Capítulo 3
Lenguajes no regulares

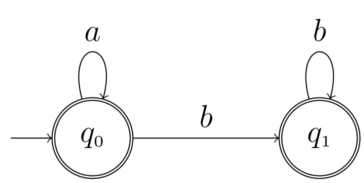
En este capítulo se demostrará que existen lenguajes no-regulares y, como consecuencia, se concluirá que hay problemas computacionales que no se pueden resolver utilizando los modelos de autómatas finitos (AFD, AFN y AFN-λ) hasta ahora considerados.

3.1. Criterio de no regularidad

El hecho de que todo autómata finito tiene solamente un número finito de estados pero el conjunto Σ^* tiene infinitas cadenas, permite concluir que hay ciertos lenguajes que no pueden ser aceptados por ningún autómata. El razonamiento se basa en una de las versiones del llamado Principio del Palomar: si infinitos objetos se distribuyen en un número finito de compartimientos, entonces hay un compartimiento que contiene dos objetos (por lo menos). A continuación presentamos un par de ejemplos.

Ejemplo Demostrar que el lenguaje $L = \{a^n b^n : n \geq 0\} = \{\lambda, a, a^2, a^3, a^4, \ldots\}$, sobre $\Sigma = \{a, b\}$, no es regular.

Solución. En primer lugar, hay que observar que el lenguaje L es diferente de a^*b^* ya que este último lenguaje está formado por las concatenaciones de cualquier número de $a$es con cualquier numero de $b$es, o sea, $a^*b^* = \{a^m b^n : m, n \geq 0\}$, mientras que cada cadena de L tiene igual número de $a$es que de $b$es. a^*b^* es regular y se puede aceptar por medio del siguiente AFD:

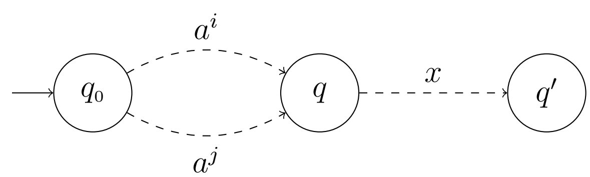


No obstante, para aceptar a L un autómata debe almacenar (de alguna forma) la información sobre el número de $a$es y compararla luego con el número de $b$es. El único tipo de memoria disponible son los estados y resulta intuitivamente claro que con un número finito de estados es imposible para un autómata almacenar la información necesaria para procesar y aceptar todas y cada una de las cadenas de la forma $a^n b^n$ con $n \geq 0$. Pero se requiere un argumento matemáticamente riguroso para demostrarlo concluyentemente.

Para concluir que L no es regular razonamos por contradicción (o reducción al absurdo): suponemos que L es regular y llegamos a una contradicción. Si L es regular, existe un AFD $M = (\Sigma, Q, q_0, F, \delta)$ tal que $L(M) = L$. Consideramos luego las potencias positivas de a, a saber, $a, a^2, a^3, a^4, \ldots$. Después de leer completamente cada una de estas cadenas, M termina en un determinado estado, pero como hay infinitas potencias de a y solamente un número finito de estados en M, deben existir dos cadenas diferentes a^i y a^j, con $i \neq j$, cuyo procesamiento completo termina en el mismo estado. Esta última afirmación es consecuencia del mencionado Principio del Palomar: si infinitos objetos se distribuyen en un número finito de compartimientos, entonces hay un compartimiento que contiene dos objetos (por lo menos). Utilizando la función $\widehat{\delta}$ (Definición 2.7.2), podemos expresar este hecho como $\widehat{\delta}(q_0, a^i) = \widehat{\delta}(q_0, a^j)$. Como M es un autómata determinista, se concluye que, para toda cadena $x \in \Sigma^*$, el procesamiento de las cadenas $a^i x$ y $a^j x$ termina en el mismo estado; en efecto,

$$\widehat{\delta}(q_0, a^i x) = \widehat{\delta}(\widehat{\delta}(q_0, a^i), x) = \widehat{\delta}(\widehat{\delta}(q_0, a^j), x) = \widehat{\delta}(q_0, a^j x).$$

En el grafo de M se puede visualizar la situación de la siguiente manera:



Esto implica que para toda $x \in \Sigma^*$, las cadenas $a^i x$ y $a^j x$ son ambas aceptadas o ambas rechazadas. La contradicción surge porque existe una cadena $x \in \Sigma^*$ tal que $a^i x$ es aceptada pero $a^j x$ es rechazada. Por simple inspección observamos que tomando $x = b^i$ se tendrá $a^i x = a^i b^i \in L$ pero $a^j x = a^j b^i \notin L$ (porque $i \neq j$). De modo que $a^i b^i$ es aceptada mientras que $a^j b^i$ es rechazada, lo cual es una contradicción.

