UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
Facultad de Ciencias

Autor: Adrián Aguilera Moreno

Autómatas y Lenguajes Formales

Tarea 1

1. Dados $0 \leq m < k$ y $2 \leq p$, sea

$$A_{k,m,p} = \{a \in \{0, 1, \cdots, p-1\}^* | a \text{ es una representación } p\text{-aria de } x \text{ y } x \textbf{ mod } k = m\}.$$

Da un método general para construir un autómata que acepte $A_{k,m,p}$.

$\triangledown$ **Solución:** Sea α una cadena p-aria, entonces $\#\alpha$ es el número que representa a esta cadena en notación decimal. De lo anterior se propone la función de transición de cadenas

$$\delta^*(0, \alpha) = \#\alpha \textbf{ mod } k$$

Así, cuando se quiera construir un autómata se debe considerar como estado final a los residuos indicados[1] de manera explícita.

Entonces $A_{k,m,p} = \langle Q, \Sigma, \delta, s, F\rangle$ tales que

$$\begin{array}{rcll} Q & = & \{0, \cdots, k-1\} & \text{Residuos al dividir entre } k. \\ \Sigma & = & \{0, \cdots, p-1\} & \text{Los posibles valores de la notación.} \\ F & = & \{\text{Los } m \text{ (residuos) indicados}\} & \end{array}$$

Definimos

$$\begin{array}{rcl} \#(\alpha d) & = & p \cdot (\#\alpha) + d \\ \delta(q, d) & = & (pq + d) \textbf{ mod } k \end{array}$$

donde $d \in 0, 1, \cdots, p-1$ es alguna transición y q un estado en el autómata.

Ahora mostremos lo anterior por inducción en α [en la cadena p-aria], esto es

$$\begin{array}{rcll} \delta^*(0, \lambda) & = & 0 & \text{Por definición de } \delta. \\ & = & \#\lambda & \text{Pues } \#\lambda = 0. \\ & = & \#\lambda \textbf{ mod } k & \text{Sabemos que } 0 \; mod \; k = 0. \end{array}$$

y como se puede ver, la cadena vacia cumple con nuestra definición. Ahora, supongamos sin pérdida de generalidad que nuestra condición se cumple para alguna cadena p-aria μ, *i.e.*

$$\delta^*(0, \mu) = \#\mu \textbf{ mod } k \qquad \text{H.I.}$$

veamos que pasa con la cadena μd, donde $d \in \{0, 1, \cdots, p-1\}$, así

$$\begin{array}{rcll} \delta^*(0, \mu) & = & \delta(\delta^*(0, \mu), d) & \text{Definición alternativa de } \delta^*. \\ & = & \delta(\#\mu \textbf{ mod } k, d) & \text{Hipótesis de inducción.} \\ & = & (p \cdot (\#\mu \textbf{ mod } k) + d) \textbf{ mod } k & \text{Definición de } \delta. \\ & = & (p \cdot \#\mu + d) \textbf{ mod } k & \text{Teoría de Números.} \\ & = & \#\mu d \textbf{ mod } k & \text{Definición dada previamente.} \end{array}$$

Ahora, veamos un ejemplo de esta construcción:

"Diseña un autómata que acepte números en notación ternaria que al ser divididos entre 3 dejen residuo de 2". Resolviendo tenemos que: Sea $A_{3,2,3}$ un autómata tal que resuelve el problema anteriormente planteado, así el estado terminal (en este caso, el único estado terminal) es 2.

[1] En caso de pedir que un valor sea múltiplo, el estado terminal debe ser 0.

Observemos los primeros 15 valores en notación terciaria

$0 - 0$	$11 - 4$	$22 - 8$	$110 - 12$
$1 - 1$	$12 - 5$	$100 - 9$	$111 - 13$
$2 - 2$	$20 - 6$	$101 - 10$	$112 - 14$
$10 - 3$	$21 - 7$	$102 - 11$	$120 - 15$

esto nos ayudará a encontrar las transiciones (al usar δ con algunas de estas cadenas terciarias). Veamos en la siguiente tabla la relación de módulos con $k = 3$:

