

**Ejercicio 1.** Dar un algoritmo que decida si dos expresiones regulares denotan el mismo lenguaje. Justificar la correctitud.

**Ejercicio 2.** Dar dos algoritmos distintos para determinar si el lenguaje aceptado por un autómata finito dado es el conjunto de todas las cadenas del alfabeto. Justificar cada uno.

**Ejercicio 3.** Dar un algoritmo que determine si un lenguaje regular dado es infinito. Justificar.

**Ejercicio 4.** ¿Cuántos autómatas finitos deterministas con dos estados pueden construirse sobre el alfabeto  $\{0, 1\}$ ?

**Ejercicio 5.** Sean  $L_1$  y  $L_2$  lenguajes regulares incluidos en  $\Sigma^*$ . Hacer un AFD con dos cintas de entrada que reconozca el lenguaje  $L = \{(u, v) : u \in L_1, v \in L_2, |u| = |v|\}$

**Ejercicio 6.** Decir Verdadero o Falso y justificar:

1. Para cada AF hay infinitos AFD que reconocen el mismo lenguaje.
2. Si  $L$  es libre de contexto, todo subconjunto de  $L$  es libre de contexto
3. Los autómatas finitos determinísticos reconocen una cadena de longitud  $n$  en exactamente  $n$  transiciones.
4. Los autómatas de pila determinísticos reconocen una cadena de longitud  $n$  en exactamente  $n$  transiciones.
5. Las máquinas de Turing determinísticas reconocen una cadena de longitud  $n$  en exactamente  $n$  transiciones.
6. Sea  $M$  un AFD y sea  $M^R$  el autómata que resulta de revertir función de transición.  $\mathcal{L}(M)$  intersección  $\mathcal{L}(M^R)$  es regular.

**Ejercicio 7.** Indicar Verdadero o Falso y justificar:

1. Toda función total de  $\mathbb{N}$  en  $\mathbb{N}$  es computable.
2. El conjunto de funciones parcialmente computables de  $\mathbb{N}$  en  $\mathbb{N}$  es c.e.

**Ejercicio 8.** Indicar V o F y justificar. Es decidible que

1. La intersección de dos conjuntos c.e. es un conjunto c.e.
2. La clausura de Kleene de un lenguaje c.e. es c.e.
3. La clausura de Kleene de lenguaje computable es computable
4. La clausura de Kleene de un lenguaje regular es regular
5. La clausura de Kleene de un lenguaje libre de contexto es libre de contexto
6. La reversa de un lenguaje computable es computable.
7. La reversa de un lenguaje c.e es c.e.

**Ejercicio 9.** Indicar V o F y justificar. Es decidible que

1. Si  $Halt : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$  fuera computable entonces la complejidad de Kolmogorov  $C : \mathbb{N} \rightarrow \mathbb{N}$ ,

$$C(s) = \min\{\#(P) : \Psi_P^{(0)} = s\}$$

sería computable.

2. La pertenencia de una palabra a un lenguaje computable es computable.
3. Todo conjunto infinito c.e. tiene un subconjunto infinito computable

**Ejercicio 10.** Sea APD  $P = (Q, \Sigma, \delta, \Gamma, q_0, F)$  y AP  $S = (Q', \Sigma', \delta', \Gamma, q'_0, F')$ . Indicar V o F y justificar.

1.  $\mathcal{L}(P)$  union  $\mathcal{L}(S)$  es libre de contexto
2.  $\mathcal{L}(P)$  interseccion  $\mathcal{L}(S)$  es libre de contexto.
3. Para todo AP hay un APD que reconoce el mismo lenguaje.
4. Es decidible si dos AFs reconocen el mismo lenguaje o no.

**Ejercicio 11.** Dar un algoritmo que codetermine un automata finito.

**Ejercicio 12.** Dado un autómata finito determinístico  $A = (Q_A, \Sigma, \delta_A, q_0, F_A)$  y dado autómata de pila determinístico  $P = (Q_P, \Sigma, \Gamma, \delta_P, p_0, F_P)$  dar un algoritmo que decida si el lenguaje  $\mathcal{L}(A) \cap \mathcal{L}(P)$  es finito. Justificar la correctitud.

**Ejercicio 13.** Un autómata de pila no determinístico,  $(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$  es un contador si  $\Gamma = \{Z_0, I\}$  el símbolo  $Z_0$  representa el valor del cero y  $I$  representa el valor 1. En cada transición el autómata solamente puede consultar si el contador es 0 o no. El contador no puede volverse negativo, por lo que no puede restar el valor 1 de un contador que actualmente es 0.

Demostrar Verdadero o Falso

- a. Si un lenguaje es reconocible por un autómata contador entonces el lenguaje complemento también.
- b. Si dos lenguaje  $L_A$  y  $L_B$  reconocibles por autómatas contadores entonces el lenguaje de su union  $L_A \cup L_B$  también.
- c. Si dos lenguaje  $L_A$  y  $L_B$  reconocibles por un autómatas contadores entonces el lenguaje de su intersección  $L_A \cap L_B$  también.
- d. Todos los lenguajes reconocibles por autómatas finitos son reconocibles por autómatas contadores.
- e. No todos los lenguajes reconocibles por un autómata contador son reconocibles por un autómata finito.

Ayuda: Considerar como se demuestran y cómo se refutan las propiedades de clausura de los lenguajes libres de contexto,

**Ejercicio 14.** Un autómata de cola es un autómata no determinístico, similar a uno de pila pero tiene una cola en vez de una pila. Formalmente, un autómata de cola es  $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$  donde

$Q$  es el conjunto de estados

$\Sigma$  es el alfabeto de la cinta entrada

$\Gamma$  es el alfabeto de cola,

$\delta : Q \times \Sigma \cup \{\lambda\} \times \Gamma \cup \{\lambda\} \rightarrow \mathcal{P}(Q \times \Gamma \cup \lambda)$  es la función de transición

$q_0 \in Q$  es el estado inicial.

$F \subseteq Q$  es el conjunto de estados finales.

Por ejemplo  $\delta(q_1, a, b) = \{(q_2, c), (q_3, d)\}$  dice que el estado  $q_1$  si lee  $a$  de la entrada y  $b$  está primero en la cola, entonces  $b$  sale de la cola, el autómata pasa al estado  $q_2$ , y pone  $c$  último en la cola. Se sabe (y no es difícil de demostrar) que los autómatas de cola reconocen todos los lenguajes libres de contexto.

Demostrar que los autómatas de cola reconocen más lenguajes que los reconocibles por autómatas de pila.

Ayuda: dar un lenguaje que no es libre de contexto.

**Ejercicio 15.** Dado  $R$  un lenguaje Regular, y dado  $L$  un lenguaje libre de contexto determinístico, ¿Es decidible si  $R$  está incluido en  $L$ ?

Respuesta: Teorema 10.6 Hopcroft 1976

**Ejercicio 16.** Sea  $g : \mathbb{N}^{(n+1)} \rightarrow \mathbb{N}$  parcial computable y sea índice  $e$  tal que  $\Phi_e^{(n)}(x_1, \dots, x_n) = g(e, x_1, \dots, x_n)$  Dar otro índice  $d$  tal que  $\Phi_d^{(n)}(x_1, \dots, x_n) = g(d, x_1, \dots, x_n)$ .

**Ejercicio 17.** Demostrar

el conjunto de todas las funciones totales computables  $\mathbb{N} \rightarrow \mathbb{N}$ ,  $Tot$ , no es c.e.

el conjunto de todas las funciones  $\mathbb{N} \rightarrow \mathbb{N}$  que no son totales computables,  $\overline{Tot}$ , no es c.e.