

TEORÍA DE AUTÓMATAS I

Informática de Sistemas

Soluciones a las cuestiones de examen del curso 1997/98

Febrero 98, 1ª semana

1. Sea un autómata finito de n estados que acepta una cadena de longitud $2n$. ¿Ha de aceptar necesariamente alguna cadena de longitud mayor que $3n$?

- a) Sí b) No

Solución: A. Si acepta una cadena de longitud $2n$ significa que hay al menos un estado por el que el autómata ha pasado al menos dos veces al aceptar la cadena; es decir, el autómata contiene un bucle, por lo que ha de aceptar cadenas de longitud arbitrariamente grande (cf. teorema 1.2 del libro de texto).

2. La unión de un conjunto de lenguajes regulares, ¿es necesariamente regular?

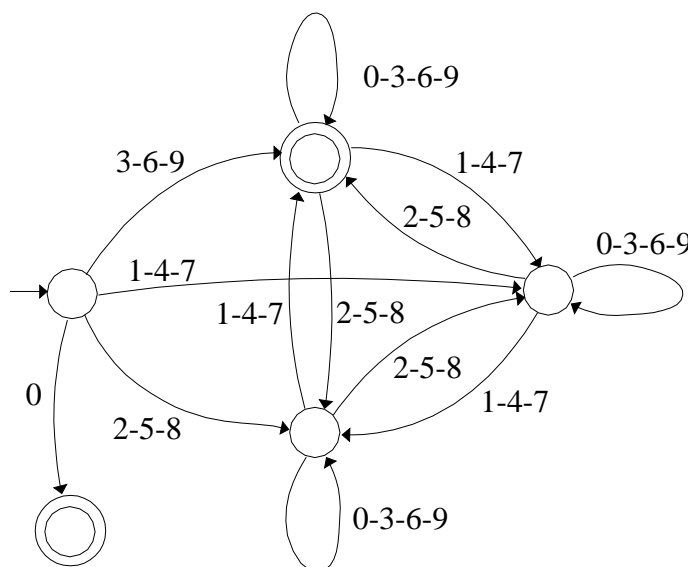
- a) Sí, siempre b) No necesariamente

Solución: B. Sólo es necesariamente regular para un conjunto finito. Contraejemplo: sea L_i el lenguaje que sólo contiene la cadena $x^i y^i$. La unión de todos los L_i desde 0 a infinito es el lenguaje no regular $x^n y^n$.

3. Dado el alfabeto $\{0, \dots, 9\}$, el lenguaje formado por los números enteros no negativos múltiplos de 3 es

- a) regular
b) estructurado por frases no regular

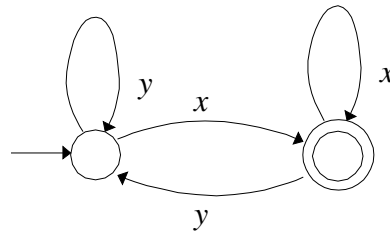
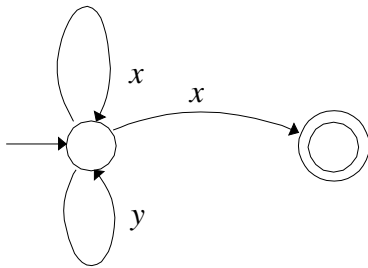
Solución: A. El siguiente autómata reconoce el lenguaje (para simplificar el diagrama, cada arco rotulado con 1-4-7 representa tres arcos rotulados con 1, 4 y 7):



4. ¿Son equivalentes estos dos autómatas?

a) Sí

b) No

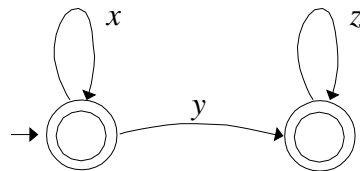


Solución: A. En ambos casos el lenguaje aceptado es $(x \cup y)^* \cdot x$.

5. ¿Son equivalentes la expresión regular $x^* \cup yz^*$ y este autómata?

a) Sí

b) No



Solución: B. El autómata acepta la cadena xyz .

6. ¿Son equivalentes la expresión regular $(x \cup yx^*y) \cdot x$ y esta gramática?

a) Sí

b) No

$$S \rightarrow xA$$

$$S \rightarrow yB$$

$$A \rightarrow xA$$

$$A \rightarrow \lambda$$

$$B \rightarrow xB$$

$$B \rightarrow yA$$

Solución: B. La expresión regular no representa la cadena yy .

