Autómatas finitos determinísticos

Christian Sebastian Russo

16 de Septiembre de 2015

1. Autómatas finitos determinísticos
   - Definición
   - Conceptos
2. Ejercicio
   - Enunciado
   - Consideraciones
   - Solución
3. Bibliografía

Autómata finito determinístico - AFD

Se define mediante la 5-upla $< Q, \Sigma, \delta, q_0, F >$ donde

- Q es un conjunto finito de **estados**
- Σ es el conjunto finito de símbolos correspondiente al **alfabeto de entrada**
- $\delta : Q \times \Sigma \to Q$ es la **función de transición**, que indica las transiciones entre los distintos estados
- $q_0 \in Q$ es el **estado inicial**
- $F \subseteq Q$ es el conjunto de **estados finales**

Determinístico: para todo estado del autómata existe como máximo una transición definida para cada símbolo del alfabeto

Relación entre cadenas y autómatas

Función de transición generalizada

La función de transición propuesta anteriormente, puede extenderse para que en lugar de **símbolos** del alfabeto acepte **cadenas** en ese alfabeto, es decir: $\widehat{\delta} : Q \times \Sigma^* \to Q$

- $\widehat{\delta}(q, \lambda) = q$
- $\widehat{\delta}(q, xa) = \delta(\widehat{\delta}(q, x), a)$ con $x \in \Sigma^*$ y $a \in \Sigma$

Cadena aceptada por un AFD

Una cadena x es aceptada por un AFD $M =< Q, \Sigma, \delta, q_0, F >$ sii $\widehat{\delta}(q_0, x) \in F$, es decir, si la secuencia de transiciones correspondientes a los símbolos de la cadena x conduce desde el estado inicial a un estado final.

Ejercicio 1-c) (Práctica 1 - Teoría de lenguajes)

Construir un autómata finito para el siguiente lenguaje:
Cadenas sobre $\Sigma = \{0, 1\}$ con cantidad par de ceros y cantidad impar de unos.

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010
- Cantidad impar de ceros, cantidad par de unos.

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010
- Cantidad impar de ceros, cantidad par de unos.
  Ej: 011, 11011, 0

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010
- Cantidad impar de ceros, cantidad par de unos.
  Ej: 011, 11011, 0
- Cantidad impar de ceros, cantidad impar de unos.

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010
- Cantidad impar de ceros, cantidad par de unos.
  Ej: 011, 11011, 0
- Cantidad impar de ceros, cantidad impar de unos.
  Ej: 01, 1110, 0001

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010
- Cantidad impar de ceros, cantidad par de unos.
  Ej: 011, 11011, 0
- Cantidad impar de ceros, cantidad impar de unos.
  Ej: 01, 1110, 0001
- Cantidad par de ceros, cantidad impar de unos.

Existen cuatro posibles combinaciones de paridades de ceros y unos para un número binario:

- Cantidad par de ceros, cantidad par de unos.
  Ej: λ, 0011, 1111, 010010
- Cantidad impar de ceros, cantidad par de unos.
  Ej: 011, 11011, 0
- Cantidad impar de ceros, cantidad impar de unos.
  Ej: 01, 1110, 0001
- Cantidad par de ceros, cantidad impar de unos.
  Ej: 001, 111, 01000

De estas cuatro posibles combinaciones, nuestro lenguaje acepta sólo la última.

Podemos representar las cuatro distintas combinaciones mencionadas anteriormente, como cuatro estados distintos:

- $Par0Par1$: Cantidad par de ceros, cantidad par de unos.
- $Impar0Par1$: Cantidad impar de ceros, cantidad par de unos.
- $Impar0Impar1$: Cantidad impar de ceros, cantidad impar de unos.
- $Par0Impar1$: Cantidad par de ceros, cantidad impar de unos.

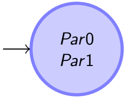
Estado Inicial del autómata:

Estado Inicial del autómata:

Al inicio tenemos la cadena λ: No tenemos ceros ni unos aún

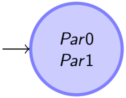
Estado Inicial del autómata:

Al inicio tenemos la cadena λ: No tenemos ceros ni unos aún



Estado Inicial del autómata:

Al inicio tenemos la cadena λ: No tenemos ceros ni unos aún



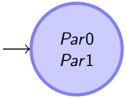
Estado Final del autómata:

Recordamos:

Una cadena x es aceptada por un AFD si la secuencia de transiciones correspondientes a los símbolos de x conduce desde el estado inicial a un estado final.

Estado Inicial del autómata:

Al inicio tenemos la cadena λ: No tenemos ceros ni unos aún



Estado Final del autómata:

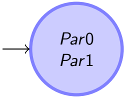
Recordamos:

Una cadena x es aceptada por un AFD si la secuencia de transiciones correspondientes a los símbolos de x conduce desde el estado inicial a un estado final.

En nuestro caso debemos aceptar las cadenas con cantidad par de ceros y cantidad impar de unos.

