Tarea 1: Lenguajes Regulares

CC3102: Teoría de la Computación

Departamento de Ciencias de la Computación

Universidad de Chile

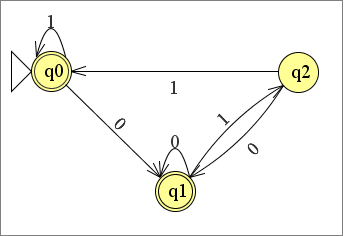
Gabriel de la Parra

06 de octubre de 2015

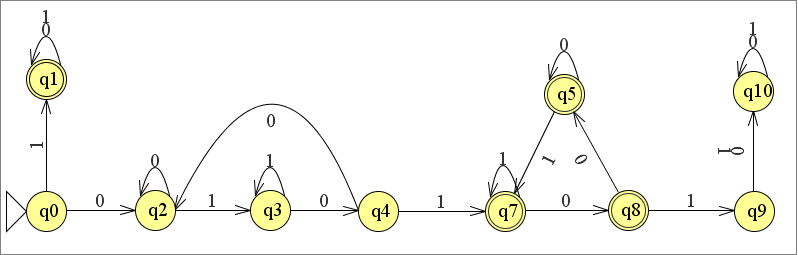
# Problema 1

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}.

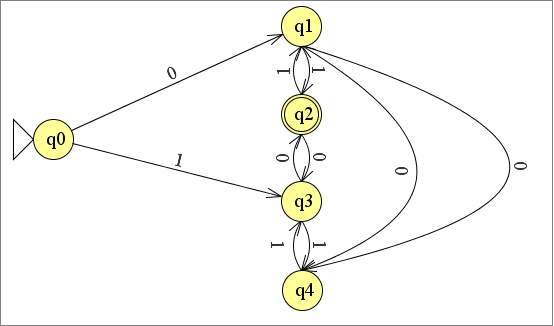
***L1 = {w|w no termina en 01}***



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



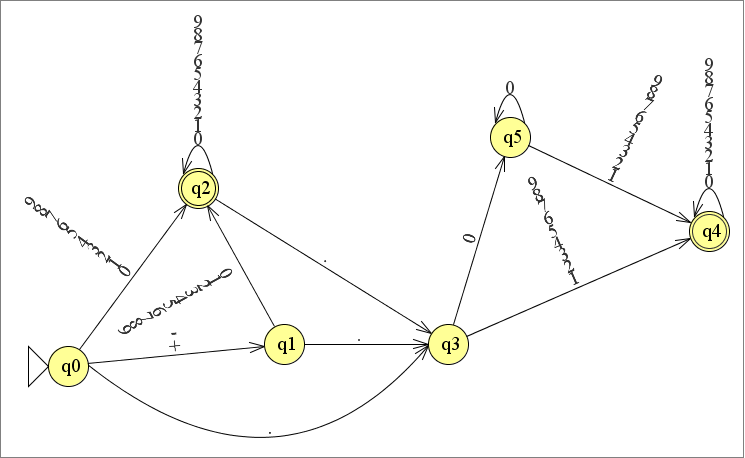
***L3 = {w|w es tal que tanto el número de 0’s como el número de 1’s es impar }***



# Problema 2

Considere el alfabeto . Entregue el AFND para el lenguaje de todas las cadenas sobre que representan correctamente un número decimal con signo. Por ejemplo, pertenecen al lenguaje . Convertir el AFND en un AFD por método visto en clase.

La construcción del AFND se presenta a continuación:



Con:

Dado por la siguiente tabla de transiciones:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Notar que 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 , la conversión del AFND a AFD debe incluir las clausuras sobre

La idea detrás de la conversión es a agrupar estados que incluyan transiciones con y rehacer los conjuntos del AFND.

Así, sea M el AFD dado por:

Tal que es el conjunto potencia de Q:

Que se puedan alcanzar por transiciones o .

Sin transiciones con se tiene que

Con dado por todas las transiciones, incluyendo las que se pueden llegar con .

