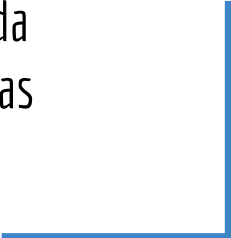




Teoría de Lenguajes

Autómatas con Salida
Autómatas de 2 cintas



Autómatas con Salida

- con los Autómatas Finitos M se verifica si una tira $x \in L(M)$

Reconocedores

- con los Autómatas con Salida, se genera una nueva tira $x \rightarrow y$ que incluso puede pertenecer a otro alfabeto

Traductores

Autómatas con Salida (cont.)

- en cada momento, los estados y el símbolo de la entrada, determinan el símbolo que se genera y si existe o no un cambio de estado
- van a existir dos “*buffers*”, uno de entrada (como en los AFs) y otro de salida
- Modelos
 - Mealy
 - Moore

Autómata de Mealy

$M: (Q, \Sigma, \Delta, \delta, \lambda, q_0)$

- Q : conjunto de estados
- Σ : alfabeto de entrada
- Δ : alfabeto de salida
- δ : función de transición / $\delta: Q \times \Sigma \rightarrow Q$ (son deterministas)
- λ : función de salida / $\lambda: Q \times \Sigma \rightarrow \Delta$
- q_0 : estado inicial / $q_0 \in Q$

*Se genera la salida inmediatamente después de computar la entrada,
símbolo a símbolo*

Autómata de Mealy

Extensión para manejar strings

$$\delta^{\wedge}: Q \times \Sigma^* \rightarrow Q$$

$$\delta^{\wedge}(q, \varepsilon) = q \quad \forall q \in Q$$

$$\delta^{\wedge}(q, wa) = \delta(\delta^{\wedge}(q, w), a) \quad \forall q \in Q \quad a \in \Sigma \quad w \in \Sigma^*$$

$$\lambda^{\wedge}: Q \times \Sigma^* \rightarrow \Delta^*$$

$$\lambda^{\wedge}(q, \varepsilon) = \varepsilon \quad \forall q \in Q$$

$$\lambda^{\wedge}(q, wa) = \lambda^{\wedge}(q, w) \cdot \lambda(\delta^{\wedge}(q, w), a) \quad \forall q \in Q \quad a \in \Sigma \quad w \in \Sigma^*$$

Ejemplo Mealy

Dada una tira $x \in \Sigma^*$, siendo $\Sigma = \{0,1,2\}$, escribe **C** si el símbolo anterior es un 0 y **R** en cualquier otra caso

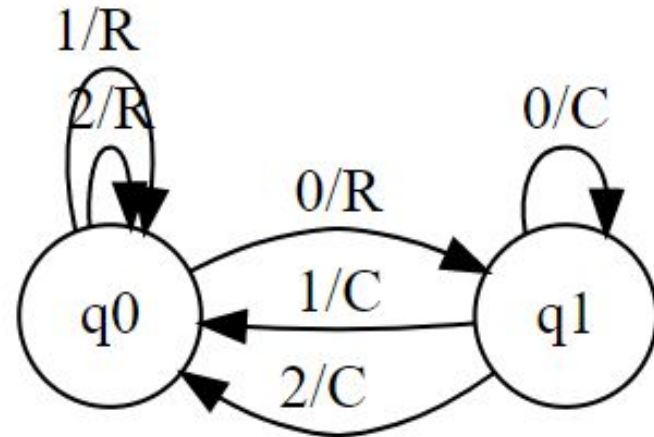
Ejemplos:

1202 \rightarrow RRRC

001 \rightarrow RCC

00010 \rightarrow RCCCR

222120 \rightarrow RRRRRR



Autómata de Moore

$M: (Q, \Sigma, \Delta, \delta, \lambda, q_0)$

- Q : conjunto de estados
- Σ : alfabeto de entrada
- Δ : alfabeto de salida
- δ : función de transición / $\delta: Q \times \Sigma \rightarrow Q$ (son deterministas)
- λ : función de salida / $\lambda: Q \rightarrow \Delta$
- q_0 : estado inicial / $q_0 \in Q$

Se genera la salida dependiendo del estado al cual se llegue después de computar la entrada, símbolo a símbolo

