Mealy \times AFD

- Σ , Q , F e q_0 são como no AFD

♦ Processamento para uma entrada w

- sucessiva aplicação da função programa
- para cada símbolo de w , da esquerda para a direita ...
- até ocorrer uma condição de parada

♦ Palavra vazia como saída

- nenhuma gravação é realizada >> não move a cabeça da fita de saída
- se todas as transições geram saída vazia
 - * Máq. de Mealy processa como um AFD

AUTÔMATOS FINITOS com SAÍDA

➤ Máquina de Mealy

◆ *Exemplo: programa × usuário*

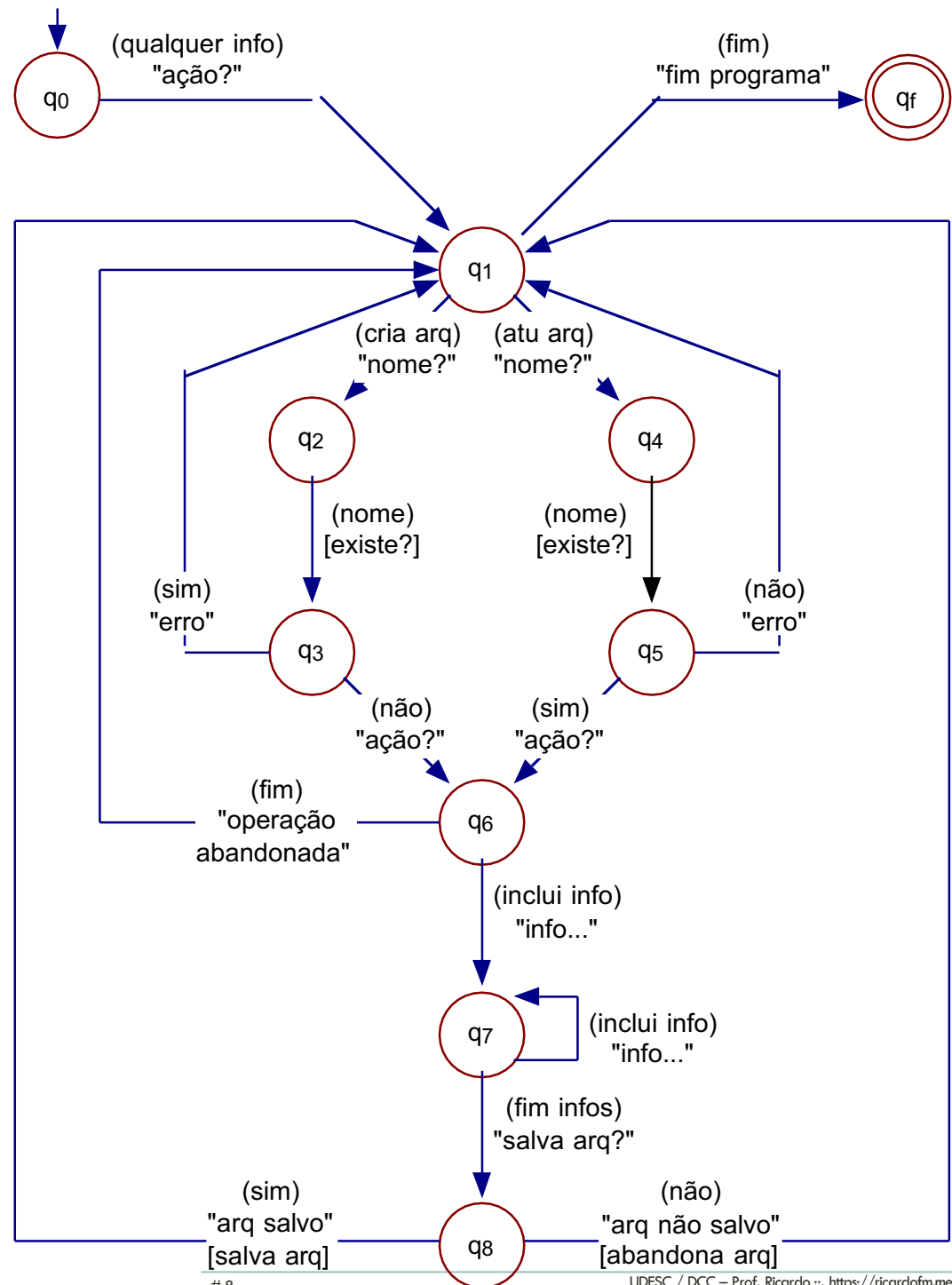
- cria e atualiza arquivos
- (...) entrada fornecida pelo usuário
- "... " saída gerada pelo programa
- [...] ação interna ao programa
- (...) resultado de uma ação interna ao programa
 - * entrada no diagrama
- $M = (\Sigma, \{q_0, q_1, \dots, q_8, q_f\}, \delta, q_0, \{q_f\}, \Delta)$
 - * $\Sigma = \Delta$
 - * símbolos válidos no diálogo

AUTÔMATOS FINITOS com SAÍDA

➤ Máquina de Mealy

Exemplo: programa × usuário

- cria e atualiza arquivos
- (...) **entrada** fornecida pelo usuário
- "..." **saída** gerada pelo programa
- [...] **ação interna** ao programa
- (...) **resultado de uma ação interna** ao programa
 - * entrada no diagrama
- $M = (\Sigma, \{q_0, q_1, \dots, q_8, q_f\}, \delta, q_0, \{q_f\}, \Delta)$
 - * $\Sigma = \Delta$
 - * **símbolos** válidos no **diálogo**



