Tarea 1: Lenguajes Regulares

CC3102: Teoría de la Computación

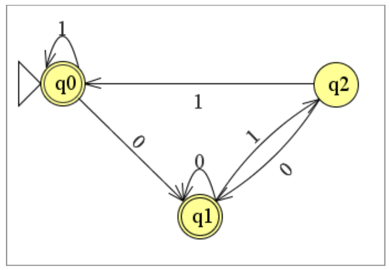
Departamento de Ciencias de la Computación
Universidad de Chile
Gabriel de la Parra

06 de octubre de 2015

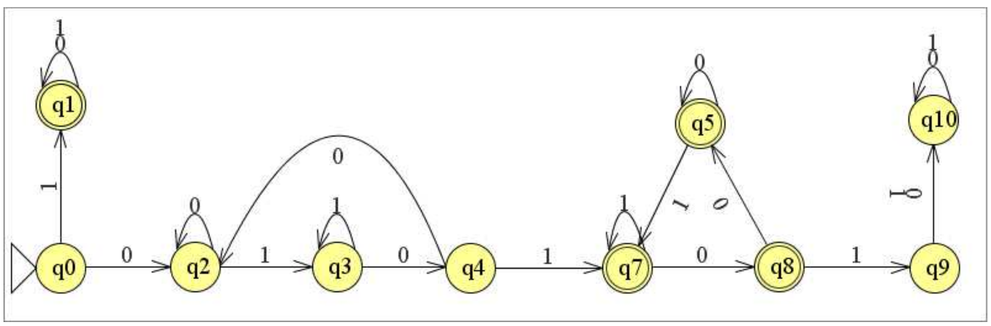
Problema 1

PARA CADA UNA DE LOS SIGUIENTES LENGUAJES, DIBUJE EL DIAGRAMA DE ESTADOS DEL AUTÓMATA FINITO DETERMINISTA QUE RECONOZCA EL LENGUAJE. TODOS LOS LENGUAJES SON SOBRE EL ALFABETO {0,1}.

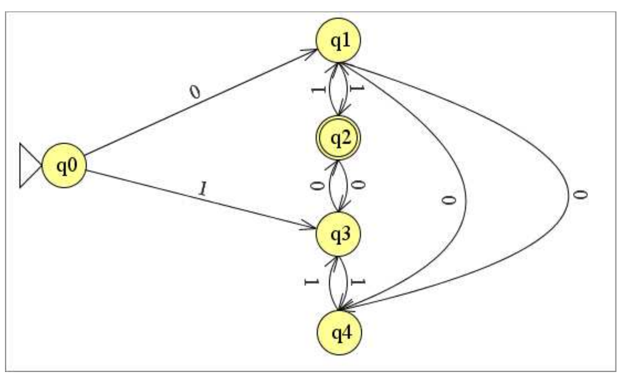
L1 = {w|w no termina en 01}



L2 = {w|w tiene sólo un substring 101 o comienza con 1 }



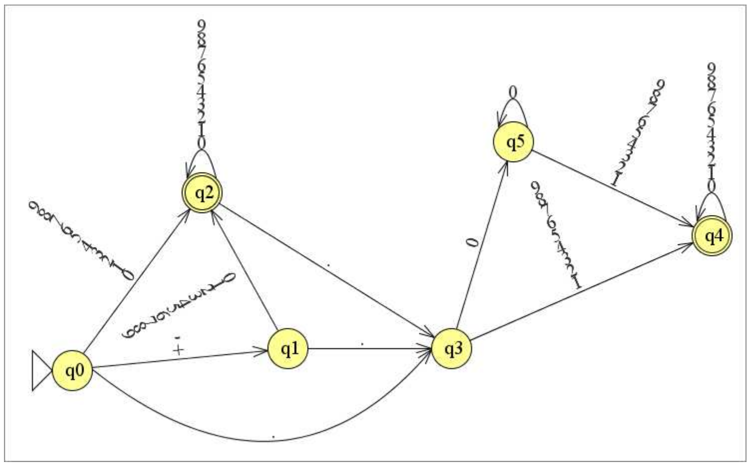
L3 = {w|w es tal que tanto el número de 0's como el número de 1's es impar }