7. ¿Es independiente de contexto el lenguaje $\{x^n y^m x^n y^m \mid m \geq n \geq 0\}$?

a) Sí

b) No

Solución: B. Intuitivamente, se ve que es así porque la pila no puede “controlar” por separado el número de x y el de y 's. La demostración rigurosa consiste en aplicar el lema de bombeo (teorema 2.5 del libro de texto).

8. ¿Es posible establecer una aplicación biunívoca (relación uno a uno) entre los autómatas de pila y los lenguajes independientes de contexto tal que a cada autómata le corresponda el lenguaje que acepta?

a) Sí

b) No

Solución: B. Tal relación no es biunívoca, porque es posible que dos autómatas diferentes acepten el mismo lenguaje.

9. ¿Puede un autómata aceptar la cadena de longitud infinita $xyxyxy\dots$?

- a) Sí b) No

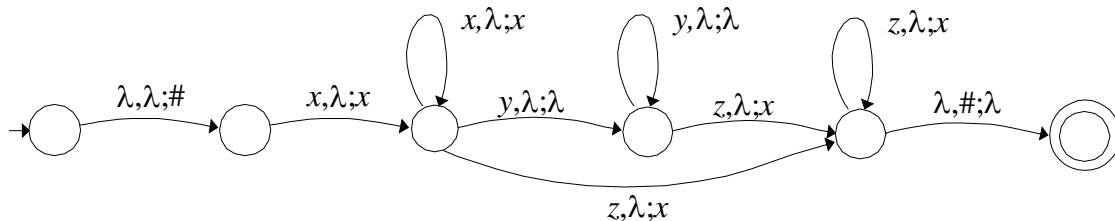
Solución: A. Una máquina de Turing es capaz de aceptar una cadena sin leer todos sus símbolos.

10. “Para todo autómata de pila no determinista M existe una máquina de Turing determinista M' que acepta el mismo lenguaje.”

- a) Verdadero b) Falso

Solución: A. $L(M)$ es independiente de contexto y, por tanto, estructurado por frases.

11. Queremos construir un autómata independiente de contexto determinista que acepte el lenguaje $x^n y^m z^n$, donde m y n son enteros no negativos. ¿Es correcta la siguiente solución? (Se supone que las transiciones que llevan al estado de captación global no violan el determinismo.)



- a) Correcta b) Incorrecta

Solución: B. El autómata no acepta la cadena y .

12. “La estrella de Kleene de un lenguaje independiente de contexto L es siempre independiente de contexto.”

- a) Verdadero b) Falso

Solución: A. Partiendo de una gramática independiente de contexto que genere L , renombramos S como S' , y añadimos las reglas $S \rightarrow S'S$ y $S \rightarrow \lambda$.

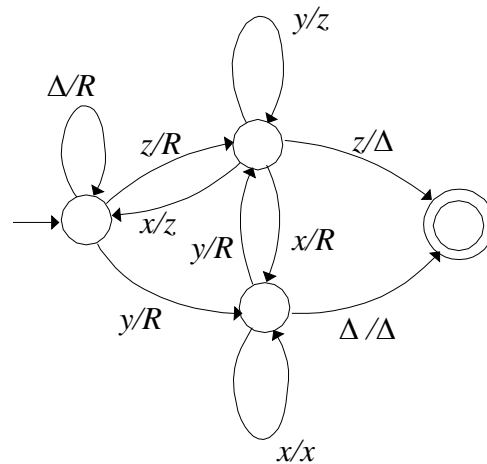
13. Dado un lenguaje estructurado por frases decidable L , ¿puede haber un autómata de pila M tal que $L(M)=L$?

- a) Sí b) No

Solución: A. Ejemplo: $L=\{x^n y^n\}$, por ser independiente de contexto, es estructurado por frases decidable.

14. ¿Pertenece la cadena yyx al lenguaje que acepta el siguiente autómata?

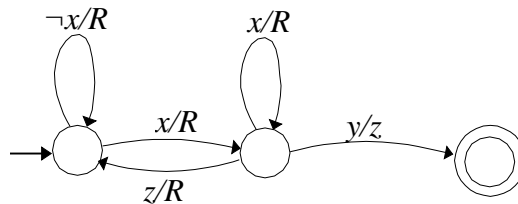
- a) Sí b) No



Solución: A. Las transiciones son: Δ/R , y/R , y/R , x/R , Δ/Δ .

15. Dado el alfabeto $\{x, y, z\}$, queremos construir una máquina de Turing determinista de tres estados que acepte el lenguaje formado por todas las cadenas que contienen la secuencia xy . ¿Es correcta la siguiente solución?

