



SOLUCION

Apellidos:

Nombre:

Ejercicio 1:

Dada la Gramática $G = \{ \{ 0, 1 \}, \{ S, B \}, S, \mathcal{P} \}$ con las producciones \mathcal{P} :

$S ::= 0SBB \mid 0BB$

$B ::= 1$

- Construir un autómata a Pila (AP) que reconozca por vaciado de pila el lenguaje que genera la gramática G . Se debe utilizar un método conocido que permita obtener un AP a partir de una gramática. Explicar los pasos del método elegido. (7 puntos)
- Comprobar el funcionamiento del AP para las palabras 0011 y 001111, así como, la generación en G de dichas palabras. (2 puntos)
- Describe el lenguaje que genera G y que acepta el AP. (1 punto)

25 minutos

Para construir un AP que acepte o reconozca el lenguaje generado por una gramática G existen 2 métodos. Cualquiera de las 2 soluciones que se muestran a continuación se dará por válida.

METODO 1

a) Construir un AP1 que acepte (reconozca) el lenguaje generado por la gramática: $G = \{ \Sigma_T, \Sigma_N, P, S \}$. La gramática ha de estar en FNG.

$AP1 = \{ \Sigma_T, \Sigma_N, \{ q \}, S, q, f, \emptyset \}$

Σ_T = alfabeto de entrada (Σ)

Σ_N = alfabeto de pila (Γ)

$\{ q \} = Q$ conjunto de estados de AP

S = Símbolo de inicio de pila

q = estado inicial de AP

F = Función de transición (movimientos)

$F = \emptyset$

ALGORITMO (para obtener los movimientos del AP):

| | |
|--|---|
| <p>Si $A ::= aZ$, $a \in \Sigma_T, A \in \Sigma_N, Z \in \Sigma_N^*$ entonces, $(q Z) \in f(q a A)$</p> <p>Si $S ::= \lambda$ entonces, $(q \lambda) \in f(q \lambda S)$</p> | <p>La gramática YA ESTÁ en FNG</p> <p>$AP1 = \{ \Sigma_T, \Sigma_N, \{ q \}, S, q, f, \emptyset \}$</p> <p>$AP1 = \{ \{ 0, 1 \}, \{ S, B \}, \{ q \}, S, q, f, \emptyset \}$</p> <p>Movimientos del AP son f:</p> <p>$f(q 0 S) = (q SBB) (q BB)$</p> <p>$f(q 1 B) = (q \lambda)$</p> |
|--|---|

b) Aceptación por parte del AP1 de las palabras (0011 y 001111).

0011: $[q, 0011, S] \vdash [q, 011, SBB] \vdash [q, 11, BBBB] \vdash [q, 1, BBB] \vdash [q, \lambda, BB]$ (NO ACEPTA)

001111: $[q, 001111, S] \vdash [q, 01111, SBB] \vdash [q, 1111, BBBB] \vdash [q, 111, BBB] \vdash [q, 11, BB] \vdash [q, 1, B] \vdash [q, \lambda, \lambda]$ (SI ACEPTA)

Generación de estas palabras por parte de la gramática.

0011: $S \rightarrow 0SBB$ (NO GENERA) y $S \rightarrow 0BB$ (NO GENERA)

001111: $S \rightarrow 0SBB \rightarrow 00BBBB \rightarrow 001BBB \rightarrow 0011BB \rightarrow 00111B \rightarrow 001111$ (SI GENERA)

c) El lenguaje que genera G y que acepta el AP1 es: $L = \{ 0^n 1^m / m = 2n, n \geq 1 \}$

METODO 2

Construir un AP2 que acepte (reconozca) el lenguaje generado por la gramática: $G = \{ \Sigma_T, \Sigma_N, \mathbb{P}, S \}$. La gramática puede estar o no en FNG.

$AP2 = \{ \Sigma_T, \{ \Sigma_N \cup \Sigma_T \}, \{ q \}, S, q, f, \emptyset \}$

Σ_T = Alfabeto de entrada (Σ)

$\{ \Sigma_N \cup \Sigma_T \}$ = Alfabeto de pila (Γ)

$\{ q \} = Q$ (Conjunto de estados del AP)

S = Símbolo de inicio de pila

q = estado inicial del AP

F = Función de transición (movimientos)

$F = \emptyset$ (Conjunto de estados finales)

ALGORITMO (para obtener los movimientos del AP):

| | |
|--|---|
| $X \in \{ \Sigma_N \cup \Sigma_T \}, A \in \Sigma_N$ $\forall A :: X$ producción de la gramática, en AP se hace: $(q X) \in f(q \lambda A)$ $\forall a \in \Sigma_T$, entonces, $(q \lambda) \in f(q a a)$ | $AP2 = \{ \Sigma_T, \Sigma_N, \{ q \}, S, q, f, \emptyset \}$ $AP2 = \{ \{ 0, 1 \}, \{ 0, 1, S, B \}, \{ q \}, S, q, f, \emptyset \}$ Movimientos del AP f : $f(q \lambda S) = (q 0SBB) (q 0BB)$ $f(q \lambda B) = (q 1)$ $f(q 0 0) = (q \lambda)$ $f(q 1 1) = (q \lambda)$ |
|--|---|

Aceptación por parte del AP2 de las palabras (0011 y 001111).