Expresión	Residuo
0 **mod** 3	0
1 **mod** 3	1
2 **mod** 3	2
3 **mod** 3	0

Nótese que

$$0 \textbf{ mod } 3 \equiv 3 \textbf{ mod } 3$$

por tanto consideraremos solo los tres primeros casos, también es claro que la columna de Residuos son los estados en $A_{3,2,3}$. A continuación se muestra la tabla de transición de $A_{3,2,3}$:

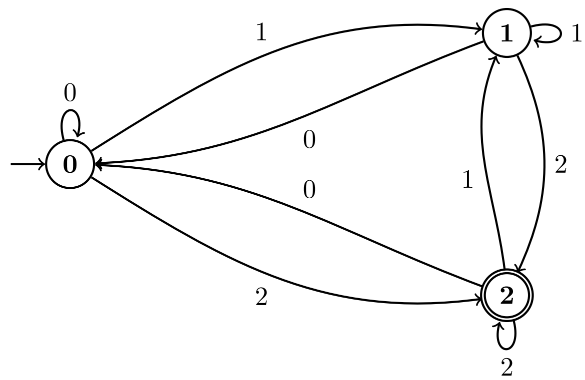
Q	**0**	**1**	②
$\rightarrow$ **0**	0	1	2
1	0	1	2
2	0	1	2

Obs. En este caso "$\rightarrow$ **0**" indica el estado inicial en el autómata, mientras que ② indica el estado terminal.

Así, podemos comprobar la información en la tabla de transición por medio de la función δ^*, esto es

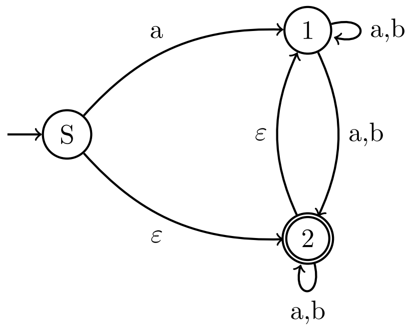
$\delta(0) = 0$	$\delta^*(11) = 1$	$\delta^*(22) = 2$	$\delta^*(110) = 0$
$\delta(1) = 1$	$\delta^*(12) = 2$	$\delta^*(100) = 0$	$\delta^*(111) = 1$
$\delta(2) = 2$	$\delta^*(20) = 0$	$\delta^*(101) = 1$	$\delta^*(112) = 2$
$\delta^*(10) = 0$	$\delta^*(21) = 1$	$\delta^*(102) = 2$	$\delta^*(120) = 0$

Por último, veamos la representación de $A_{3,2,3}$ como una gráfica de estados



◁

2. Transforma el siguiente autómata no determinista con transiciones-ε en uno determinista usando los métodos vistos en clase.



$\triangledown$ **Solución:** Sea $A_\varepsilon = \langle Q, \Sigma, \Delta, S, F\rangle$ el autómata representado por la gráfica de estados anterior, y sea $Q = \{S, 1, 2\}$ el conjunto de estados, entonces

$$\mathcal{P}(Q) = \{\phi, \{S\}, \{1\}, \{2\}, \{S,1\}, \{S,2\}, \{1,2\}, \{S,1,2\}\}$$

Veamos las transiciones a través de la función δ_D [que como sabemos, existe una equivalencia $\delta_D = \Delta^* = \Delta^*_\varepsilon$], así

$$\begin{array}{cc}
\Delta^*_\varepsilon(\{S\}, a) = \{1\} & \Delta^*_\varepsilon(\{S\}, b) = \phi \\
\Delta^*_\varepsilon(\{1\}, a) = \{1,2\} & \Delta^*_\varepsilon(\{1\}, b) = \{1,2\} \\
\Delta^*_\varepsilon(\{2\}, a) = \{2\} & \Delta^*_\varepsilon(\{2\}, b) = \{2\} \\
\Delta^*_\varepsilon(\{1,2\}, a) = \{1,2\} & \Delta^*_\varepsilon(\{1,2\}, b) = \{1,2\} \\
\Delta^*_\varepsilon(\{S,1\}, a) = \{1,2\} & \Delta^*_\varepsilon(\{S,1\}, b) = \{1,2\} \\
\Delta^*_\varepsilon(\{S,2\}, a) = \{1,2\} & \Delta^*_\varepsilon(\{S,2\}, b) = \{2\} \\
\Delta^*_\varepsilon(\{S,1,2\}, a) = \{1,2\} & \Delta^*_\varepsilon(\{S,1,2\}, b) = \{1,2\}
\end{array}$$