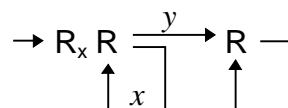
- a) Correcta. b) Incorrecta.



Solución: B. No es determinista porque no está completamente definida: no hay ninguna transición para el segundo estado cuando la celda examinada está vacía. (Podría suponerse que se trata de un diagrama incompleto, pero entonces el autómata tendría al menos cuatro estados.)

16. Dado el alfabeto $\{x, y\}$, queremos construir una máquina de Turing que acepte el lenguaje formado por las cadenas que **no** contienen ninguna y . ¿Es correcta la siguiente solución?

- a) Correcta b) Incorrecta



Solución: B. La máquina debería aceptar la cadena vacía.

Febrero 98, 2ª semana

17. “El complementario de la concatenación de dos lenguajes es igual a la concatenación de los complementarios de ambos; es decir, $\forall L_1, \forall L_2, c(L_1 \cdot L_2) = c(L_1) \cdot c(L_2)$.”

- a) Verdadero b) Falso

Solución: B. Contraejemplo: $L_1=\{xy\}$, $L_2=\{z\}$. Por un lado, $xyz \notin c(L_1 \cdot L_2)$ —porque $xyz \in L_1 \cdot L_2$ — y, por otro, $xyz \in c(L_1) \cdot c(L_2)$ —porque $x \in c(L_1)$ e $yz \in c(L_2)$ —.

18. La intersección de un conjunto de lenguajes regulares, ¿es regular?

- a) Sí, siempre b) No necesariamente

Solución: B. Sólo es necesariamente regular para un conjunto finito.

19. Dado un autómata finito no determinista M , ¿es posible construir a partir de M un analizador (un programa de ordenador) para el lenguaje $L(M)$ sin tener que construir primero un autómata determinista equivalente?

- a) Sí, siempre b) Depende de los casos

Solución: A. Con una estrategia de retroceso (págs. 44-45 del libro de texto).

20. ¿Cuántas cadenas de longitud 6 representa la expresión regular

$x \cdot (y \cup z) \cdot x \cdot (x \cup y \cup z)^* \cdot x$?

- a) Menos de 10 b) 10 o más

Solución: B. Son las $2 \times 3 \times 3 = 18$ cadenas representadas por la expresión regular

$x \cdot (y \cup z) \cdot x \cdot (x \cup y \cup z) \cdot (x \cup y \cup z) \cdot x$.

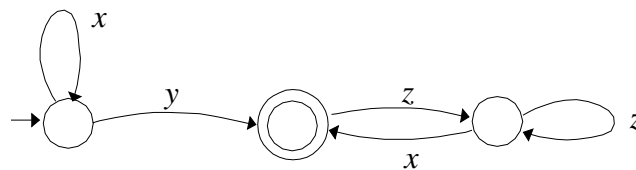
21. Dado un lenguaje regular L , definimos L' como el lenguaje formado por todas las cadenas de L que tienen tantas x 's como y 's. ¿Es L' regular?

- a) Sí, siempre b) Depende de L

Solución: B. Si $L = \Sigma^*$, L' es el lenguaje formado por todas las cadenas que tienen tantas x 's como y 's, el cual no es regular.

22. ¿Son equivalentes la expresión regular $x^* \cdot y \cdot (z^* x)^*$ y este autómata?

- a) Sí b) No



Solución: B. El autómata no acepta la cadena xyx .

23. ¿Son equivalentes la expresión regular $xy^*z \cdot (xy^*z)^*$ y esta gramática?

- a) Sí b) No

$$\begin{aligned}
S &\rightarrow xA \\
A &\rightarrow yA \\
A &\rightarrow zB \\
B &\rightarrow zA \\
B &\rightarrow \lambda
\end{aligned}$$

Solución: B. La expresión regular no representa la cadena $xyzzyz$.

24. ¿Dónde hay más cadenas de longitud menor que 1000: en $L(G_1)$ o en $L(G_2)$?

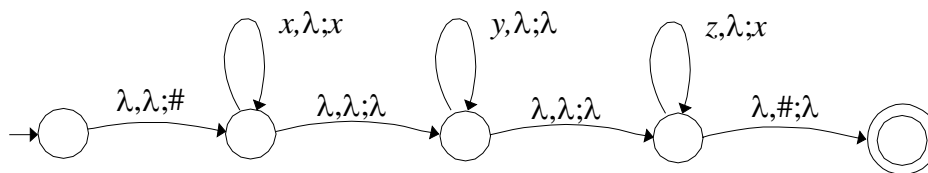
$$G_1 = \begin{cases} S \rightarrow X \\ S \rightarrow Y \\ X \rightarrow xXy \\ Y \rightarrow xxYy \\ X \rightarrow \lambda \\ Y \rightarrow \lambda \end{cases} \quad G_2 = \begin{cases} S \rightarrow X \\ X \rightarrow Y \\ X \rightarrow xXy \\ Y \rightarrow xxYy \\ X \rightarrow \lambda \\ Y \rightarrow \lambda \end{cases}$$