Ejemplo Demostrar que el lenguaje de los palíndromes sobre $\Sigma = \{0, 1\}$ no es un lenguaje regular. Recuérdese que un palíndrome es una cadena u que coincide con su reflexión, o sea, tal que $u = u^R$.

Solución. Como en el ejemplo anterior, razonamos por contradicción. Suponemos que L es regular; entonces existe un AFD $M = (\Sigma, Q, q_0, F, \delta)$ tal que $L(M) = L$. A continuación consideramos las potencias positivas de 1, a saber, $1, 1^2, 1^3, 1^4, \ldots$. Después de leer completamente cada una de estas cadenas, M termina en un determinado estado; pero como hay infinitas potencias de 1 y solamente un número finito de estados en M, deben existir dos cadenas diferentes 1^i y 1^j, con $i \neq j$, cuyo procesamiento termina en el mismo estado. Es decir, $\widehat{\delta}(q_0, 1^i) = \widehat{\delta}(q_0, 1^j)$. Como M es un autómata determinista, se concluye que, para toda $x \in \Sigma^*$, el procesamiento de las cadenas $1^i x$ y $1^j x$ termina en el mismo estado.

En otras palabras, $\widehat{\delta}(q_0, 1^i x) = \widehat{\delta}(q_0, 1^j x)$ para toda cadena $x \in \Sigma^*$. Esto implica que para toda $x \in \Sigma^*$, las cadenas $1^i x$ y $1^j x$ son ambas aceptadas o ambas rechazadas. Se llega a una contradicción si podemos encontrar una cadena $x \in \Sigma^*$ tal que $1^i x$ sea aceptada pero $1^j x$ sea rechazada (o viceversa). Procediendo por inspección (ensayo y error) observamos que tomando $x = 01^i$ se tendrá que $1^i x = 1^i 01^i$ es un palíndrome pero $1^j x = 1^j 01^i$ no lo es (porque $i \neq j$). De modo que $1^i 01^i$ es aceptada mientras que $1^j 01^i$ es rechazada, lo cual es una contradicción.

Nótese que podríamos haber construido un razonamiento completamente similar considerando las infinitas potencias positivas de 0 $(0, 0^2, 0^3, 0^4, \ldots,)$ en vez de las de 1.

Utilizando la noción de distinguiblidad entre cadenas introducida en la sección 2.15, el argumento usado en los dos ejemplos anteriores se puede convertir en un criterio práctico para demostrar que ciertos lenguajes no son regulares. Recordamos la definición de cadenas distinguibles (Definición 2.15.1). Dado un lenguaje L sobre un alfabeto Σ, dos cadenas u y v en Σ^* son L-distinguibles si

$$(\exists x \in \Sigma^*) \left[(ux \in L \text{ y } vx \notin L) \text{ ó } (ux \notin L \text{ y } vx \in L) \right].$$

Es decir, u y v son L-distinguibles si existe x en Σ^* tal que una de las cadenas ux ó vx está en L y la otra no.

3.1.1. Criterio de no regularidad. Sea Σ un alfabeto dado y L un lenguaje sobre Σ. Si en Σ^* hay infinitas cadenas L-distinguibles dos a dos (es decir, dos cadenas diferentes son L-distinguibles entre sí), entonces L no es regular.

Presentaremos dos demostraciones de este teorema. La primera es una consecuencia directa del Teorema de Myhill-Nerode mientras que la segunda demostración es un razonamiento independiente de tal teorema y es similar al argumento utilizado en los ejemplos previos de la presente sección.
Primera demostración. Si hay infinitas cadenas L-distinguibles entre sí, entonces la relación de indistinguibilidad I_L determina infinitas clases de equivalencia sobre Σ^*; es decir I_L tiene índice infinito. Por el Teorema de Myhill-Nerode, L no es regular.