Autómata de Moore

Extensión para manejar strings

$$\delta^{\wedge}: Q \times \Sigma^* \rightarrow Q$$

$$\delta^{\wedge}(q, \varepsilon) = q \quad \forall q \in Q$$

$$\delta^{\wedge}(q, wa) = \delta(\delta^{\wedge}(q, w), a) \quad \forall q \in Q \quad a \in \Sigma \quad w \in \Sigma^*$$

$$\lambda^{\wedge}: Q \rightarrow \Delta^* \quad \text{entonces....}$$

$$\lambda': Q \times \Sigma^* \rightarrow \Delta^*$$

$$\lambda'(q, \varepsilon) = \varepsilon \quad \forall q \in Q$$

$$\lambda'(q, wa) = \lambda'(q, w) \cdot \lambda(\delta(\delta^{\wedge}(q, w), a)) \quad \forall q \in Q \quad a \in \Sigma \quad w \in \Sigma^*$$

Ejemplo Moore

Dada una tira $x \in \Sigma^*$, siendo $\Sigma = \{0,1,2\}$, escribe **C** si el símbolo anterior es un 0 y **R** en cualquier otra caso

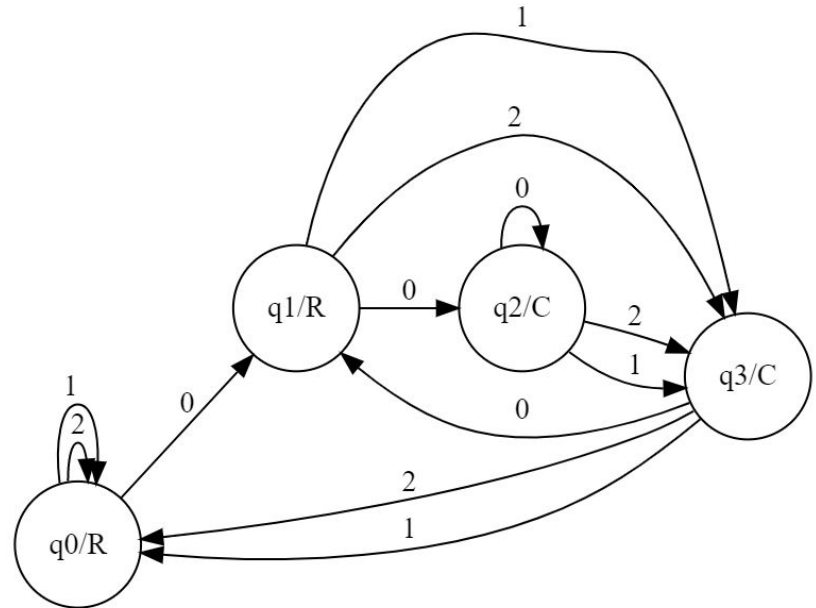
Ejemplos:

1202 \rightarrow RRRC

001 \rightarrow RCC

00010 \rightarrow RCCCR

222120 \rightarrow RRRRRR



Equivalencia Mealy & Moore

1) Mealy \Rightarrow Moore

$$Me : (Q, \Sigma, \Delta, \delta, \lambda, q_0) \rightarrow Mo : (Q', \Sigma, \Delta, \delta', \lambda', q'_0)$$

donde si se tiene $\delta(q, a) = p \wedge \lambda(q, a) = b$

se genera lo siguiente:

un estado $p^b \in Q' / \lambda'(p^b) = b \wedge \delta'(q^s, a) = p^b \quad \forall q^s, s \in \Delta$

Equivalencia Mealy & Moore

2) Moore \Rightarrow Mealy

$$Mo : (Q, \Sigma, \Delta, \delta, \lambda, q_0) \rightarrow Me : (Q, \Sigma, \Delta, \delta, \lambda', q_0)$$

donde si se tiene $\delta(q, a) = p \wedge \lambda(p) = b \quad q, p \in Q, a \in \Sigma, b \in \Delta$

se genera lo siguiente:

$$\lambda'(q, a) = b$$

(Notar que el δ es el mismo para ambos autómatas)