Problema 2

Considere el alfabeto $\Sigma = \{0,1,\ldots,9,+,-,.\}$. Entregue el AFND N para el lenguaje R de todas las cadenas sobre Σ que representan correctamente un número decimal con signo. Por ejemplo, $+3.2, 10.01, .78, -.1415, 47, +1$ pertenecen al lenguaje R. Convertir el AFND en un AFD por método visto en clase.

La construcción del AFND $N = \{Q, \Sigma, \delta, q_0, F\}$ se presenta a continuación:



Con:

$$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$$

$$\Sigma = \{0,1,2, \ldots ,9, +, -, .\}$$

$$\delta: Qx\Sigma \to Q$$

Dado por la siguiente tabla de transiciones:

	$\mathbf{1,2,\ldots 9}$	$\mathbf{0}$	$+,-$	$.$
$\boldsymbol{q_0}$	q_2	q_2	q_1	q_3
$\boldsymbol{q_1}$	q_2	q_2	$\boldsymbol{q_6}$	q_3
$\boldsymbol{q_2}$	q_2	q_2	$\boldsymbol{q_6}$	q_3
$\boldsymbol{q_3}$	q_4	q_5	$\boldsymbol{q_6}$	$\boldsymbol{q_6}$
$\boldsymbol{q_4}$	q_4	q_4	$\boldsymbol{q_6}$	$\boldsymbol{q_6}$
$\boldsymbol{q_5}$	q_4	q_5	$\boldsymbol{q_6}$	$\boldsymbol{q_6}$

$$y\ F = \{q_2, q_4\}$$

Notar que q_6 se ha omitido del diagrama de estados por simplicidad de diseño y estética, sin embargo se llega a éste a través de las transiciones definidas en la tabla.

El autómata entregado en la primera parte ya corresponde a un AFD. Sin embargo, suponiendo que fuese un AFND y tuviese transiciones en ε, la conversión del AFND a AFD debe incluir las clausuras sobre ε.
La idea detrás de la conversión es a agrupar estados que incluyan transiciones con ε y rehacer los conjuntos del AFND.

Así, sea M el AFD dado por:

$$M = \{Q', \Sigma, \delta', q_0', F'\}$$

Tal que Q' es el conjunto potencia de Q:

$$Q' = \mathcal{P}(Q) = \{\emptyset, \{q_0\}, \{q_1\}, \dots, \{q_0, q_1\}, \{q_0, q_2\}, \dots, \{q_0, q_1, q_2, q_3, q_4, q_5\}\}$$

Que se puedan alcanzar por transiciones δ o ε.
Sin transiciones con ε se tiene que

$$Q' = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\} = Q$$

Con δ' dado por todas las transiciones, incluyendo las que se pueden llegar con ε.
Así, para un estado $R \in Q', a \in \Sigma$:

$$\delta'(R, a) = \bigcup_{r \in R} E(\delta(r, a))$$

Y al no haber transiciones con ε:

$$\delta' = \delta$$

Para el estado inicial q_0', formado por los estados de Q a los que se puede llegar con ε desde q_0

$$q_0' = E(\{q_0\}) = \ q_0$$

F' Los estados R de Q' que tienen algún estado final de Q:

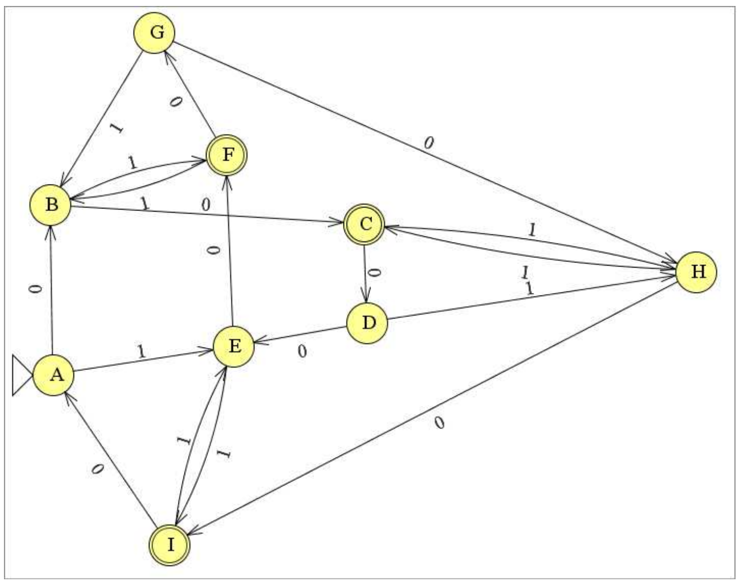
$$F' = \{q_2, q_4\}$$

Problema 3

CONSIDERE EL AFD DESCRITO POR $M = (Q, \Sigma, \delta, A, F)$, DONDE $Q = \{A, B, C, D, E, F, G, H, I\}, \Sigma = \{0,1\}, F = \{C, F, I\}$, Y LA FUNCIÓN $\delta: Q \times \Sigma \to Q$ ESTÁ DEFINIDO POR LA SIGUIENTE TABLA:

	0	1
A	B	E
B	C	F
C	D	H
D	E	H
E	F	I
F	G	B
G	H	B
H	I	C
I	A	E

Para entregar el AFD M^* equivalente a M con mínimo número de estados, se considera primero que el autómata M se puede representar por el siguiente diagrama de estados:



Para minimizar, lo primero es eliminar los estados que no tienen transiciones hacia ellos. En este caso no existe ninguno.

Posteriormente se procede a buscar los estados equivalentes. En una tabla de equivalencia de estados se pueden marcar aquellos que son finales de los no finales. Así, en una primera instancia se tendrá:

A									
B									
C	X	X							
D			X						
E			X						
F	X	X		X	X				
G			X						
H			X						
I	X	X		X	X		X	X	
	A	B	C	D	E	F	G	H	I

Al buscar la equivalencia entre las transiciones de los estados sin marcar, se puede construir la siguiente tabla auxiliar, con negrita los estados que ya están marcados en la tabla anterior:

	0	1
A,B	**B,C**	**E,F**
A,D	B,E	E,H
A,E	**B,F**	**E,I**
A,G	B,H	E,B
A,H	**B,I**	**E,C**
B,D	**C,E**	**F,H**
B,E	C,F	F,I
B,G	**C,H**	**F,B**
B,H	C,I	F,C
D,E	E,F	**H,I**
D,G	E,H	H,B
D,H	E,I	**H,C**
E,G	**F,H**	**I,B**
E,H	F,I	I,C
G,H	**G,I**	**B,C**

Al agregar los estados que producen estos estados en negrita, se puede obtener la siguiente tabla de equivalencia:

A									
B	X								
C	X	X							
D		X	X						
E	X		X	X					
F	X	X		X	X				
G		X	X		X	X			
H	X		X	X		X	X		
I	X	X		X	X		X	X	
	A	B	C	D	E	F	G	H	I

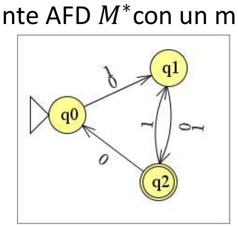
Agrupando por fila/columna, se pueden distinguir las siguientes tuplas:

$$q_0 = \{A, D, G\}, el\ estado\ inicial$$
$$q_1 = \{B, E, H\}$$
$$q_2 = \{C, I, F\}, el\ estado\ final$$