Segunda demostración. Razonamos por contradicción (o reducción al absurdo): suponemos que L es regular y llegamos a una contradicción. Si L es regular, existe un AFD $M = (\Sigma, Q, q_0, F, \delta)$ tal que $L(M) = L$. Por hipótesis, existen infinitas cadenas L-distinguibles dos a dos, $u_1, u_2, u_3, \ldots$. Después de leer completamente cada una de estas cadenas, M termina en un determinado estado, pero como solamente hay un número finito de estados en M, por el Principio del Palomar se deduce que deben existir dos cadenas diferentes u_i y u_j, con $i \neq j$, cuyo procesamiento completo termina en el mismo estado, es decir, $\widehat{\delta}(q_0, u_i) = \widehat{\delta}(q_0, u_j)$. Como M es un autómata determinista, se concluye que, para toda cadena $x \in \Sigma^*$, el procesamiento de las cadenas $u_i x$ y $u_j x$ termina en el mismo estado; en efecto,

$$\widehat{\delta}(q_0, u_i x) = \widehat{\delta}(\widehat{\delta}(q_0, u_i), x) = \widehat{\delta}(\widehat{\delta}(q_0, u_j), x) = \widehat{\delta}(q_0, u_j x).$$

Esto implica que para toda $x \in \Sigma^*$, las cadenas $u_i x$ y $u_j x$ son ambas aceptadas o ambas rechazadas, es decir, o bien ($u_i x \in L$ y $u_j x \in L$), o bien ($u_i x \notin L$ y $u_j x \notin L$). Esto contradice el hecho de que u_i y u_j son L-distinguibles.

Según este criterio, para concluir que L no es regular basta exhibir infinitas cadenas en Σ^* que sean L-distinguibles dos a dos. No es necesario utilizar explícitamente autómatas al invocar el criterio de no regularidad. En la mayoría de los casos será suficiente mostrar que $a, a^2, a^3, \ldots$ son infinitas cadenas L-distinguibles dos a dos (en caso de que $a \in \Sigma$). En algunos problemas resulta conveniente mostrar que las potencias pares, $a^2, a^4, a^6, \ldots$ (o bien, las potencias impares) son infinitas cadenas L-distinguibles dos a dos. Los siguientes ejemplos ilustran el uso de esta técnica.

Ejemplo Utilizar el criterio de no regularidad para demostrar que el lenguaje $L = \{a^n b^n : n \geq 0\}$, sobre $\Sigma = \{a, b\}$, no es regular.

Solución. Vamos a comprobar que $a, a^2, a^3, \ldots$ son infinitas cadenas L-distinguibles dos a dos. Sean $i, j \geq 1$, con $i \neq j$. Queremos mostrar que a^i y a^j son L-distinguibles, para lo cual hay que encontrar una cadena x tal que $a^i x \in L$ pero $a^j x \notin L$. La escogencia de x es obvia: $x = b^i$. Se tiene que $a^i x = a^i b^i \in L$ pero $a^j x = a^j b^i \notin L$. Esto muestra que a^i y a^j son L-distinguibles si $i \neq j$ y, por el criterio 3.1.1, L no puede ser regular.

Ejemplo Utilizar el criterio de no regularidad para demostrar que el lenguaje de los palíndromes sobre $\Sigma = \{0, 1\}$ no es un lenguaje regular.

Solución. Vamos a comprobar que $1, 1^2, 1^3, \ldots$ son infinitas cadenas L-distinguibles dos a dos. Sean $i, j \geq 1$, con $i \neq j$. Queremos mostrar que 1^i y 1^j son L-distinguibles, para lo cual hay que encontrar una cadena x tal que $1^i x \in L$ pero $1^j x \notin L$. Escogemos $x = 01^i$. Se tendrá entonces que $1^i x = 1^i 0 1^i \in L$ pero $1^j x = 1^j 0 1^i \notin L$. Esto muestra que 1^i y 1^j son L-distinguibles si $i \neq j$ y, por criterio 3.1.1, L no puede ser regular.