Así, para un estado :

Y al no haber transiciones con :

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

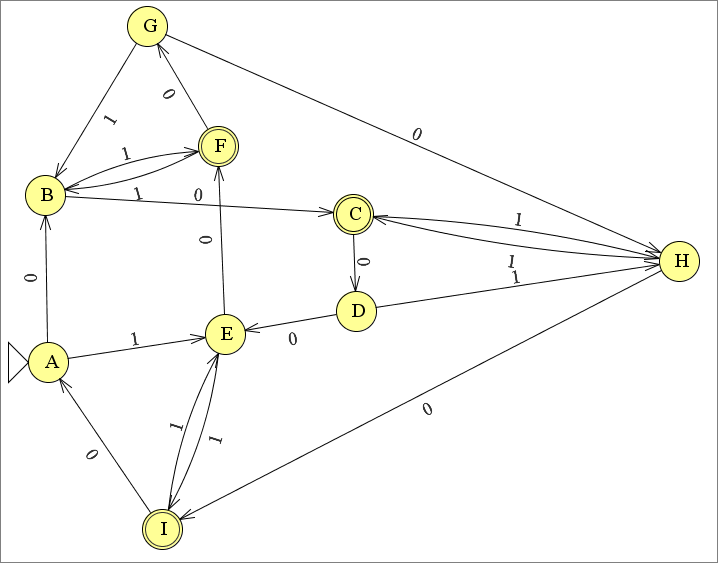
Los estados de que tienen algún estado final de Q:

# Problema 3

Considere el descrito por , donde , y la función 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 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** |

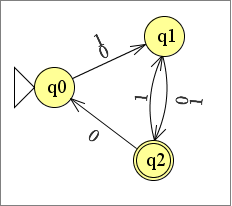
Al agregar los estados que producen estos estados en negrita, se puede obtener la siguiente tabla de equivalencia:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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:

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

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



# Problema 4

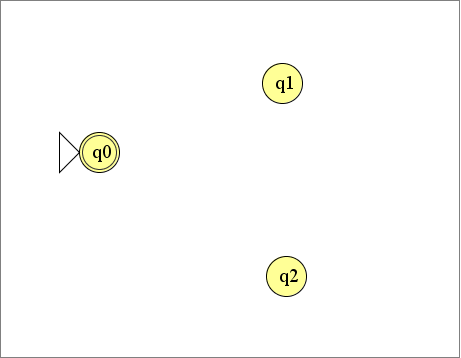
Demuestre que, para todo , el lenguaje es regular. Para ello, muestre un AFD para y demuestre que efectivamente su AFD reconoce el lenguaje. b) demuestre la correctitud.

Para poder crear un AFD para , se partirá por explicar un caso de ejemplo. Suponer un . 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 |

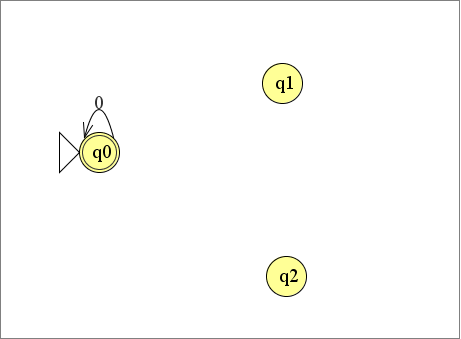
El residuo de puede tener 3 valores Se puede construir un AFD que acepte creando 3 estados para cada uno de los residuos y generando las transiciones para todos los estados hasta completar todas las transiciones sobre el alfabeto , es decir, 2 transiciones por cada estado, donde cada estado indica el residuo de . Nótese que por lo que el estado debe ser estado final.

Así, inicialmente se tiene un AFD dado por:

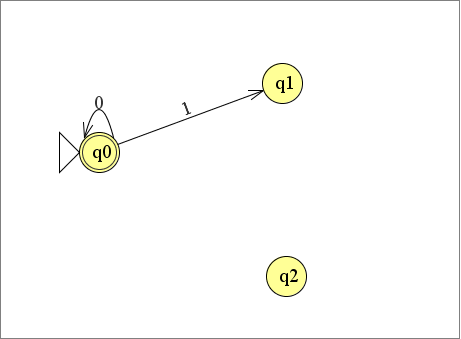


Al avanzar por los números de la tabla, se puede ir trazando las transiciones, considerando que se debe llegar al estado con i el residuo.