Al hacer las transiciones entre estos estados se tiene lo siguiente:

$$\delta(q_0, 0) = q_1$$
$$\delta(q_0, 1) = q_1$$
$$\delta(q_1, 0) = q_2$$
$$\delta(q_1, 1) = q_2$$
$$\delta(q_2, 0) = q_0$$
$$\delta(q_2, 1) = q_1$$

De esta manera se puede construir el siguiente AFD M^*con un mínimo de estados de la siguiente forma:



Problema 4

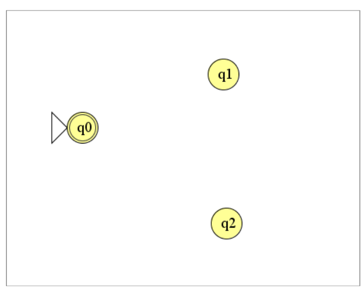
DEMUESTRE QUE, PARA TODO $m \geq 1$, EL LENGUAJE $T_m = \{x | x\ es\ un\ numero\ binario\ multiplo\ de\ m\}$ ES REGULAR. PARA ELLO, MUESTRE UN AFD PARA T_m Y DEMUESTRE QUE EFECTIVAMENTE SU AFD RECONOCE EL LENGUAJE. B) DEMUESTRE LA CORRECTITUD.

Para poder crear un AFD para T_m, se partirá por explicar un caso de ejemplo. Suponer un $m = 3$. Y considerando la siguiente tabla, los números binarios desde 0 hasta 9, y su residuo sobre 3:

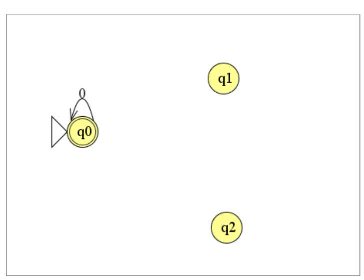
x	*Binario*	*r(x/m)*
0	*0*	*0*
1	*1*	*1*
2	*10*	*2*
3	*11*	*0*
4	*100*	*1*
5	*101*	*2*
6	*110*	*0*
7	*111*	*1*
8	*1000*	*2*
9	*1001*	*0*

El residuo de $\frac{x}{3}$ puede tener 3 valores $\{0,1,2\}$. Se puede construir un AFD que acepte T_3 creando 3 estados para cada uno de los residuos y generando las transiciones para todos los estados hasta completar todas las transiciones sobre el alfabeto $\Sigma = \{0,1\}$, es decir, 2 transiciones por cada estado, donde cada estado indica el residuo de $x/3$. Nótese que $\frac{0}{3} = 0$ por lo que el estado q_0 debe ser estado final.

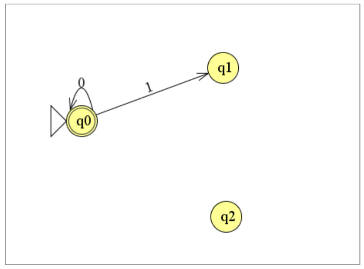
Así, inicialmente se tiene un AFD dado por:



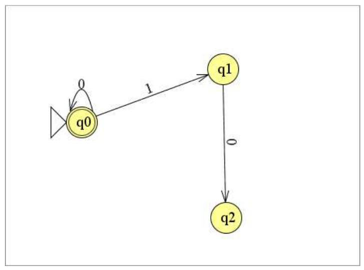
Al avanzar por los números de la tabla, se puede ir trazando las transiciones, considerando que se debe llegar al estado q_i con i el residuo.
De esta forma para x=0 (binario 0), el residuo r=0, se debe ir al estado q_0.con el binario dado. Así $\delta(q_0, 0) \rightarrow q_0$:



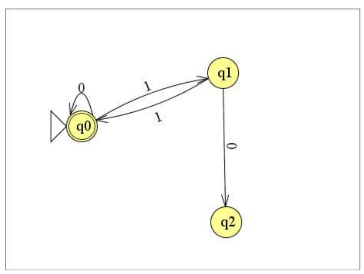
Para x=1 (binario 1), r=1, por lo que se puede trazar una transición $\delta(q_0, 1) \rightarrow q_1$:



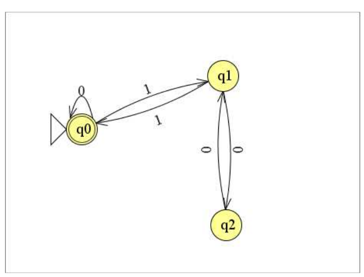
Para x=2 (10), r=2, por lo que se debe ir al estado q_2. $\delta(q_0, 10) = \delta(\delta(q_0, 1), 0) = \delta(q_1, 0) \rightarrow q_2$



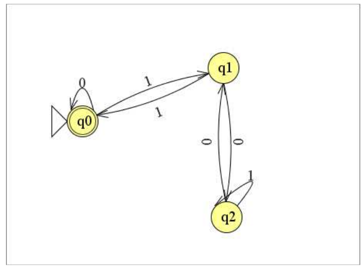
Para x=3 (11), r=0: $\delta(q_0, 11) = \delta(\delta(q_0, 1), 1) = \delta(q_1, 1) \rightarrow q_0$



El procedimiento se puede completar hasta que se complete el AFD, es decir, sean 2m transiciones. Esto ocurrirá en el número 2m-1.
Para el caso x=4; r=1; $\delta(q_0, 100) \rightarrow q_1$



Llegado a este punto, 2m-1=5; r=2, $\delta(q_0, 101) \rightarrow q_2$



Se tiene completo el AFD que reconoce T_m.

Ahora, para un caso general, para cualquier número x, se tiene que

$$x = q * m + r\ ;\ 0 < r < m - 1\ ;\ q \geq 0$$

Así, para un m dado, se puede construir el autómata A_m de la forma

$$A_m = \{Q, \Sigma, \delta, q_0, F\}$$

Dónde Q tiene m estados: $Q = \{q_0, q_1, \ldots, q_{m-1}\}$ donde cada índice m de q_m indica el residuo de $\frac{x}{m}$.para el largo actual de la cadena que se está recorriendo.

Las transiciones se pueden construir como se vio anteriormente, tomando en consideración que para cada número x, se debe llegar al estado q_i donde i es el residuo de x.

Como $\frac{0}{m} = 0, r = 0$, luego para cualquier A_m, q_0 es el estado inicial. $F = \{q_0\}$

Se debe demostrar que una palabra está dentro del lenguaje si y solo si, ésta puede ser reconocido por A_m

$$w \in T_m \Leftrightarrow \mathrm{w} \in \mathcal{L}(A_m)$$

Para demostrar $w \in T_m \Leftarrow \mathrm{w} \in \mathcal{L}(A_m)$:
El AFD tiene transiciones $\delta(q_i, w_{i+1}) \rightarrow q_{i+1}$por las que se procesa la palabra. Cabe notar que $\delta(q_0, x\ múltiplo\ de\ m) \rightarrow q_0 \in F$ y que para otro x, el autómata no acepta, ya que las demás transiciones no son estados finales. Se puede argumentar que el AFD A_m tiene la característica de volver a su estado inicial luego de procesar la palabra sobre los m estados. Es decir, para una cadena cualquiera que representa un número x, existen transiciones dadas que hacen que la cadena se procese por las mismas transiciones de forma cíclica. Lo anterior se podía demostrar por inducción.

Así también se debe demostrar $w \in T_m \Rightarrow \mathrm{w} \in \mathcal{L}(A_m)$:

Cualquier cadena w que representa un número se puede representar bajo una expresión regular, es decir, puede que tenga un prefijo o un sufijo o una subcadena que indique que el número es múltiplo de m. Así por ejemplo, los múltiplos de 2 terminan en 0, los múltiplos de 4 en doble cero. Se podría demostrar que para estos casos, la cadena pasa por transiciones de estos casos y que existe el A_m que contempla estas transiciones.