Ejemplo Utilizar el criterio de no regularidad para demostrar que el lenguaje $L = \{a^{n^2} : n \geq 1\}$ no es regular. L está formado por cadenas de $a$es cuya longitud es un cuadrado perfecto, sobre el alfabeto $\Sigma = \{a\}$.

Solución. Vamos a comprobar que las cadenas de L, o sea $a, a^4, a^9, a^{16}, a^{25}, \ldots$ son L-distinguibles dos a dos. Sean $i, j \geq 1$, con $i \neq j$; sin pérdida de generalidad podemos suponer que $i < j$. Queremos mostrar que a^{i^2} y a^{j^2} son L-distinguibles. Si escogemos $x = a^{2i+1}$, se tendrá que $a^{i^2} x = a^{i^2} a^{2i+1} = a^{i^2+2i+1} = a^{(i+1)^2} \in L$. De otro lado, $a^{j^2} x = a^{j^2} a^{2i+1} = a^{j^2+2i+1} \notin L$ ya que

$$j^2 < j^2 + 2i + 1 < j^2 + 2j + 1 = (j+1)^2.$$

Esto quiere decir que el número de $a$es que hay en a^{j^2+2i+1} está estrictamente comprendido entre dos cuadrados perfectos consecutivos, a saber, j^2 y $(j+i)^2$. Por consiguiente, $a^{j^2+2i+1} \notin L$.

Ejercicios de la sección 3.1

① Ya sea utilizando un argumento por contradicción o el criterio de no regularidad, demostrar que los siguientes lenguajes no son regulares:

(i) $L = \{a^n b^{2n} : n \geq 0\}$, sobre $\Sigma = \{a, b\}$.
(ii) $L = \{a^{2n} b^n : n \geq 0\}$, sobre $\Sigma = \{a, b\}$.
(iii) $L = \{a^n b^m : n, m \geq 0,\ n \neq m\}$, sobre $\Sigma = \{a, b\}$.
(iv) $L = \{a^m b^n : m \geq n \geq 0\}$, sobre $\Sigma = \{a, b\}$.
(v) $L = \{0^n 1 0^n : n \geq 1\}$, sobre $\Sigma = \{0, 1\}$.
(vi) $L = \{0^m 1^n 0^m : m, n \geq 0\}$, sobre $\Sigma = \{0, 1\}$.
(vii) $L = \{1^n 0 1^n 0 : n \geq 1\}$, sobre $\Sigma = \{0, 1\}$.
(viii) $L = \{0 1^n 0 1^n : n \geq 1\}$, sobre $\Sigma = \{0, 1\}$.
(ix) $L = \{uu : u \in \Sigma^*\}$, sobre $\Sigma = \{a, b\}$.
(x) $L = \{uu^R : u \in \Sigma^*\}$, sobre $\Sigma = \{a, b\}$.

② $\Sigma = \{0, 1\}$. Ya sea utilizando un argumento por contradicción o el criterio de no regularidad, demostrar que el lenguaje $L = \{u \in \Sigma^* : \#_0(u) = \#_1(u)\}$ no es regular.

③ Utilizar el criterio de no regularidad para demostrar que el lenguaje

$$L = \{a^n : n \text{ es un número primo}\},$$

sobre $\Sigma = \{a\}$, no es regular. Ayuda: usar el Teorema de Legendre según el cual, si p y q son primos, entonces la progresión aritmética $\{p + kq : k \geq 0\}$ contiene infinitos primos.

④ Demostrar que a^*b^* es un lenguaje regular que es la unión de dos lenguajes disyuntos no-regulares.

⑤ Según la definición de los conjuntos regulares, la unión de un número finito de lenguajes regulares también es regular. Comprobar mediante un contraejemplo que la unión de una familia infinita de conjuntos regulares no necesariamente es regular.

⑥ Sea L un lenguaje no-regular y N un subconjunto finito de L. Demostrar que $L - N$ tampoco es regular.

⑦ Demostrar o refutar las siguientes afirmaciones:

(i) Un lenguaje no-regular debe ser infinito.
(ii) Si el lenguaje $L_1 \cup L_2$ es regular, también lo son L_1 y L_2.

(iii) Si los lenguajes L_1 y L_2 no son regulares, el lenguaje $L_1 \cap L_2$ tampoco puede ser regular.

