



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
FACULTAD DE MATEMÁTICAS
DEPARTAMENTO DE MATEMÁTICA

Tarea 1

Teoría de Autómatas y Lenguajes Formales — IIC2223

Fecha de Entrega: 2020-08-27

Nicholas Mc-Donnell

Problema 1:

Sea $\Sigma = \{\#, a\}$. Para $i \geq 1$ se define $a^i = a \overset{i-\text{veces}}{\dots} a$. Para $n \geq 2$ se define el Lenguaje $L_n \subseteq \Sigma^*$ de todas las palabras de la forma:

$$\#a^{i_1}\#a^{i_2}\dots\#a^{i_k}$$

para algún $k \geq 0$ tal que $1 \leq i_j \leq n$ para todo $1 \leq j \leq k$. Por ejemplo, para $n = 3$ se tiene que $\#aa\#a$ y $\#a\#aaa\#aa$ pertenecen a L_3 , pero $\#aaaa$ no pertenece a L_3 . Notar que cuando $k = 0$ se tiene que $w = \varepsilon$, y por lo tanto siempre se cumple que $\varepsilon \in L_n$.

- (a) Para un n arbitrario, muestre como construir un autómata finito determinista \mathcal{A}_n tal que $\mathcal{L}(\mathcal{A}_n) = L_n$. Explique por qué su construcción cumple con lo pedido.
- (b) Para un n arbitrario, muestre como construir un autómata finito no-determinista \mathcal{B}_n con $n + 1$ estados tal que $\mathcal{L}(\mathcal{B}_n) = L_n$. Explique por qué su construcción cumple con lo pedido.

Solución problema 1:

- (a) Sea $Q = \{q_1, q_2, \dots, q_{n+1}, q_{n+2}\}$, $F = \{q_1, \dots, q_n\}$ y $q_0 = q_n$. Ahora, definimos δ de la siguiente manera:

$$\delta(q_i, a) = \begin{cases} q_{i+1} & \text{si } 1 \leq i < n \\ q_1 & \text{si } i = n + 1 \\ q_{n+2} & \text{si } i = n \text{ o } i = n + 2 \end{cases}$$

$$\delta(q_i, \#) = \begin{cases} q_{n+1} & \text{si } 1 \leq i \leq n \\ q_{n+2} & \text{si } n + 1 \leq i \leq n + 2 \end{cases}$$

Con lo anterior se define \mathcal{A}_n . Por claridad, se denota q_{n+2} como un estado ‘basura’, q_{n+1} como un ‘comienzo’¹ y q_i con $1 \leq i \leq n$ como un estado de i ‘a’. Ahora, sea $w \in L_n$, se nota que $w = \#a^i \cdot z$ con $z \in L_n$, donde \cdot es la operación de concatenación. Dado lo anterior, se nota que es suficiente demostrar que ε es aceptada por \mathcal{A}_n y que dado una palabra $w = \#a^i \cdot z$ en L_n , con $z \in L_n$ aceptada por \mathcal{A}_n , es aceptada por

¹En el sentido de que representa un string que comienza con #

\mathcal{A}_n . Lo anterior se puede expresar en la siguiente inducción sobre k . Para el caso base, si $k = 0$, se tiene la palabra vacía ε , como es la ejecución vacía se queda en q_0 el cual pertenece a F , por lo que la palabra es aceptada. Ahora, sea $w = \#a^{i_{k+1}} \cdot z$ donde $z = \#a^{i_k} \dots \#a^{i_1}$ y los $1 \leq i_j \leq n$, se tiene que z es aceptada por \mathcal{A}_n por hipótesis inductiva. Luego, se procesa w , comenzando en q_0 se tiene la secuencia de ejecución es $q_n \xrightarrow{\#} q_{n+1} \xrightarrow{a} q_1$ ahora se tiene i_{k+1} ‘a’, por lo que se tiene que de $q_1 \xrightarrow{a^{i_{k+1}-1}} q_{i_{k+1}}$ donde $i_{k+1} \leq n$, con lo que se empieza la ejecución de z en el estado $q_{i_{k+1}}$, y notamos que $\delta(q_j, \#) = q_n$ para $1 \leq j \leq n$, lo cual es cierto en nuestro caso, por lo que se usa la hipótesis inductiva, y se tiene que para z hay una ejecución que la acepta. Juntando esto con la ejecución de $\#a^{i_{k+1}}$ se tiene una ejecución que acepta a w , con lo que se cumple la hipótesis inductiva para $k + 1$.

