Universidad Nacional de Colombia

Introducción a la teoría de la computación

Parcial 1

Oscar Julian Rodriguez - 1022445731- osrodriguezc@unal.edu.co
Juan Pablo Urrutia Parrado - 1193382462 - jurrutia@unal.edu.co

August 30, 2022

1

Dado el lenguaje regular $L = \{1^k : k \equiv 1 \mod 4 \quad k \equiv 2 \mod 4\}$ construir un autómata M que sea capaz de reconocerlo.Es decir $L(M) = L$.

- **Expresión regular**

  El autómata M dados inputs tales como 1,11,11111,111111,111111111,1111111111 debe aceptarlos, y dados inputs como 111,1111 debe rechazarlos. Visto de otra forma, debe ser capaz de reconocer cadenas de $1's$ cuya longitud sea un múltiplo de 4 sumado 1 o un múltiplo de 4 sumado 2; además de reconocer las cadenas 1 y 11. Una aproximación para construir la expresión regular es plantear dos casos posibles:

  - El primer caso considera $1 \cdot (1111)^*$ que representaría las cadenas de 1's cuya longitud sea un múltiplo de 4 sumado 1 donde también está incluido el caso de reconocer la cadena 1 pues al tratarse con la operación de la estrella de Kleene en (1111^*) se incluye la cadena vacía por lo tanto $1 = 1 \cdot \epsilon_0$.

  - De manera similar, el segundo caso considera $11 \cdot (1111)^*$ que representaría las cadenas de 1's cuya longitud sea un múltiplo de 4 sumado 2 donde también está incluido el caso de reconocer la cadena 11 pues al tratarse con la operación de la estrella de Kleene en (1111^*) se incluye la cadena vacía por lo tanto $11 = 11 \cdot \epsilon_0$.

  Finalmente, basta con unir ambos casos mediante la operación de unión, por consiguiente la expresión regular queda definida así:

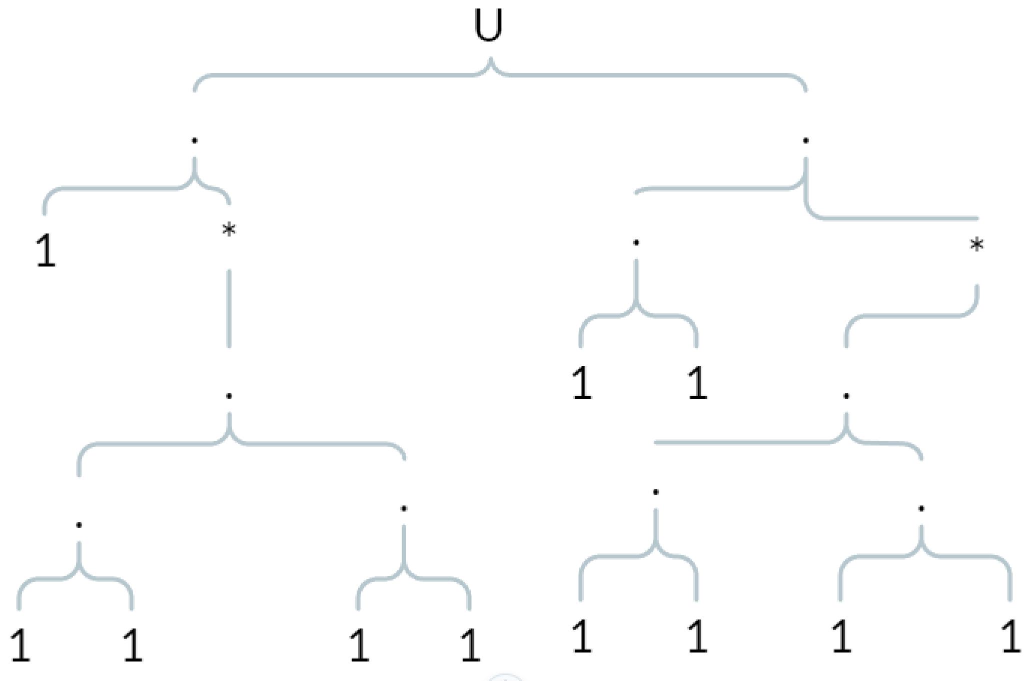
  $$(1 \cdot (1111)^*) \cup (11 \cdot (1111)^*)$$

- **Parsing**

  Ya teniendo la expresión regular del lenguaje regular se procede a construir el árbol sintáctico, pero antes se reescribe la expresión regular para no dar lugar a ambigüedad alguna, y que sean más visibles las operaciones:

  $$((1) \cdot (1 \cdot (1 \cdot ((1) \cdot (1))))^*) \cup ((11) \cdot (1 \cdot (1 \cdot ((1) \cdot (1))))^*)$$

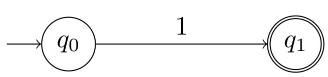
Luego el árbol sintáctico queda representado de la siguiente manera:



- **Autómata asociado**

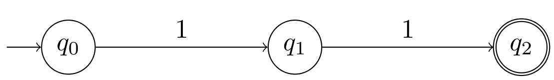
  Para realizar un autómata a partir del anterior árbol, es necesario hacer automatas que reconozcan desde las hojas hacia arriba, y cada vez que se sube un nivel, se usan los anteriores automatas para construír. En la práctica se obtienen los siguientes resultados:

  1. **Autómata que reconoce** 1

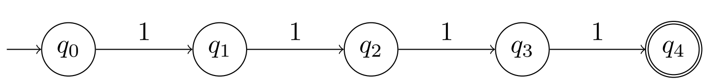
     Con $\Sigma = \{1\}$, $Q = \{q_0, q_1\}$, q_0 es el estado inicial, $F = \{q_1\}$ y $\delta : Q \text{ x } \Sigma \rightarrow Q$ tal que $\delta(q_0, 1) = q_1$ y $\delta(q_1, 1)$ conduce a un estado limbo.

  2. **Autómata que reconoce** 11

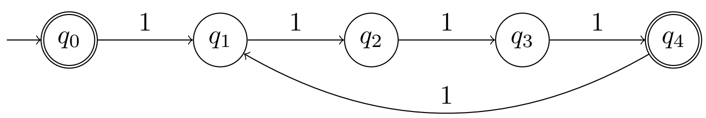
     Con $\Sigma = \{1\}$, $Q = \{q_0, q_1, q_2\}$, q_0 es el estado inicial, $F = \{q_2\}$ y $\delta : Q \text{ x } \Sigma \rightarrow Q$ tal que $\delta(q_0, 1) = q_1$, $\delta(q_1, 1) = q_2$ y $\delta(q_2, 1)$ conduce a un estado limbo.

  3. **Autómata que reconoce** 1111

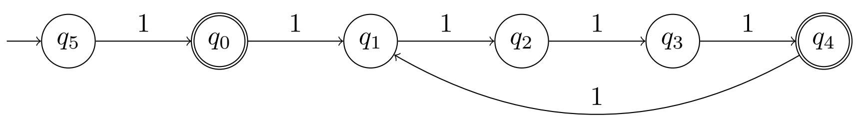
Con $\Sigma = \{1\}$, $Q = \{q_0, q_1, q_2, q_3, q_4\}$, q_0 es el estado inicial, $F = \{q_4\}$ y $\delta : Q$ x $\Sigma \to Q$ tal que $\delta(q_0, 1) = q_1$, $\delta(q_1, 1) = q_2$, $\delta(q_2, 1) = q_3$, $\delta(q_3, 1) = q_4$, y $\delta(q_4, 1)$ conduce a un estado limbo.

4. **Autómata que reconoce** $(1111)^*$



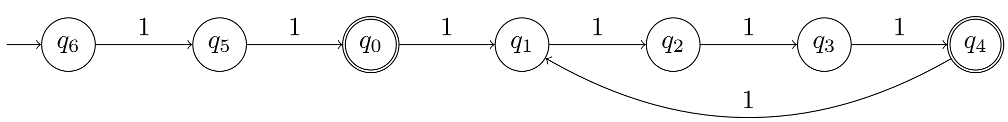
Con $\Sigma = \{1\}$, $Q = \{q_0, q_1, q_2, q_3, q_4\}$, q_0 es el estado inicial, $F = \{q_0, q_4\}$ y $\delta : Q$ x $\Sigma \to Q$ tal que $\delta(q_0, 1) = q_1$, $\delta(q_1, 1) = q_2$, $\delta(q_2, 1) = q_3$, $\delta(q_3, 1) = q_4$, y $\delta(q_4, 1) = q_1$.

5. **Autómata que reconoce** $1 \cdot (1111)^* = M_1$



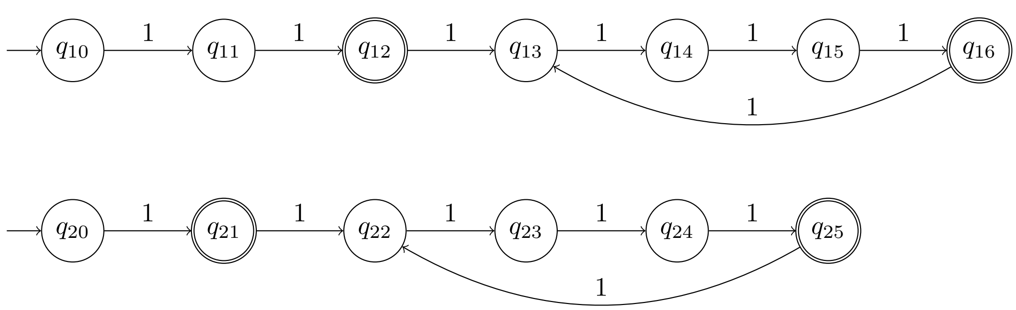
Con $\Sigma = \{1\}$, $Q_1 = \{q_0, q_1, q_2, q_3, q_4, q_5\}$, q_5 es el estado inicial, $F_1 = \{q_0, q_4\}$ y $\delta_1 : Q_1$ x $\Sigma \to Q_1$ tal que $\delta_1(q_5, 1) = q_0$, $\delta_1(q_0, 1) = q_1$, $\delta_1(q_1, 1) = q_2$, $\delta_1(q_2, 1) = q_3$, $\delta_1(q_3, 1) = q_4$, y $\delta_1(q_4, 1) = q_1$.