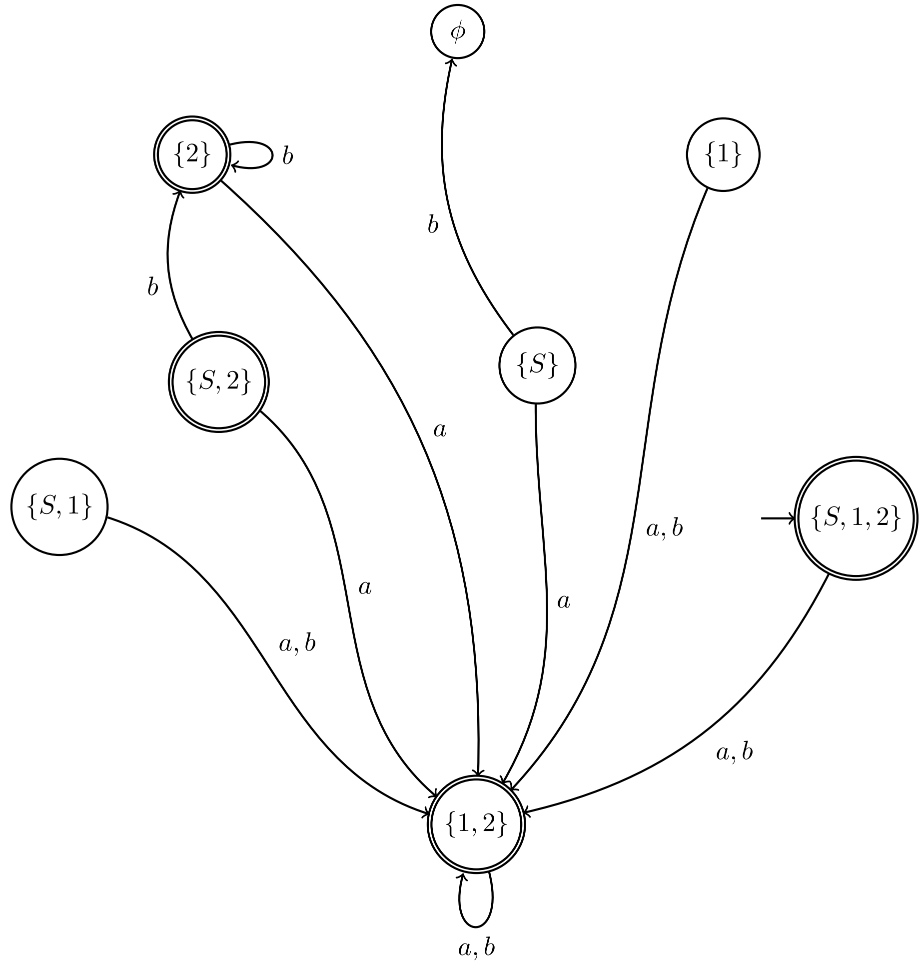
Es necesario hacer notar que $\Delta^*_\varepsilon(\{S\}, \varepsilon) = \{2\}$, como no queremos transiciones-ε, entonces reasignamos $\Delta^*_\varepsilon(\{S\}, a) = \{1,2\}$. Mismo caso con $\Delta^*_\varepsilon(\{2\}, \varepsilon) = \{1\}$, así reasignamos a $\Delta^*_\varepsilon(\{2\}, a) = \{1,2\}$.