Con lo anterior se tiene que $L_n \subseteq \mathcal{L}(\mathcal{A}_n)$. Para demostrar la otra contención, sea $w \in \Sigma^* \setminus L_n$, como $w \notin L_n$, se tiene que $w = z_1 \cdot \#\# \cdot z_2$, $w = z_1 \cdot a^{n+1} \cdot z_2$, $w = z_1 \cdot \#$ o $w = a \cdot z_1$ donde $z_1, z_2 \in \Sigma^*$.

- I. Para el primer caso, s.p.d.g. se asume que hay una ejecución de z_1 tal que no termine en q_{n+2} . Dado eso, se tiene que el estado en el que se empieza a evaluar $\#\# \cdot z_2$ es q_i con $1 \leq i \leq n + 1$. Si $i = n + 1$ se tiene que $\delta(q_{n+1}, \#) = q_{n+2}$, por lo que se entra al estado ‘basura’, con lo que w no es aceptada. En cambio, si $i \leq n$ se tiene que $\delta(q_i, \#) = q_{n+1}$, pero ahora volvemos a la situación anterior, ya que $\delta(q_{n+1}, \#) = q_{n+2}$, por lo que w no es aceptada.
- II. Al igual que el caso anterior, se asume que existe una ejecución de z_1 tal que no termine en q_{n+2} . Por lo que el estado donde se empieza a evaluar $a^{n+1} \cdot z_2$ es q_i con $1 \leq i \leq n + 1$. Si $i \leq n$, se tiene que al llegar a $a^{n+1-(n-i)} \cdot z_2$ se está en el estado q_n , pero el siguiente carácter es a , y $\delta(q_n, a) = q_{n+2}$. En el otro caso, $i = n + 1$ con lo que $\delta(q_{n+1}, a) = q_1$ y queda $a^n \cdot z_2$ para procesar, similarmente al caso recién mencionado se llega a $a \cdot z_2$ en el estado q_n , pero $\delta(q_n, a) = q_{n+2}$.
- III. Como se menciono en los casos anteriores se asume que existe una ejecución de z_1 que no termina en q_{n+2} . Ahora, si $i = n + 1$ se tiene que $\delta(q_{n+1}, \#) = q_{n+2}$, en cambio si $i \leq n$ se tiene que $\delta(q_i, \#) = q_{n+1}$ y ninguno es un estado de aceptación, por lo que w no es aceptada.
- IV. Como $q_0 = q_n$ y $\delta(q_n, a) = q_{n+2}$ se tiene que w no es aceptada.

Como $w \notin \mathcal{L}(\mathcal{A}_n)$, se tiene que $L_n = \mathcal{L}(\mathcal{A}_n)$.

- (b) Para construir \mathcal{B}_n se toma \mathcal{A}_n y se quita el estado basura y las transiciones asociadas al mismo. Se nota que este NFA tiene $n + 1$ estados y que cuando se llega a una transición

no definida la palabra no es aceptada, se puede usar la misma explicación de la parte anterior, con la diferencia de que en vez de hacer la transición a q_{n+2} y esperar a que se termine de evaluar la palabra, inmediatamente se ve que la palabra no es aceptada. ■

Problema 2:

- (a) Demuestre que para todo lenguaje regular L con $\varepsilon \notin L$, existe un autómata finito no-determinista $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ tal que $L = \mathcal{L}(\mathcal{A})$, $|I| = 1$ y $|F| = 1$.
- (b) Demuestre que existe un lenguaje regular L con $\varepsilon \notin L$, tal que para todo autómata finito determinista $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ con $L = \mathcal{L}(\mathcal{A})$ se cumple que $|F| \geq 2$.

Solución problema 2:

- (a) Dado un lenguaje regular L tal que $\varepsilon \notin L$. Por definición se tiene que existe un DFA \mathcal{A} tal que $\mathcal{L}(\mathcal{A}) = L$, como todo DFA es un NFA, se tiene que existe un NFA \mathcal{A} tal que $\mathcal{L}(\mathcal{A}) = L$. Ahora, como $\varepsilon \notin L$ se sabe que $F \cap I = \emptyset$. Dado eso, se construye \mathcal{A}' de la siguiente forma, usando \mathcal{A} como base se agregan dos estados, q_α, q_Ω , el único estado inicial y el estado final respectivamente. Ahora, para cada estado inicial de \mathcal{A} se ven las transiciones que salen de ellos y se agregan a las transiciones de q_α , i.e. para cada $q_i \in I_{\mathcal{A}}$, $q_j \in Q_{\mathcal{A}}$ y $a_k \in \Sigma$ tal que $\Delta_{\mathcal{A}}(q_i, a_k, q_j)$ se tiene que $\Delta_{\mathcal{A}'}(q_\alpha, a_k, q_j)$, pero con el $q_j \in Q_{\mathcal{A}'}$ correspondiente. Similarmente, se construye lo mismo para los $q_i \in F_{\mathcal{A}}$, $q_j \in Q_{\mathcal{A}}$ y $a_k \in \Sigma$, i.e. en \mathcal{A}' se tiene $\Delta_{\mathcal{A}'}(q_j, a_k, q_\Omega)$. Ahora, para demostrar que $\mathcal{L}(\mathcal{A}') = L$ se hace la doble contención con $\mathcal{L}(\mathcal{A})$. Entonces, dado $w \in \mathcal{L}(\mathcal{A})$ se tiene que existe una ejecución ρ por \mathcal{A} tal que w es aceptada, ρ se ve la de siguiente manera $p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_{n-1}} p_{n-1} \xrightarrow{a_n} p_n$, donde $p_0 \in I_{\mathcal{A}}$ y $p_n \in F_{\mathcal{A}}$, entonces se tiene que $\rho' : q_\alpha \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_{n-1}} p_{n-1} \xrightarrow{a_n} q_\Omega$ es una ejecución de \mathcal{A}' sobre w , ya que $\Delta_{\mathcal{A}}(p_0, a_1, p_1)$ se tiene si y solo si $\Delta_{\mathcal{A}'}(q_\alpha, a_1, p_1)$, similarmente $\Delta_{\mathcal{A}}(p_{n-1}, a_n, p_n)$ se tiene si y solo si $\Delta_{\mathcal{A}'}(p_{n-1}, a_n, q_\Omega)$. Como $q_\Omega \in F$, se tiene que $w \in \mathcal{L}(\mathcal{A}')$, por lo que $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$ ². Pero el argumento se puede hacer al revés, o sea, dado una ejecución ρ' de \mathcal{A}' que acepta $w \in \mathcal{L}(\mathcal{A}')$ se tiene que $\rho' : q_\alpha \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_{n-1}} p_{n-1} \xrightarrow{a_n} q_\Omega$, ahora por construcción existe $p_0 \in I_{\mathcal{A}}$ tal que $\Delta_{\mathcal{A}}(p_0, a_1, p_1)$, y se tiene que existe $p_n \in F_{\mathcal{A}}$ tal que $\Delta_{\mathcal{A}}(p_{n-1}, a_n, p_n)$, por lo que $\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_{n-1}} p_{n-1} \xrightarrow{a_n} p_n$ es una ejecución de \mathcal{A} que acepta w . Por lo que se tiene que $\mathcal{L}(\mathcal{A}') = \mathcal{L}(\mathcal{A}) = L$. Más aún se tiene que $|I_{\mathcal{A}'}| = |F_{\mathcal{A}'}| = 1$.

²Como $F_{\mathcal{A}} \cap I_{\mathcal{A}} = \emptyset$ toda ejecución de \mathcal{A} es de largo al menos uno y pasa por al menos dos estados, por lo que ρ' siempre está bien definida.

(b) Sea $\Sigma = \{a\}$ y $L = \{a, aa\}$, sea \mathcal{A} un DFA tal que $\mathcal{L}(\mathcal{A}) = L$. Luego, existe una ejecución ρ_a de \mathcal{A} sobre a que la acepta. Como $|a| = 1$ entonces $\rho_a : q_0 \xrightarrow{a} p_1$ donde q_0 es el estado inicial y $p_1 \in F$. Similarmente, existe una ejecución ρ_{aa} que acepta aa , vemos que $\rho_{aa} : q_0 \xrightarrow{a} p_1 \xrightarrow{a} p_2$, donde el p_1 es el mismo estado que en la ejecución ρ_a ³ y $p_2 \in F$. Luego, se tienen dos opciones $p_1 \neq p_2$ y $p_1 = p_2$ con la primera se tiene que $|F| \geq 2$, consiguiendo lo pedido. Por lo tanto se observará el segundo caso, se nota que $\delta(p_1, a) = p_1$, ahora sea ρ_{aaa} la ejecución de \mathcal{A} sobre la palabra aaa , se tiene que $\rho_{aaa} : q_0 \xrightarrow{a} p_1 \xrightarrow{a} p_1 \xrightarrow{a} p_1$, y como $p_1 \in F$ se tiene que aaa es aceptada por \mathcal{A} , pero se recuerda que $aaa \notin L$ y $\mathcal{L}(\mathcal{A}) = L$, lo que es una contradicción. Se tiene entonces que L cumple lo pedido. ■

³Si fuera distinto, δ no sería función, y por ende \mathcal{A} no sería DFA.