Lenguajes Formales, Autómatas y Computabilidad

Práctica 2: Autómatas finitos

Versión del 4 de septiembre de 2024

Ejercicio 1. Construir autómatas finitos para los siguientes lenguajes:

a. Cadenas sobre $\Sigma = \{0\}$ de longitud par.

b. Cadenas sobre $\Sigma = \{0, 1\}$ con cantidad par de ceros.

c. Cadenas sobre $\Sigma = \{0, 1\}$ con cantidad impar de unos.

d. Cadenas sobre $\Sigma = \{0, 1\}$ con cantidad par de ceros y cantidad impar de unos.

e. Cadenas sobre $\Sigma = \{0, 1\}$ que, interpretadas como un número binario, sean congruentes a cero módulo 5.[1]

Ejercicio 2. Construir autómatas finitos para los siguientes lenguajes sobre $\Sigma = \{0, 1\}$:

a. Cadenas que comiencen con 010.

b. Cadenas que terminen con 010.

c. Cadenas que contengan la subcadena 000.

d. Cadenas que no contengan la subcadena 000.

e. Cadenas que contengan la subcadena 000 exactamente una vez
(la cadena 0000 no pertenece a este lenguaje).

f. Cadenas que no contengan la subcadena 000 ni la 010.

Ejercicio 3. Construir autómatas finitos para los siguientes lenguajes:

a. Identificadores de cualquier longitud que comiencen con una letra o guión y contengan letras, dígitos o guiones.

b. Constantes enteras con signo.

c. Constantes enteras con signo opcional.

d. Constantes reales con signo. Ejemplos: +123.456, -55.0, +00.430.

e. Constantes reales con signo opcional y partes enteras y fraccionarias opcionales. Ejemplos: los anteriores más 123.456, -55., +.43.

f. Constantes reales con notación exponencial opcional. Ejemplos: los anteriores más -55.*E*5, +.43*E*-6.

[1] *Pista*: Pensar qué significa en términos aritméticos agregar un dígito al final de un número binario, y cómo afecta esto a la congruencia módulo 5.

Ejercicio 4. Dado un autómata finito para $\mathcal{L}$, indicar cómo construir autómatas finitos para los siguientes lenguajes. Indicar en cada caso si es necesario que el autómata de entrada sea determinístico o no, y de qué tipo es el autómata resultante.

a. $\mathcal{L}^{\mathrm{c}}$, el complemento de $\mathcal{L}$.

b. $\mathcal{L}^*$, la clausura de Kleene de $\mathcal{L}$.

c. $\mathcal{L}^{\mathrm{r}}$, la reversa de $\mathcal{L}$.

d. $\mathrm{Ini}(\mathcal{L}) = \{\alpha \mid \exists \beta \text{ tal que } \alpha\beta \in \mathcal{L}\}$, los prefijos de $\mathcal{L}$.

e. $\mathrm{Fin}(\mathcal{L}) = \{\alpha \mid \exists \gamma \text{ tal que } \gamma\alpha \in \mathcal{L}\}$, los sufijos de $\mathcal{L}$.

f. $\mathrm{Sub}(\mathcal{L}) = \{\alpha \mid \exists (\beta, \gamma) \text{ tales que } \gamma\alpha\beta \in \mathcal{L}\}$, las subcadenas de $\mathcal{L}$.

g. $\mathrm{Máx}(\mathcal{L}) = \{\alpha \in \mathcal{L} \mid \forall \omega \in \Sigma^+, \alpha\omega \notin \mathcal{L}\}$, las cadenas maximales de $\mathcal{L}$.

h. $\mathrm{Mín}(\mathcal{L}) = \{\alpha \in \mathcal{L} \mid \text{ningún prefijo propio de } \alpha \text{ pertenece a } \mathcal{L}\}$, las cadenas minimales de $\mathcal{L}$. Es decir, $\mathrm{Mín}(\mathcal{L}) = \{\alpha \in \mathcal{L} \mid \nexists (\omega_1, \omega_2) \text{ tales que } \alpha = \omega_1\omega_2 \wedge \omega_1 \in \mathcal{L} \wedge \omega_2 \neq \lambda\}$.

i. $\mathcal{L}_T = \{\alpha \in \Sigma^* \mid \exists (\omega_1 \in \mathcal{L}, \omega_2 \in \Sigma^*) \text{ tales que } \alpha = \omega_1\omega_2\} = \mathcal{L}.\Sigma^*$.

Ejercicio 5. Dados autómatas finitos para $\mathcal{L}_1$ y $\mathcal{L}_2$ indicar cómo construir autómatas finitos para los siguientes lenguajes, con las mismas consideraciones que en el ejercicio anterior:

a. $\mathcal{L}_1 \cup \mathcal{L}_2$ *b.* $\mathcal{L}_1 \cap \mathcal{L}_2$ *c.* $\mathcal{L}_1.\mathcal{L}_2$ *d.* $\mathcal{L}_1 \setminus \mathcal{L}_2$

Ejercicio 6. Demostrar que para todo autómata finito determínistico su relación de transición $\vdash$ cumple:

a. Determinismo: $\Big((q,\alpha) \overset{*}{\vdash} (r,\lambda) \wedge (q,\alpha) \overset{*}{\vdash} (s,\lambda)\Big) \Longrightarrow r = s$

b. Concatenación: $\Big((q,\alpha) \overset{*}{\vdash} (q_1,\lambda) \wedge (q_1,\beta) \overset{*}{\vdash} (r,\lambda)\Big) \Longrightarrow (q,\alpha\beta) \overset{*}{\vdash} (r,\lambda)$

c. Siempre toma un estado: $(q,\alpha\beta) \overset{*}{\vdash} (r,\lambda) \Longrightarrow \exists q_1 \Big((q,\alpha) \overset{*}{\vdash} (q_1,\lambda) \wedge (q_1,\beta) \overset{*}{\vdash} (r,\lambda)\Big)$

d. Linealidad: $(q,\alpha) \overset{n}{\vdash} (r,\lambda) \Longleftrightarrow |\alpha| = n$

e. Invariancia: $(q,\alpha) \overset{*}{\vdash} (q,\lambda) \Longrightarrow \forall i \in \mathbb{N}\Big((q,\alpha_i) \overset{*}{\vdash} (q,\lambda)\Big)$

Ejercicio 7. Dar un autómata finito determinístico que acepte todas las cadenas sobre el alfabeto $\{a, b, c\}$ que cumplan simultáneamente las siguientes reglas:

a. Cada a debe estar seguida inmediatamente de una b.

b. La cantidad de b debe ser par.

c. La cadena no debe terminar en c.

Ejercicio 8. Decimos que una subcadena de otra cadena es un grupo de repetición (o meseta) si todos sus símbolos son iguales y ninguno de los símbolos adyacentes a ella coincide con los que la forman. Por ejemplo, en la palabra *aaabbbbaaa* hay tres grupos de repetición (*aaa*, *bbb b* y *aaa*).

Se considera el lenguaje $\mathcal{L}$ sobre el alfabeto $\{a, b\}$ formado por las cadenas en las que, si existen grupos de repetición, su longitud es alternativamente par e impar. Es decir, la palabra *aabbbaaaab* pertenece al lenguaje $\mathcal{L}$, ya que esta formada por cuatro grupos de repetición de longitudes 2, 3, 4 y 1, mientras que la palabra *bbaa* no pertenece, al estar formada por dos grupos de repetición de longitudes 2 y 2.

Dar un autómata finito que acepte $\mathcal{L}$.