(iv) Si el lenguaje L^* es regular, también lo es L.

(v) Si L es regular y N es un subconjunto finito de L, entonces $L - N$ es regular.

(vi) Un lenguaje regular no-vacío L es infinito si y sólo si en cualquier expresión regular de L aparece por lo menos una $*$ (estrella de Kleene).

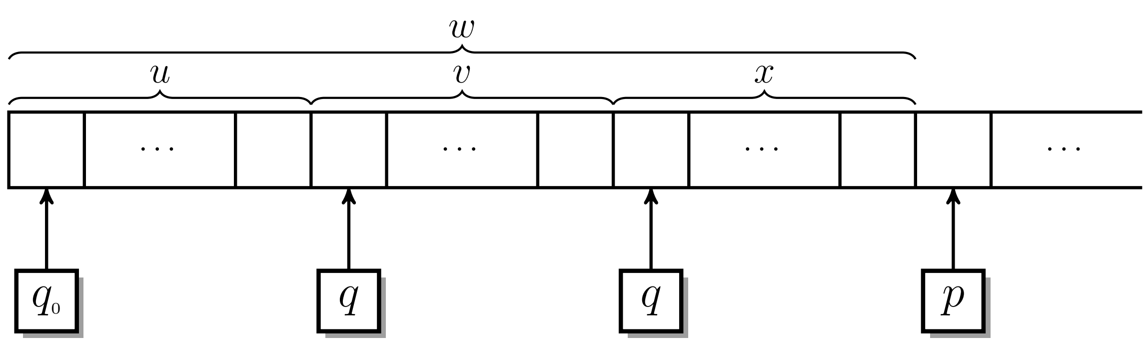
3.2. Lema de bombeo (← Opcional)

El llamado “lema de bombeo” (*pumping lemma*, en inglés) es una propiedad de los lenguajes regulares que también se puede usar para demostrar, razonando por contradicción, que ciertos lenguajes no son regulares. En comparación con el criterio de no regularidad, el Lema de bombeo no es una herramienta práctica en casos concretos porque el razonamiento es mucho más complejo. No obstante, se trata de un enunciado clásico en la teoría de lenguajes formales que se puede generalizar para otras colecciones de lenguajes, con importantes consecuencias, tal como veremos en capítulos posteriores.

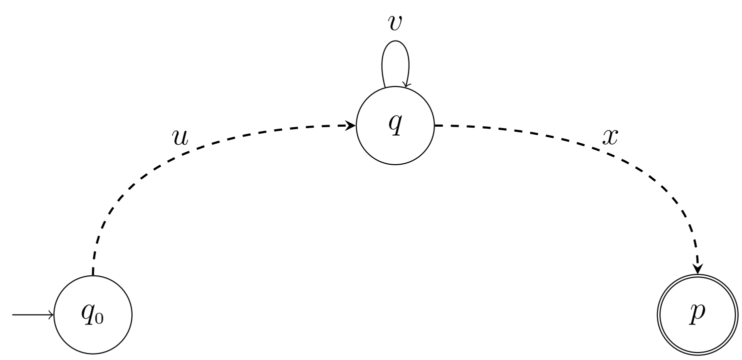
3.2.1. Lema de bombeo. Para todo lenguaje regular L (sobre un alfabeto dado Σ) existe una constante $n \in \mathbb{N}$, llamada constante de bombeo para L, tal que toda cadena $w \in L$, con $|w| \geq n$, satisface la siguiente propiedad:

$$(B)\ \begin{cases} w \text{ se puede descomponer como } w = uvx, \text{ donde } |uv| \leq n, \\ v \neq \lambda, \text{ y para todo } i \geq 0 \text{ se tiene } uv^i x \in L. \end{cases}$$

Demostración. Por el Teorema de Kleene y por los teoremas de equivalencia de los modelos AFD, AFN y AFN-λ, existe un AFD M tal que $L(M) = L$. Sea n el número de estados de M. Si $w \in L$ y $|w| \geq n$, entonces durante el procesamiento completo de w, hay por lo menos un estado que se repite. Sea q el primer estado que se repite. Tal como se muestra en la siguiente gráfica, w se puede descomponer como concatenación de tres cadenas, esto es, $w = uvx$, donde $|uv| \leq n$, $v \neq \lambda$.



