RELACIÓN 1 - ALBERTO LLAMAS GONZÁLEZ

① Describir el lenguaje generado por las siguientes gramáticas:

- $S \to aS_1b$

  $S_1 \to aS_1 \mid bS_1 \mid \varepsilon$

$S \Rightarrow aS_1b \Rightarrow abS_1b$

$\Rightarrow aaS_1b$

$L = \{a^i u b : u \in \{a,b\}^*; i > 0\}$

- $S \to aSb \mid S_1$

  $S_1 \to \varepsilon$

$S \Rightarrow aSb \Rightarrow a^2Sb^2 \ldots \Rightarrow a^nS_1b^n \Rightarrow a^nb^n$

$\Rightarrow S_1 \Rightarrow \varepsilon$

$L = \{a^ib^i : i \geq 0\}$

- $S \to aSb \mid S_1$

  $S_1 \to aS_1 \mid bS_1 \mid \varepsilon$

$S \Rightarrow aSb \Rightarrow \ldots \Rightarrow a^iS_1b^i$

$\Rightarrow S_1 \Rightarrow aS_1 \to au \quad u \in \{a,b\}^*$

$\Rightarrow bS_1 \Rightarrow bu \quad u \in \{a,b\}^*$

$L = \{a^i u b^i ; i \geq 0,\ u \in \{a,b\}^*\}$

- $S \to aSb \mid S_1$

  $S_1 \to cS_1d \mid \varepsilon$

$S \Rightarrow aSb$

$\Rightarrow S_1 \Rightarrow cS_1d$

$L = \{a^ic^jd^jb^i ; i,j \geq 0\}$

② Encontrar gramáticas regulares o gramáticas libres de contexto que generen los siguientes lenguajes en el alfabeto $A = \{a, b, c\}$

1. $L = \{a^i b^j c^{i+j} \mid i, j \in \mathbb{N}\}$

$S \to aSc \mid aS_1c \mid \varepsilon$

$S_1 \to bS_1c \mid \varepsilon$

2. $u \in L \iff$ verifica que u empieza por el símbolo 'a' y acaba con el símbolo 'c'

$S \to aSc \mid aS_1c$

$S_1 \to aS_1 \mid bS_1 \mid cS_1 \mid \varepsilon$

3. $u \in L \iff$ verifica que u contiene un número par de símbolos a

$S \to S_1 a S_1 a S_1 \mid SaSaS$

$S_1 \to bS_1 \mid cS_1 \mid \varepsilon$

Escaneado con CamScanner

(3) Determinar si el lenguaje sobre el alfabeto $A = \{a, b\}$ generado por la siguiente gramática es regular (justifica la respuesta)

$$S \rightarrow S_1 b S_2$$
$$S_1 \rightarrow a S_1 \mid \varepsilon$$
$$S_2 \rightarrow a S_2 \mid b S_2 \mid \varepsilon$$

Al tratarse de una gramática regular al producir reglas con la forma $A \rightarrow uB$ (Lineal por la derecha) podemos afirmar que nos va a generar un lenguaje regular.

Veamos el lenguaje que genera

$$S \Rightarrow S_1 b S_2 \Rightarrow a S_1 b S_2 \Rightarrow a S_1 b a S_2$$
$$\Rightarrow a S_2 b b S_2$$

$$L = \{a^i b u \; ; \; u \in \{a, b\}^*, \; i \geq 0\}$$

Escaneado con CamScanner

④ AFD $M = (Q, A, \delta, q_0, F)$

$Q = \{q_0, q_1, q_2\}$

$A = \{0, 1\}$

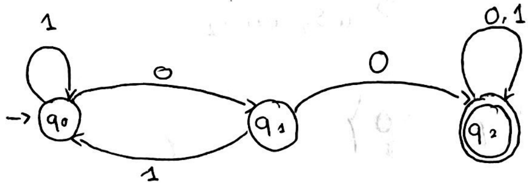
$\delta(q_0, 0) = q_1$ $\qquad$ $\delta(q_0, 1) = q_0$

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

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

$F = \{q_2\}$

Dibujar el autómata y describir informalmente el lenguaje aceptado. Ejemplo



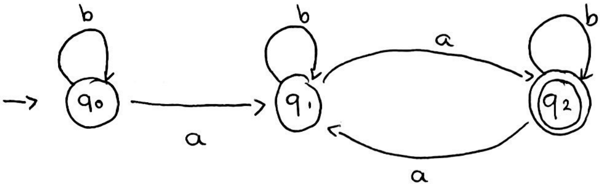
El autómata acepta cualquier palabra que tenga dos ceros seguidos