Así, se presenta la tabla de transiciones del autómata A [el autómata en DFA resultado de transformar A_ε]:

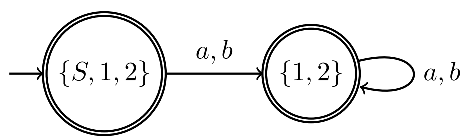
Q	$\{\mathbf{S}\}$	ϕ	$\{\mathbf{1}\}$	final: $\{\mathbf{2}\}$	final: $\{\mathbf{1,2}\}$	$\{\mathbf{S,1}\}$	final: $\{\mathbf{S,2}\}$	final: $\{\mathbf{S,1,2}\}$
$\rightarrow \{\mathbf{S}\}$	–	b	a	ε	–	–	–	–
ϕ	–	–	–	–	–	–	–	–
$\{\mathbf{1}\}$	–	ε	–	–	a,b	–	–	–
$\{\mathbf{2}\}$	–	–	ε	a,b	–	–	–	–
$\{\mathbf{1,2}\}$	–	–	ε	–	a,b	–	–	–
$\{\mathbf{S,1}\}$	–	–	–	ε	a,b	–	–	–
$\{\mathbf{S,2}\}$	–	–	–	b	a,ε	–	–	–
$\{\mathbf{S,1,2}\}$	–	–	–	–	a,b,ε	–	–	–

TABLA EN MODIFICACIÓN.

Luego, definamos $A = \langle Q_D, \Sigma, \delta_D, s_D, F_D\rangle$ cuya gráfica de estados se presenta a continuación



Como en A existen estados inaccesibles, podemos precindir de estos, lo que nos genera una nueva gráfica de estados de A, esta es

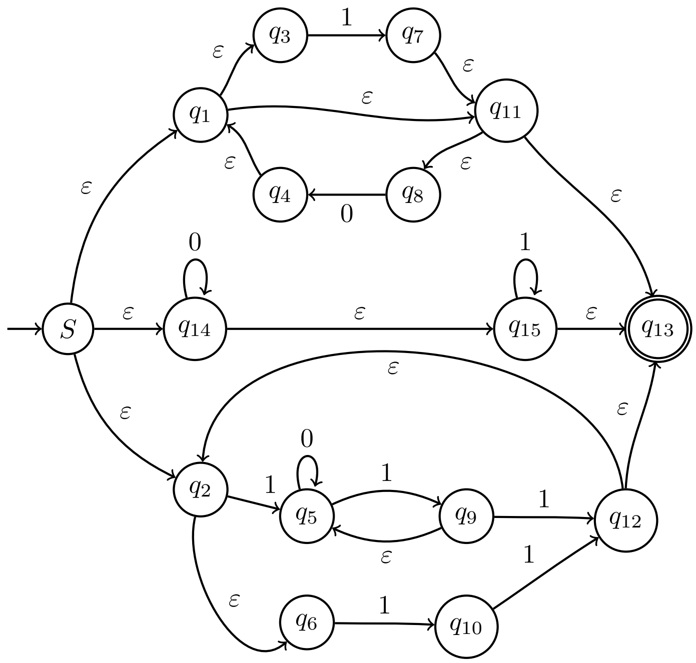


$\triangleleft$

3. Construye un autómata que acepte el lenguaje generado por la expresión regular siguiente

$$(1 + 1(0^*1)^*1)^* + (0^*1^* + (10)^*)$$

$\triangledown$ **Solución:** A continuación se muestra la gráfica de transiciones del autómata que acepta la expresión regular



Así, el autómata referente a la anterior gráfica de transiciones es $A_\varepsilon = \langle Q, \Sigma, \delta, S, F\rangle$, donde $Q = \{S, q_0, q_1, \cdots, q_{14}, q_{15}\}$, $\Sigma = \{0, 1\}$, $\delta = (1 + 1(0^*1)^*1)^* + (0^*1^* + (10)^*)$, $S = \{S\}$ y $F = \{q_{13}\}$. $\triangleleft$

4. Da una gramática que genere el lenguaje aceptado por el autómata del ejercicio 1.

$\triangledown$ **Solución:** Sea $G_1 = \langle \Sigma, \Gamma, S, \rightarrow_{G_1}\rangle$ la gramática que genera el lenguaje aceptado por el autómata del ejercicio 1, entonces

$$\begin{aligned} \Sigma &= \{0, 1, \cdots, p-1\} \\ \Gamma &= \{0, 1, \cdots, k-1\}/\{m\} \\ S &= \{0\} \\ \rightarrow_{G_1} &= \{S \rightarrow mS, S \rightarrow (m + a * k)S, S \rightarrow (m + a * k), S \rightarrow m\} \end{aligned}$$

donde $a \in \mathbb{N}$ y m, k corresponden a la definición dada en el ejercicio 1. $\triangleleft$