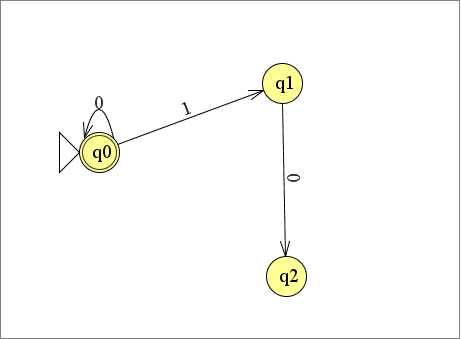
De esta forma para x=0 (binario 0), el residuo r=0, se debe ir al estado .con el binario dado. Así :



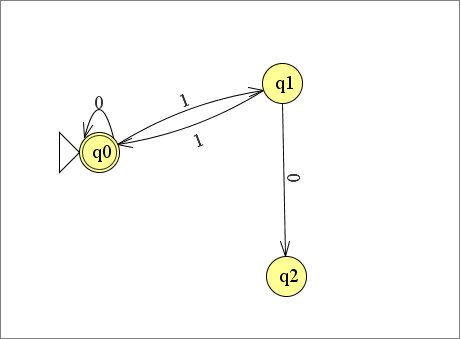
Para x=1 (binario 1), r=1, por lo que se puede trazar una transición :



Para x=2 (10), r=2, por lo que se debe ir al estado .

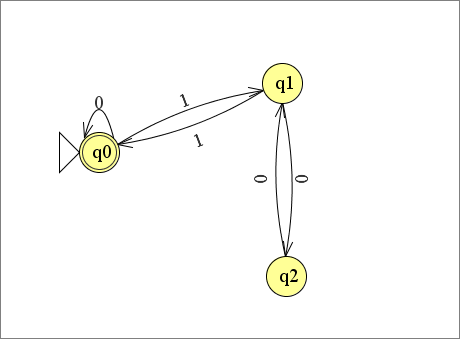


Para x=3 (11), r=0:

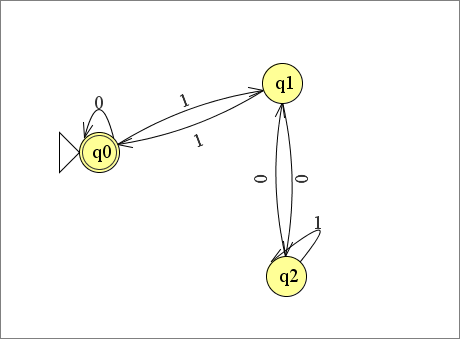


El procedimiento se puede completar hasta que se complete el AFD, es decir, sean 2m transiciones. Esto ocurrirá en el número 2m-1.

Para el caso x=4; r=1;



Llegado a este punto, 2m-1=5; r=2,



Se tiene completo el AFD que reconoce .

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

; ;

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

Dónde tiene estados: donde cada índice de indica el residuo de .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 , se debe llegar al estado donde es el residuo de .

Como , , luego para cualquier , es el estado inicial.

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

Para demostrar :

El AFD tiene transiciones por las que se procesa la palabra. Cabe notar que y que para otro x, el autómata no acepta, ya que las demás transiciones no son estados finales. Se puede argumentar que el AFD tiene la característica de volver a su estado inicial luego de procesar la palabra sobre los estados. Es decir, para una cadena cualquiera que representa un número , 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

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 que contempla estas transiciones.

# Problema 5

Si es un lenguaje y un símbolo, definimos la operación cociente de y como Por ejemplo, si , entonces . Demuestre que si es regular entonces es regular. Debe demostrar que su construcción es correcta.

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

Tal que el lenguaje que reconoce este AFD es

Se puede demostrar, que para , debe existir una transición sobre , desde un estado que lleve a :

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

En este, son todos los estados de , sin .

De la misma manera, corresponden a todas las transiciones de M, sin la transición final de

El estado inicial .

F’ es el estado .

Es evidente que si acepta a , entonces existen las transiciones en de tal forma que

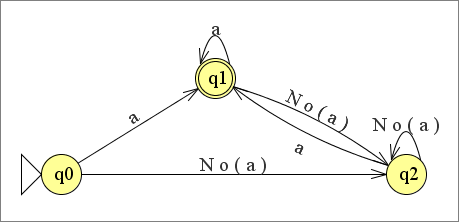
De lo anterior, y al existir este autómata que reconoce a , se podría concluir que , y que por lo tanto es un lenguaje regular.

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 es la concatenación de dos lenguajes regulares y .

Con

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

De la forma:



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

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

Esto sin embargo, podría no ser suficiente, ya que se podía pensar en que 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 y la transición , un estado final, se puede concluir que efectivamente, existe una serie de transiciones tal que se pueda llegar hasta . Con esto, se sabe que el camino existe, y por lo tanto, se puede inferir que es regular.