Puesto que $w \in L(M)$, p es un estado de aceptación. Nótese que tanto u como x pueden ser la cadena vacía λ, pero $v \neq \lambda$. Además, la cadena v se puede “bombear”, en el sentido de que $uv^i x$ es aceptada por M para todo $i \geq 0$. La propiedad de bombeo de v se puede apreciar en el grafo del autómata M, como se exhibe la siguiente página.



Uso del lema de bombeo. El lema de bombeo se puede usar para concluir que un cierto lenguaje dado L no es regular, recurriendo a un razonamiento por contradicción (o reducción al absurdo). El razonamiento utilizado tiene la siguiente forma:

1. Si L fuera regular, existiría una constante de bombeo n para L.
2. Se escoge una cadena "adecuada" $w \in L$ y se aplica la propiedad (B) del lema de bombeo, descomponiendo w como

$$w = uvx, \quad v \neq \lambda, \quad |uv| \leq n.$$

3. Se llega a la siguiente contradicción:
   (I) Por el lema de bombeo, $uv^i x \in L$, para todo $i \geq 0$.
   (II) $uv^i x$ no puede estar en L, para algún $i \in I$. Por lo general, basta escoger valores pequeños de i como $i = 0$ ó $i = 2$.

Ejemplo Usar el lema de bombeo para demostrar que el lenguaje $L = \{a^i b^i : i \geq 0\}$ no es regular.

Solución. Si L fuera regular, existiría una constante de bombeo n para L. Sea $w = a^n b^n \in L$. Entonces w se puede descomponer como $w = uvx$, con $|v| \geq 1$ y $|uv| \leq n$. Por lo tanto, u y v constan únicamente de $a$es:

$$\begin{aligned} u &= a^r, && \text{para algún } r \geq 0, \\ v &= a^s, && \text{para algún } s \geq 1. \end{aligned}$$

Entonces,

$$x = a^{n-(r+s)} b^n = a^{n-r-s} b^n.$$

Por el lema de bombeo, $uv^i x \in L$ para todo $i \geq 0$. En particular, si $i = 0$, $ux \in L$. Pero $ux = a^r a^{n-r-s} b^n = a^{n-s} b^n$. Como $n - s \neq n$, la cadena $ux \notin L$ lo cual es una contradicción. Se concluye entonces que L no puede ser regular.

Tomando $i = 2$ también se llega a una contradicción: por un lado, $uv^2 x \in L$, pero

$$uv^2 x = a^r a^s a^s a^{n-r-s} b^n = a^{r+2s+n-r-s} b^n = a^{n+s} b^n.$$

Como $s \geq 1$, $a^{n+s}b^n$ no está en L.

El argumento anterior también sirve para demostrar que el lenguaje $L = \{a^i b^i : i \geq 1\}$ no es regular.

Ejemplo Demostrar que el lenguaje de los palíndromes sobre $\{a, b\}$ no es un lenguaje regular.

Solución. Si L fuera regular, existiría una constante de bombeo n para L. Sea $w = a^n b a^n \in L$. Entonces w se puede descomponer como $w = uvx$, con $|v| \geq 1$, $|uv| \leq n$, y para todo $i \geq 0$, $uv^i x \in L$. Por lo tanto, u y v constan únicamente de $a$es:

$$\begin{aligned} u &= a^r, \qquad \text{para algún } r \geq 0, \\ v &= a^s, \qquad \text{para algún } s \geq 1. \end{aligned}$$

Entonces,

$$x = a^{n-(r+s)} b a^n = a^{n-r-s} b a^n.$$

Tomando $i = 0$, se concluye que $ux \in L$, pero

$$ux = a^r a^{n-r-s} b a^n = a^{n-s} b a^n.$$

Como $s \geq 1$, $a^{n-s}ba^n$ no es un palíndromo. Esta contradicción muestra que L no puede ser regular.

Ejemplo Demostrar que el lenguaje $L = \left\{a^{i^2} : i \geq 0\right\}$ no es regular. L está formado por cadenas de $a$es cuya longitud es un cuadrado perfecto.