101001 : $(101001, q_0) \vdash (01001, q_0) \vdash (1001, q_1) \vdash$

$\vdash (001, q_0) \vdash (01, q_1) \vdash (1, q_2) \vdash^{*} (\varepsilon, q_2)$

101001 es aceptada

5) Dado el siguiente autómata M, descríbelo usando el formalismo $M = (Q, A, \delta, q_0, F)$. Indica cuál es el lenguaje aceptado por dicho autómata. Mostrar algún ejemplo para aceptar y rechazar cadenas



$Q = \{q_0, q_1, q_2\}$

$A = \{a, b\}$

Función de transición

$\delta(q_0, a) = q_1$ $\delta(q_1, b) = q_1$

$\delta(q_1, a) = q_2$ $\delta(q_2, b) = q_2$

$\delta(q_2, a) = q_1$ $\delta(q_0, b) = q_0$

$F = \{q_2\}$

$L = \{ b^i a b^j a^{2n+1} b^k ; i, j, k \geq 0, n \in \mathbb{N} \}$

$(baab, q_0) \vdash (aab, q_0) \vdash (ab, q_1) \vdash (b, q_2) \vdash^* (\varepsilon, q_2)$

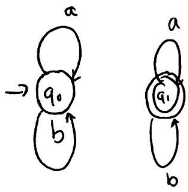
Aceptada

$(baaba, q_0) \vdash (aaba, q_0) \vdash (aba, q_1) \vdash (ba, q_2) \vdash (a, q_2) \vdash^*$

$\vdash^* (\varepsilon, q_1)$ NO ES Aceptada

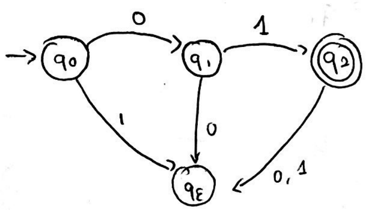
6. Dibujar los AFDs que aceptan los siguientes lenguajes con alfabeto $\{0,1\}$:

a) El lenguaje vacío, $\emptyset$

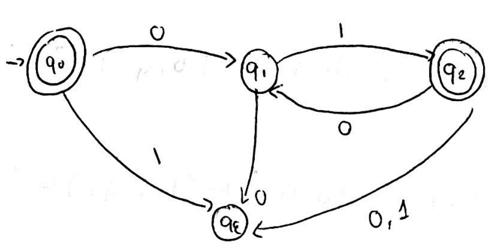


b) El lenguaje formado por la palabra vacía $\{\varepsilon\}$

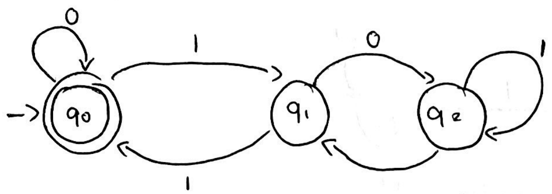
c) El lenguaje formado por la palabra 01



d) Lenguaje formado por sucesiones de 01 incluyendo el vacío

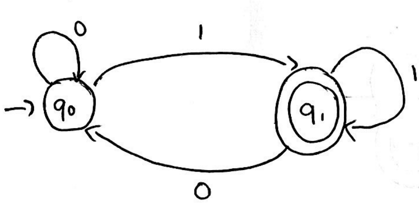


⑥ e) Lenguaje formado por las cadenas donde el nº de unos es divisible por 3

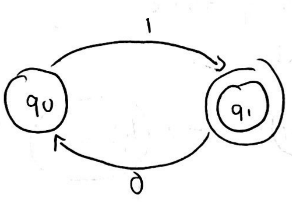


⑦ Construir un AFD capaz de aceptar una cadena $u \in \{0,1\}^*$ que contenga la subcadena 10101. Ahora (de manera independiente) construye un AFND que acepte el mismo lenguaje. ¿Cuál de los dos diseños es más sencillo?

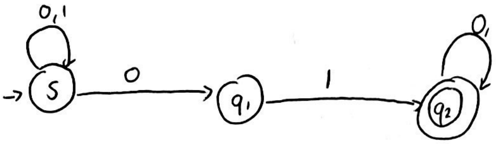
AFD



AFND (es más sencillo)



⑧ ¿Qué lenguaje acepta este AFND? Mostrar algún ejemplo de uso para aceptar y rechazar cadenas. Obtener un AFD equivalente



Lenguaje que contiene ~~una~~ la subcadena 01

$(1010, S) \vdash (010, S) \vdash (10, q_1) \vdash (0, q_2) \vdash^* (\varepsilon, q_2)$ Acepta

$(100, S) \vdash (00, S) \vdash (0, q_1)$

No acepta

$\vdash (0, S) \vdash (\varepsilon, S)$

AFD equivalente