Equivalencia de autómatas con salida

➤ $q_1 \equiv q_2$ si $\lambda(q_1, x) = \lambda(q_2, x) \quad \forall x \in \Sigma^*$

de donde se deduce que si $q_1 \equiv q_2$ entonces $\delta^*(q_1, x) \equiv \delta^*(q_2, x)$

➤ $q_1 \equiv_n q_2$ si $\lambda(q_1, x) = \lambda(q_2, x) \quad \forall x \in \Sigma^*, |x| = n$

Equivalencia de autómatas con salida

Dadas $M_1 : (Q_1, \Sigma, \Delta, \delta_1, \lambda_1, q_0)$ y $M_2 : (Q_2, \Sigma, \Delta, \delta_2, \lambda_2, p_0)$

➤ $q \text{ E } p$ si $\lambda_1(q, x) = \lambda_2(p, x) \quad \forall x \in \Sigma^* \quad q \in Q_1 \quad p \in Q_2$

➤ $M_1 \text{ E } M_2 \Leftrightarrow \forall q \in Q_1, \exists p \in Q_2 / q \text{ E } p$

Minimización de autómatas con salida

- ❑ Dado un $M : (Q, \Sigma, \Delta, \delta, \lambda, q_0) \rightarrow M' : (Q', \Sigma, \Delta, \delta', \lambda', q'_0)$
- ❑ Se tratan de agrupar estados que sean equivalentes de longitud n
- ❑ Se basa en la misma idea del algoritmo de AFD
- ❑ El tema es ver cual es la partición inicial Π_0
- ❑ Π_0 se calcula de la siguiente manera:
 - ❑ $\forall p, q \in Q$, si se cumple $\lambda(p, a) = \lambda(q, a) \quad \forall a \in \Sigma$
entonces p y q van a estar en la misma clase en la Π_0
- ❑ Luego se aplica para cada conjunto de la Π_0 lo mismo que el algoritmo de minimización de AFD, es decir, $\delta(p, a)$ y $\delta(q, a) \quad \forall a \in \Sigma$ y $\forall p, q$ de cada clase

Minimización de autómatas con salida (cont.)

$M' : (Q', \Sigma, \Delta, \delta', \lambda', q'_0)$ queda determinado de la siguiente manera:

- $Q' : \{ C_k / \text{cada } C_k \text{ es una clase de la } \Pi_{\text{final}} \}$
- $q'_0 : C_k$ que contiene a q_0
- $\delta'(C_k, a) = C_j$ si $\delta(q, a) = p \quad \forall a \in \Sigma$, siendo $q \in C_k$ y $p \in C_j$
- $\lambda'(C_k, a) = \lambda(q, a) \quad \forall a \in \Sigma$, siendo $q \in C_k$

Otros formalismos...

- Extensión de Autómatas con Salida, que admiten transiciones epsilon
- Autómatas de 2 cintas, que computan pares de tiras de entrada $\langle w_1, w_2 \rangle$

Autómata de 2 cintas

Descripción formal del AFD de dos cintas

Un autómata finito determinista de dos cintas es una tupla $M = (Q_1, Q_2, \Sigma, \delta, F, q_0)$ dónde:

- Σ es el alfabeto del autómata
- $q_0 \in (Q_1 \cup Q_2)$ estado inicial del autómata
- $F \subseteq (Q_1 \cup Q_2)$ conjunto de estados finales del autómata
- $\delta : (Q_1 \cup Q_2) \times \Sigma \rightarrow (Q_1 \cup Q_2)$ función de transición
- $\delta' : (Q_1 \cup Q_2) \times \Sigma^* \times \Sigma^* \rightarrow (Q_1 \cup Q_2)$ función de transición extendida:
 - $\delta'(q_i, ax, y) = \delta'(\delta(q_i, a), x, y)$ si $q_i \in Q_1$
 - $\delta'(q_i, x, ay) = \delta'(\delta(q_i, a), x, y)$ si $q_i \in Q_2$
 - $\delta'(q_i, \epsilon, \epsilon) = q_i$

Un par de tiras $(v, w) \in L(M) \Leftrightarrow \delta'(q_0, v, w) \in F$.

Autómata de 2 cintas

Aplicación

$L = \{ \langle a^k b^p c^{t-1}, b^{k-1} a c^t \rangle, \text{ con } k, t > 0, p \geq 0 \}$ (Parcial 1 - 2019)