Solución. Si L fuera regular, existiría una constante de bombeo n para L. Sea $w = a^{n^2} \in L$. Entonces w se puede descomponer como $w = uvx$, con $|v| \geq 1$, $|uv| \leq n$. Por el lema de bombeo, $uv^2x \in L$, pero por otro lado,

$$n^2 < n^2 + |v| = |uvx| + |v| = |uv^2x| \leq n^2 + |uv| \leq n^2 + n < (n+1)^2.$$

Esto quiere decir que el número de símbolos de la cadena uv^2x no es un cuadrado perfecto y, por consiguiente, $uv^2x \notin L$. En conclusión, L no es regular.

Ejercicios de la sección 3.2

① Utilizar el lema de bombeo demostrar que los siguientes lenguajes no son regulares:

(i) $L = \{a^n b^{2n} : n \geq 0\}$, sobre $\Sigma = \{a, b\}$.

(ii) $L = \{a^{2n} b^n : n \geq 0\}$, sobre $\Sigma = \{a, b\}$.

(iii) $L = \{a^n b^m : n, m \geq 0,\ n \neq m\}$, sobre $\Sigma = \{a, b\}$.

(iv) $L = \{a^m b^n : m \geq n \geq 0\}$, sobre $\Sigma = \{a, b\}$.

(v) $L = \{0^n 1 0^n : n \geq 1\}$, sobre $\Sigma = \{0, 1\}$.

(vi) $L = \{0^m 1^n 0^m : m, n \geq 0\}$, sobre $\Sigma = \{0, 1\}$.

(vii) $L = \{1^n01^n0 : n \geq 1\}$, sobre $\Sigma = \{0, 1\}$.
(viii) $L = \{01^n01^n : n \geq 1\}$, sobre $\Sigma = \{0, 1\}$.
(ix) $L = \{uu : u \in \Sigma^*\}$, sobre $\Sigma = \{a, b\}$.
(x) $L = \{uu^R : u \in \Sigma^*\}$, sobre $\Sigma = \{a, b\}$.

② Encontrar la falacia en el siguiente argumento: "Según la propiedad (B) del enunciado del lema de bombeo, si L es un lenguaje regular se tiene que $uv^ix \in L$ para todo $i \geq 0$. Por consiguiente, L posee infinitas cadenas y, en conclusión, todo lenguaje regular es infinito."

3.3. Problemas insolubles para autómatas finitos

La existencia de lenguajes no-regulares implica que hay ciertos problemas que no pueden ser resueltos por medio de autómatas finitos. Se puede pensar que un autómata es un mecanismo que produce dos salidas para cada entrada $u \in \Sigma^*$: SÍ (cuando u es aceptada) y NO (cuando u es rechazada). Los problemas que puede resolver un autómata son entonces *problemas de decisión*, esto es, problemas para los cuales hay dos únicas salidas, SÍ o NO. Esta idea la precisamos a continuación.

Dado un alfabeto Σ, consideremos una propiedad $\mathcal{P}$ referente a las cadenas de Σ^*. El problema de decisión para $\mathcal{P}$ es el siguiente:

(3.3.1) Dada $u \in \Sigma^*$, ¿satisface u la propiedad $\mathcal{P}$?

Denotamos con $L_{\mathcal{P}}$ el lenguaje de todas las cadenas que satisfacen la propiedad $\mathcal{P}$, es decir,

$$L_{\mathcal{P}} = \{u \in \Sigma^* : u \text{ satisface } \mathcal{P}\} = \{u \in \Sigma^* : \mathcal{P}(u)\}.$$

Decimos que el problema (3.3.1) puede ser resuelto usando autómatas finitos si existe un AFD M tal que $L(M) = L_{\mathcal{P}}$. En tal caso, para cada entrada $u \in \Sigma^*$ se tendrá

$$\begin{cases} \text{Si } u \text{ es aceptada por } M \Longrightarrow u \in L(M) \Longrightarrow u \text{ satisface } \mathcal{P}. \\ \text{Si } u \text{ es rechazada por } M \Longrightarrow u \notin L(M) \Longrightarrow u \text{ no satisface } \mathcal{P}. \end{cases}$$

Por el Teorema de Kleene se concluye que el problema de decisión (3.3.1) se puede resolver usando autómatas finitos si y sólo si $L_{\mathcal{P}}$ es regular.