Estado Inicial del autómata:

Al inicio tenemos la cadena λ: No tenemos ceros ni unos aún



Estado Final del autómata:

Recordamos:

Una cadena x es aceptada por un AFD si la secuencia de transiciones correspondientes a los símbolos de x conduce desde el estado inicial a un estado final.

En nuestro caso debemos aceptar las cadenas con cantidad par de ceros y cantidad impar de unos. Las que cumplen esta condición son las que llegan al estado *Par0Impar1*

Solución

$$< Q, \Sigma, \delta, Par0Par1, F >$$

$< Q, \Sigma, \delta, \mathit{Par0Par1}, F >$
$Q = \{\mathit{Par0Par1}, \mathit{Par0Impar1}, \mathit{Impar0Impar1}, \mathit{Impar0Par1}\}$

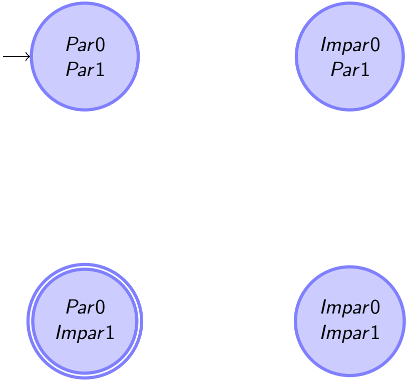
Solución

$$< Q, \Sigma, \delta, Par0Par1, F >$$
$$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$$
$$\Sigma = \{0, 1\}$$

$< Q, \Sigma, \delta, Par0Par1, F >$
$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$
$\Sigma = \{0, 1\}$
$F = \{Par0Impar1\}$

$< Q, \Sigma, \delta, Par0Par1, F >$
$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$
$\Sigma = \{0, 1\}$
$F = \{Par0Impar1\}$
δ:

$< Q, \Sigma, \delta, Par0Par1, F >$
$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$
$\Sigma = \{0, 1\}$
$F = \{Par0Impar1\}$
δ:



Solución

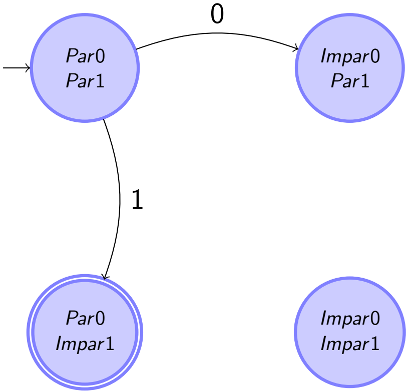
$< Q, \Sigma, \delta, Par0Par1, F >$

$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$

$\Sigma = \{0, 1\}$

$F = \{Par0Impar1\}$

δ:



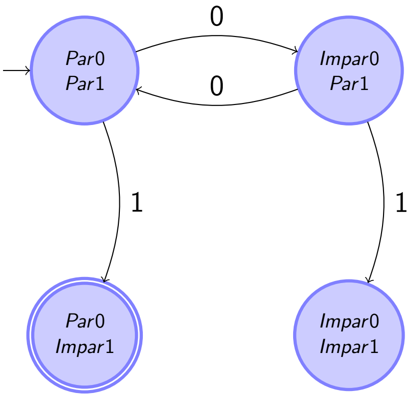
$< Q, \Sigma, \delta, Par0Par1, F >$

$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$

$\Sigma = \{0, 1\}$

$F = \{Par0Impar1\}$

δ:



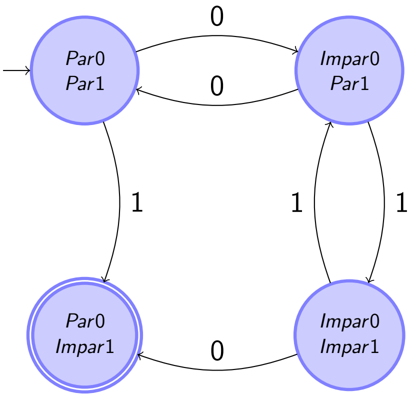
$< Q, \Sigma, \delta, Par0Par1, F >$

$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$

$\Sigma = \{0, 1\}$

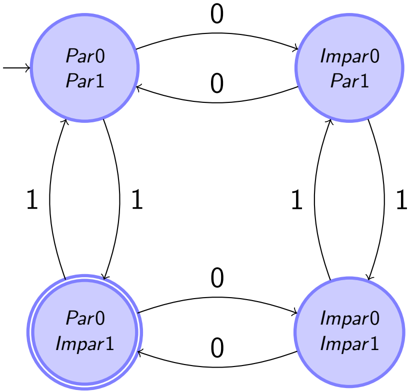
$F = \{Par0Impar1\}$

δ:



Solución

$< Q, \Sigma, \delta, Par0Par1, F >$
$Q = \{Par0Par1, Par0Impar1, Impar0Impar1, Impar0Par1\}$
$\Sigma = \{0, 1\}$
$F = \{Par0Impar1\}$
δ:



Bibliografía

Introduction to Automata theory, languages, and computation
Hopcroft, John E. y Ullman, Jeffrey D.