Problema 5

SI L ES UN LENGUAJE Y $a \in \Sigma$ UN SÍMBOLO, DEFINIMOS LA OPERACIÓN COCIENTE DE L Y a COMO $L/a \stackrel{\text{def}}{=} \{w \in \Sigma^* \mid wa \in L\}$ POR EJEMPLO, SI $L = \{a, aab, baa\}$, ENTONCES $L/a = \{\varepsilon, ba\}$. DEMUESTRE QUE SI L ES REGULAR ENTONCES L/a ES REGULAR. DEBE DEMOSTRAR QUE SU CONSTRUCCIÓN ES CORRECTA.

Sea L un lenguaje, tal que existe un AFD M que lo reconoce de la forma:

$$M = \{Q, \Sigma, \delta, q_0, F\}$$

Tal que el lenguaje que reconoce este AFD es L

$$\mathcal{L}(M) = L$$

Se puede demostrar, que para L, debe existir una transición sobre a, desde un estado q_{n-1} que lleve a $q_n \in F$:

$$\delta(q_{n-1}, a) = q_n$$

En tal caso, se puede construir un nuevo AFD, M' de la forma:

$$M' = \{Q', \Sigma, \delta', q_0', F'\}$$

En este, Q' son todos los estados de Q, sin q_n.
De la misma manera, δ' corresponden a todas las transiciones de M, sin la transición final de $\delta(q_{n-1}, a) \rightarrow q_n$
El estado inicial $q_0' = q_0$.
F' es el estado q_{n-1}.

Es evidente que si L acepta a wa, entonces existen las transiciones en L' de tal forma que $\delta^*(q_0', w) \rightarrow q_{n-1}$

De lo anterior, y al existir este autómata que reconoce a w, se podría concluir que $\mathcal{L}(M') =$, y que por lo tanto $L \backslash a$ es un lenguaje regular.

Ahora bien, para abordar este problema, se propuso inicialmente otro tipo de demostración. Esta demostración se considera que no debería ser válida para evaluación, sin embargo, si útil para ilustración. Considérese el supuesto que L es la concatenación de dos lenguajes regulares L_1 y L_2.

$$L = L_1 \cdot L_2$$

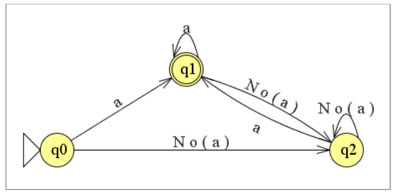
Con

$$L_1 = L/a$$

Y un L_2 que se puede definir como un el *lenguaje regular* reconocido por un AFD A que acepta palabras terminadas en a

$$L_2 = \{w \mid w\ termina\ en\ a\}$$

De la forma:



Se demostrará por contradicción que L_1 es regular. Asumiendo que L_1 es no regular, se tiene que de la concatenación

$$L = L_1 \cdot L_2$$

No se podría asegurar que L sea regular, sin embargo, como se indica que L es regular, L_1 no puede no ser regular.

Esto sin embargo, podría no ser suficiente, ya que se podía pensar en que L_1 sea irregular y que la concatenación de un irregular con un regular podría dar un lenguaje regular. Se sabe que la concatenación de dos lenguajes regulares es regular, sin embargo no se puede asegurar que la concatenación entre uno regular y uno irregular, sea regular.

Finalmente, si existe el autómata que reconoce a L, y se sabe que tiene un estado q_{n-1} y la transición $\delta(q_{n-1}, a) \rightarrow q_n$, un estado final, se puede concluir que efectivamente, existe una serie de transiciones δ^* tal que se pueda llegar hasta q_{n-1}. Con esto, se sabe que el camino existe, y por lo tanto, se puede inferir que es regular.