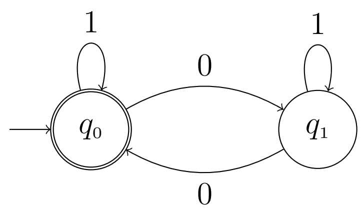
Ejemplo Sea $\Sigma = \{0, 1\}$. El problema de decisión

Dada $u \in \Sigma^*$, ¿tiene u un número par de ceros?

se puede resolver usando autómatas finitos. La propiedad $\mathcal{P}$ en este caso es "tener un número par de ceros" y $L_{\mathcal{P}}$ es el lenguaje

$$L_{\mathcal{P}} = \{u \in \Sigma^* : u \text{ tiene un número par de ceros}\} = \{u \in \Sigma^* : \#_0(u) \text{ es par}\}.$$

Sabemos que $L_{\mathcal{P}}$ es regular y es aceptado por el siguiente AFD M:



Entonces M resuelve el problema de decisión para $\mathcal{P}$.

Ejemplo Sea $\Sigma = \{0, 1\}$. El problema de decisión

Dada $u \in \Sigma^*$, ¿es u un palíndrome?

no se puede resolver usando autómatas finitos. La propiedad $\mathcal{P}$ en este caso es "ser un palíndrome" y $L_{\mathcal{P}}$ es el lenguaje de todos los palíndromes, el cual se demostró en la sección anterior que no es regular.

Ejemplo Sea $\Sigma = \{0, 1\}$. Demostrar que ningún autómata finito puede resolver el siguiente problema de decisión:

Dada $u \in \Sigma^*$, ¿tiene u el mismo número de ceros que de unos?

Solución. La propiedad $\mathcal{P}$ en este caso es "tener el mismo número de ceros que de unos" y $L_{\mathcal{P}}$ es el lenguaje $L_{\mathcal{P}} = \{u \in \Sigma^* : \#_0(u) = \#_1(u)\}$. Utilizando los argumentos de la sección anterior (ya sea por contradicción o por el criterio de no regularidad), es posible demostrar que $L_{\mathcal{P}}$ no es regular (véase el ejercicio ② de la sección 3.1). Por consiguiente, este problema es insoluble por medio de autómatas finitos.

Ejercicios de la sección 3.3

Determinar si los siguientes problemas de decisión se pueden o no resolver utilizando autómatas finitos. Ya sea la respuesta afirmativa o negativa, se debe presentar una demostración rigurosa.

① $\Sigma = \{a, b\}$. Dada $u \in \Sigma^*$, ¿tiene u un número par de $a$es y un número impar de $b$es?

② $\Sigma = \{a, b\}$. Dada $u \in \Sigma^*$, ¿se cumple que $\#_a(u) = \#_b(u) + 1$?

③ $\Sigma = \{a, b\}$. Dada $u \in \Sigma^*$, ¿es el número de $a$es en u un múltiplo 3? Nota: 0 se considera múltiplo de cualquier otro número natural.

④ $\Sigma = \{a, b\}$. Dada $u \in \Sigma^*$, ¿es el número de $a$es en u un múltiplo del número de $b$es en u?

⑤ $\Sigma = \{0, 1\}$. Dada $u \in \Sigma^*$, ¿el primer símbolo de u (de izquierda a derecha) coincide con el último símbolo de u (de izquierda a derecha)?

⑥ $\Sigma = \{0, 1\}$. Dada $u \in \Sigma^*$, ¿tiene u por lo menos un cero y por lo menos dos unos?

⑦ $\Sigma = \{0, 1\}$. Dada $u \in \Sigma^*$, ¿aparece la subcadena 01 una única vez en u?

⑧ $\Sigma = \{0, 1\}$. Dada $u \in \Sigma^*$, ¿se cumple $\#_0(u) \neq \#_0(u)$?

⑨ $\Sigma = \{0, 1\}$. Dada $u \in \Sigma^*$, ¿se cumple $\#_1(u) > \#_0(u)$?

⑩ $\Sigma = \{0, 1\}$. Dada $u \in \Sigma^*$, ¿se puede escribir u como la concatenación de dos mitades iguales? Es decir, ¿es $u = ww$, para alguna cadena $w \in \Sigma^*$?