6. **Autómata que reconoce** $11 \cdot (1111)^* = M_2$



Con $\Sigma = \{1\}$, $Q_2 = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$, q_6 es el estado inicial, $F_2 = \{q_0, q_4\}$ y $\delta_2 : Q_2$ x $\Sigma \to Q_2$ tal que $\delta_2(q_5, 1) = q_0$, $\delta_2(q_0, 1) = q_1$, $\delta_2(q_1, 1) = q_2$, $\delta_2(q_2, 1) = q_3$, $\delta_2(q_3, 1) = q_4$, y $\delta_2(q_4, 1) = q_1$.

7. **Autómata que reconoce** $(1 \cdot (1111)^*) \cup (11 \cdot (1111)^*) = M_1 \cup M_2 = M_3$

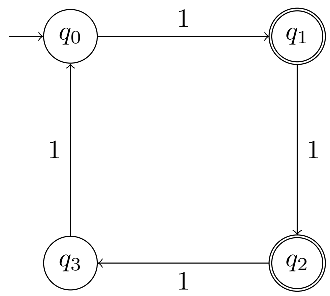


Con $\Sigma = \{1\}$, $Q_3 = Q1 \times Q2$, (q_{10}, q_{20}) es la tupla de estados iniciales, $F_3 = F_1 \times F_2$ y $\delta_3 : Q_3$ x $\Sigma \to Q_3$ tal que $\delta_3((q_n, q_m), 1) = (\delta_1(q_n, 1), \delta_2(q_m, 1))$. Es importante notar que para este último autómata, es como si se usaran dos sub-autómatas para verificar cada lado de la unión, que se usan de manera simultanea. En efecto, estos son M_1 y M_2.

Luego se obtiene el autómata no optimizado M_3 como resultado final.

- **Autómata optimizado**

  A continuación se construye el autómata que reconozce el lenguaje regular L:



  De esta manera, abstrayendo la información del grafo se define el autómata $M = (\Sigma, Q, q_0, F, \delta)$, donde:

  1. $\Sigma = \{1\}$
  2. $Q = \{q_0, q_1, q_2, q_3\}$
  3. $q_0 \to$ **estado inicial**
  4. $F = \{q_1, q_2\}$
  5. $\delta : Q$ **x** $\Sigma \to Q$ como:
     (a) $\delta(q_0, 1) = q_1$
     (b) $\delta(q_1, 1) = q_2$
     (c) $\delta(q_2, 1) = q_3$
     (d) $\delta(q_3, 1) = q_0$

- **Compilación** Para definir la palabra o compilación del autómata M definido anteriormente utilizaremos la siguiente tabla donde simbolizaremos los estados q_0 como 1, q_1 como 11, q_2 como 111 y q_3 como 1111, entonces:

Estados	**Input**	**Output**
1	1	11
11	1	111
111	1	1111
1111	1	1

  Se definirá entonces $W_M = Q0F00\Sigma0\delta$, por lo tanto reemplazando cada objeto en su lugar, donde $Q = 1111$ pues se tienen 4 estados para el autómata, $F = 110111$ pues los estados de aceptación son $q_1 = 11$ y $q_2 = 111$, $\Sigma = 1$ pues el alfabeto es 1 y para la función delta seguiremos la tabla, por ejemplo, para codificar $\delta(q_0, 1) = q_1$ se sabe que $q_0 = 1$ y $q_1 = 11$ entonces quedaría codificado como 101011, asi entonces:

$$W_M = 111101101110010101011001101011100111010111100111 10101$$

- **Prueba en inputs**

  Ahora se ejecutará el autómata utilizando los inputs 11111 que debería ser aceptado pues se tiene que $k = 5$ por consiguiente $5 \equiv 1 \mod 4$ ya que $5 - 1 = 4 \cdot 1$ y el input 111 que debería ser rechazado pues se tiene que $k = 3$ y no se cumple que $3 \equiv 1 \mod 4$ y tampoco $3 \equiv 2 \mod 4$, veamoslo :

  1. Para el input 11111:
     - $\delta(q_0, 1) = q_1$
     - $\delta(q_1, 1) = q_2$
     - $\delta(q_2, 1) = q_3$
     - $\delta(q_3, 1) = q_0$
     - $\delta(q_0, 1) = q_1$
       De esta manera vemos que el estado final fué q_1 luego el input 11111 fué aceptado pues $q_1 \in F$.
  2. Para el input 111:
     - $\delta(q_0, 1) = q_1$
     - $\delta(q_1, 1) = q_2$
     - $\delta(q_2, 1) = q_3$

     De esta manera vemos que el estado final fué q_3 luego el input 11111 fué rechazado pues $q_3 \notin F$.