a) En $L(G_1)$

b) En $L(G_2)$

Solución: B. $L(G_1) = \{x^n y^n\} \cup \{x^{2n} y^n\}$, mientras que $L(G_2) = \{x^n y^m \mid m \leq n \leq 2m\} = L(G_2)$, de modo que $L(G_1) \subset L(G_2)$; por tanto, todas las cadenas de $L(G_1)$ de longitud menor que 1000 pertenecen también a $L(G_2)$. En cambio, en el conjunto $L(G_2) - L(G_1) = \{x^n y^m \mid m < n < 2m\}$ hay muchas cadenas de longitud menor que 1000; por ejemplo, para $m=300$ hay 299 cadenas.

25. “El lenguaje que acepta el siguiente autómata es independiente de contexto determinista.”



a) Verdadero

b) Falso

Solución: A. Es el lenguaje y^n , que es regular y, por tanto, independiente de contexto determinista.

26. ¿Es independiente de contexto el lenguaje $\{x^n y^m x^n y^m \mid m \geq 0, 0 \leq n \leq q\}$?

a) Sí

b) No

Solución: A. Existe una gramática de $3 \cdot (q+1)$ reglas tal que para cada i , $0 \leq i \leq q$, existen tres reglas: $S \rightarrow x^i A_i$, $A_i \rightarrow y A_i y$, $A_i \rightarrow x^i$. Estas tres reglas generan todas las cadenas del tipo $x^i y^m x^i y^m$ con $m \geq 0$.

27. Dado un lenguaje independiente de contexto determinista L , ¿puede haber un autómata finito M tal que $L(M)=L$?

- a) Sí b) No

Solución: A. Ejemplo: $L=\{x^n y^m\}$; por ser regular, es independiente de contexto determinista.

28. ¿Puede existir para una gramática independiente de contexto un autómata finito no determinista que acepte el mismo lenguaje?

- a) Sí b) No

Solución: A. Toda gramática regular G es independiente de contexto y, a la vez, existe una autómata finito determinista que acepta el mismo lenguaje que G .

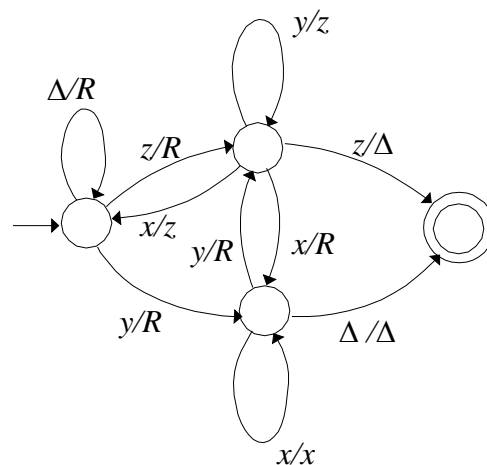
29. ¿Es posible establecer una aplicación biunívoca (relación uno a uno) entre las máquinas de Turing deterministas y los lenguajes estructurados por frases de modo que a cada máquina le corresponda el lenguaje que acepta?

- a) Sí b) No

Solución: B. Tal relación no es biunívoca, porque es posible que dos máquinas de Turing diferentes acepten el mismo lenguaje.

30. ¿Pertenece la cadena $yyxx$ al lenguaje que acepta el siguiente autómata?

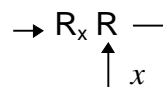
- a) Sí b) No



Solución: A. Las transiciones son: Δ/R , y/R , y/R , x/z , z/R , x/R , Δ/Δ .

31. Dado el alfabeto $\{x, y\}$, queremos construir una máquina de Turing que se detenga si y sólo si encuentra la secuencia xy . ¿Es correcta la siguiente solución?

- a) Correcta. b) Incorrecta.



Solución: B. La máquina se detiene si el contenido inicial de la cinta es $\Delta x \Delta \Delta \Delta \dots$

32. “La estrella de Kleene de un lenguaje estructurado por frases L es siempre un lenguaje estructurado por frases.”

- a) Verdadero b) Falso

Solución: A. Partiendo de una gramática que genere L , renombramos S como S' , y añadimos las reglas $S \rightarrow S'S$ y $S \rightarrow \lambda$.