AUTÔMATOS FINITOS com SAÍDA

➤ Máquina de Moore

- para cada estado da máquina
 - * gera uma palavra de saída
 - * pode ser vazia
- possui uma função de saída

♦ *Máquina de Moore*

- 7-upla $M = (\Sigma, Q, \delta, q_0, F, \Delta, \delta_s)$
- Σ alfabeto de símbolos de entrada
- δ função programa ou de transição
 - * $\delta: Q \times \Sigma \rightarrow Q$
 - * função parcial
- Q conjunto finito de estados
- q_0 estado inicial tq $q_0 \in Q$
- F conjunto de estados finais tq $F \subseteq Q$
- Δ alfabeto de símbolos de saída
- δ_s função de saída
 - * $\delta_s: Q \rightarrow \Delta^*$
 - * é total

AUTÔMATOS FINITOS com SAÍDA

➤ Moore × AFD × Mealy

- Σ , Q , δ , F e q_0 são como no AFD
- Δ é como na Máquina de Mealy

◆ Processamento para uma entrada w

- sucessiva aplicação da função programa
- para cada símbolo de w
- da esquerda para a direita
- até ocorrer uma condição de parada

◆ Palavra vazia como saída

- nenhuma gravação é realizada
 - * não move a cabeça da fita de saída
- se todos os estados geram saída vazia
 - * Máquina de Moore processa como um AFD

AUTÔMATOS FINITOS com SAÍDA

♦ *Exemplo: Analisadores Léxicos*

- p/ compiladores ou tradutores de linguagens
- analisador léxico
 - * basicamente é um AF
 - * em geral, determinístico
- identifica os componentes básicos da linguagem
 - * números, identificadores, separadores, etc.
- *estado final*
 - * associado a cada unidade léxica identificada
 - * *saída*: descreve ou codifica a unidade léxica
- *estado não-final*
 - * *saída*: palavra vazia

AUTÔMATOS FINITOS com SAÍDA

Equivalência Moore \times Mealy

♦ Equivalência

- ☐ *não* é *válida* para a *entrada vazia*
- ☐ Máquina de *Moore*
 - * gera a *palavra correspondente ao estado inicial*
- ☐ Máquina de *Mealy*
 - * *não* gera saída
 - * não executa transição alguma
- ☐ demais casos
 - * *equivalência*

AUTÔMATOS FINITOS com SAÍDA

Equivalência Moore \times Mealy

♦ *Teorema: Moore \rightarrow Mealy*

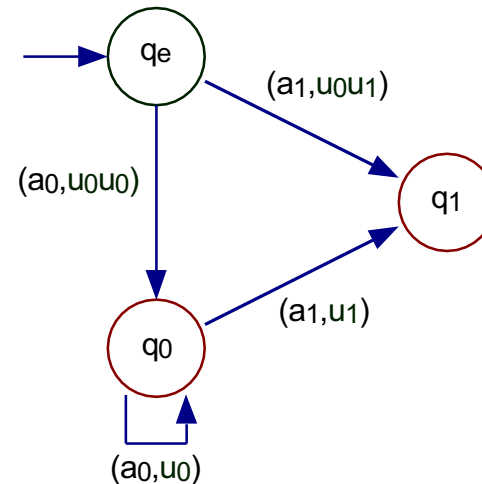
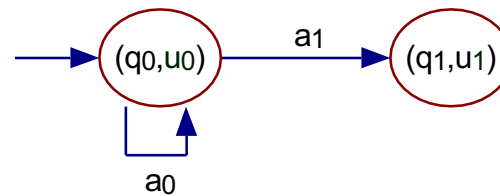
- qq Máq de Moore
- pode ser simulada por uma Máq de Mealy
- para *entradas não-vazias*

♦ *Correspondente Máq de Mealy*

- simulação da saída do estado inicial de Moore?
 - * introduz um novo estado q_e
 - * referenciado somente na primeira transição a ser executada
 - * saída referente ao estado inicial de Moore

AUTÔMATOS FINITOS com SAÍDA

Equivalência Moore \times Mealy



AUTÔMATOS FINITOS com SAÍDA

Equivalência Moore \times Mealy

Prova

□ $MO = (\Sigma, Q, \delta_{MO}, q_0, F, \Delta, \delta_S)$, Máq de Moore qq

□ seja a Máquina de Mealy

$$ME = (\Sigma, Q \cup \{q_e\}, \delta_{ME}, q_e, F, \Delta)$$

$$\square \square \delta_{ME}(q_e, a) =$$

$$(\delta_{MO}(q_0, a), \delta_S(q_0)\delta_S(\delta_{MO}(q_0, a)))$$

$$\square \square \delta_{ME}(q, a) = (\delta_{MO}(q, a), \delta_S(\delta_{MO}(q, a)))$$

□ ou seja, δ_{ME} construída a partir de

* δ_{MO}

* δ_S

AUTÔMATOS FINITOS com SAÍDA

Equivalência Moore \times Mealy

◆ *Teorema.* Mealy \rightarrow Moore

- Qq Máquina de Mealy
- pode ser simulada por uma Máquina de Moore

◆ Correspondente Máq de Moore

- transições com saídas diferentes que atingem um mesmo estado
 - * um estado para cada símbolo de saída
 - * em geral, aumenta o número de estados