LÒGICA I LLENGUATGES

CURSO 2022-23

TERCERA PRUEBA PARCIAL (Grupo B)

(a) Construir un autómata determinista M tal que $L(M) = \{x \in \{0,1\}^* : x$ comienza en 0 y acaba en 1 $\}$.

(2 puntos)

(b) Consideremos el autómata indeterminista $M = (\{A, B, C, D, E\}, \{0, 1\}, \Delta, A, \{E\})$ donde Δ está definida por la siguiente tabla:

A	0	A
A	1	A
A	1	B
B	0	C
B	λ	C
C	1	D
D	1	E
D	λ	E
E	0	E
E	1	E

Se pide entonces:

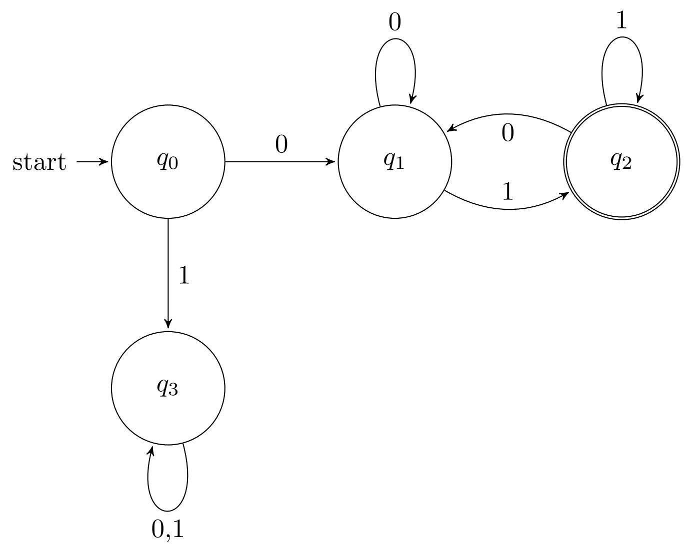
(1) Describir el lenguaje $L(M)$ informalmente. (1 punto)

(2) Describir el lenguaje $L(M)$ mediante una expresión regular. (1 punto)

(3) Siguiendo el método visto en clase, transformar el autómata M en un autómata determinista equivalente. (4 puntos)

(3) Programar en JAVA o en C el autómata determinista obtenido en (3). (2 puntos)

Solución : (a)



(b) (1) $L(M) = \{x \in \{0,1\}^* : x$ contiene 11 o 101 como subpalabra $\}$.

(2) $L(M) = L(\alpha)$ donde $\alpha = (0 \cup 1)^*(11 \cup 101)(0 \cup 1)^*$.

(3) Construimos el autómata determinista M' equivalente a M. Tenemos $\Lambda(A) = A$, $\Lambda(B) = BC$, $\Lambda(C) = C$, $\Lambda(D) = DE$, $\Lambda(E) = E$.

El estado inicial del auómata determinista M es $\Lambda(A) = A$. Tenemos entonces:

$\delta'(A,0) = \Lambda(A) = A$, $\delta'(A,1) = \Lambda(A) \cup \Lambda(B) = ABC$,
$\delta'(ABC,0) = \Lambda(A) \cup \Lambda(C) = AC$, $\delta'(ABC,1) = \Lambda(A) \cup \Lambda(B) \cup \Lambda(D) = ABCDE$,
$\delta'(AC,0) = \Lambda(A) = A$, $\delta'(AC,1) = \Lambda(A) \cup \Lambda(B) \cup \Lambda(D) = ABCDE$,
$\delta'(ABCDE,0) = \Lambda(A) \cup \Lambda(C) \cup \Lambda(E) = ACE$, $\delta'(ABCDE,1) = \Lambda(A) \cup \Lambda(B) \cup \Lambda(D) \cup \Lambda(E) = ABCDE$,
$\delta'(ACE,0) = \Lambda(A) \cup \Lambda(E) = AE$, $\delta'(ACE,1) = \Lambda(A) \cup \Lambda(B) \cup \Lambda(D) \cup \Lambda(E) = ABCDE$,
$\delta'(AE,0) = \Lambda(A) \cup \Lambda(E) = AE$, $\delta'(AE,1) = \Lambda(A) \cup \Lambda(B) \cup \Lambda(E) = ABCE$,
$\delta'(ABCE,0) = \Lambda(A) \cup \Lambda(C) \cup \Lambda(E) = ACE$, $\delta'(ABCE,1) = \Lambda(A) \cup \Lambda(B) \cup \Lambda(D) \cup \Lambda(E) = ABCDE$.

Por tanto, los estados que hemos obtenido son los siguientes: A, ABC, AC, $ABCE$, $ABCDE$, AE y ACE. Como E es el estado aceptador de M, los estados aceptadores de M' son $ABCE$, $ABCDE$, AE y ACE. Se observa que estos cuatro estados aceptadores son equivalentes, ya que la función δ' aplicada a cualquiera de ellos nunca nos lleva a un estado no aceptador. Por tanto, podemos juntar los cuatro estados aceptadores en un único estado.

(4) Representamos al estado A por 0, al estado ABC por 1, al estado AC por

2 y al estado aceptador $ABCDE$ por 3. Como hemos indicado anteriormente, podemos eliminar los otros estados aceptadores, por ser equivalentes al estado $ABCDE$. Podemos escribir entonces el siguiente método en Java para simular el autómata M':

```
public boolean simular (String entrada)
{ int q = 0, i = 0;
  char c = entrada.charAt(0);
  while (c != '$')
  { switch(q)
        { case 0:
        if (c == '1') q = 1;
        break;
        case 1:
        if (c == '0') q = 2; else if (c == '1') return true;
        break;
        case 2:
        if (c == '0') q = 0; else if (c == '1') return true;
        break;}
  c = entrada.charAt(++i); } return false; }
```