0011: $[q, 0011, S] \vdash [q, 0011, 0SBB] \vdash [q, 011, SBB] \vdash [q, 011, 0BBBB] \vdash [q, 11, BBBB] \vdash [q, 11, 1BBB] \vdash [q, 1, BBB] \vdash [q, 1, 1BB] \vdash [q, \lambda, BB]$ (NO ACEPTA)

001111: $[q, 001111, S] \vdash [q_0, 001111, 0SBB] \vdash [q_1, 01111, SBB] \vdash [q_1, 01111, 0BBBB] \vdash [q_1, 1111, BBBB] \vdash [q_1, 1111, 1BBB] \vdash [q_1, 111, BBB] \vdash [q_1, 111, 1BB] \vdash [q_1, 11, BB] \vdash [q_1, 11, 1B] \vdash [q_1, 1, B] \vdash [q_1, 1, 1] \vdash [q_1, \lambda, \lambda]$ (SI ACEPTA)

Generación de estas palabras por parte de la gramática.

0011: $S \rightarrow 0SBB$ (NO GENERA) y $S \rightarrow 0BB$ (NO GENERA)

001111: $S \rightarrow 0SBB \rightarrow 00BBBB \rightarrow 001BBB \rightarrow 0011BB \rightarrow 00111B \rightarrow 001111$ SI GENERA

c) El lenguaje que genera G y que acepta el AP2 es: $L = \{ 0^n 1^m / m = 2n, n \geq 1 \}$



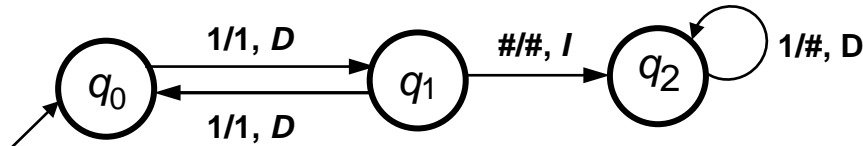
SOLUCION

Apellidos:

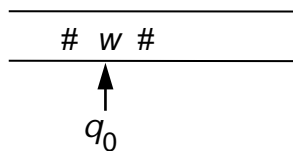
Nombre:

Ejercicio 2:

Sea la Máquina de Turing M definida según el siguiente grafo:

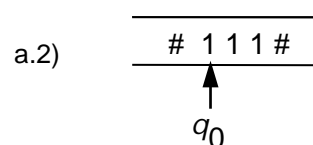
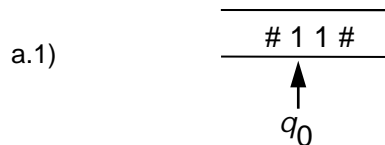


Y cuya configuración inicial es la siguiente:



Donde $w \in 1^*$ es un número entero codificado en unario. M inicialmente está en el estado q_0 leyendo el primer 1 de w.

- a) ¿Qué función aritmética sobre cada w calcula M? ¿Cuál es la configuración final de M tras recibir las entradas de los apartados a.1) y a.2)? (2 puntos)



- b) Escribir (y describir brevemente) el contenido inicial de la cinta de una Máquina de Turing Universal (MTU) programada para simular a la máquina M con la entrada del apartado a.2). Utilicen la siguiente codificación binaria: $q_0 \equiv 00$; $q_1 \equiv 01$; $q_2 \equiv 10$; Izqda I $\equiv 1$; Dcha D $\equiv 0$ (2 puntos)
- c) Escribir (y describir brevemente) el contenido de la cinta de la MTU después de simular el primer movimiento que realiza la máquina M con la entrada del apartado a.2). ¿A qué estado accede el módulo simulador tras recolocar el *? ¿Por qué? (3 puntos)
- d) Escribir (y describir brevemente) el contenido final de la cinta de la MTU cuando termine de simular a la máquina M con la entrada del apartado a.2). ¿En qué estado se para la MTU? ¿Por qué? (3 puntos)

NOTA: Todos los apartados se responderán en la carilla de atrás.

Durante el examen se da fotocopia con el grafo de los tres módulos de la MTU.

25 minutos

Continuación ejercicio 2. RESPUESTAS. SOLUCIONES

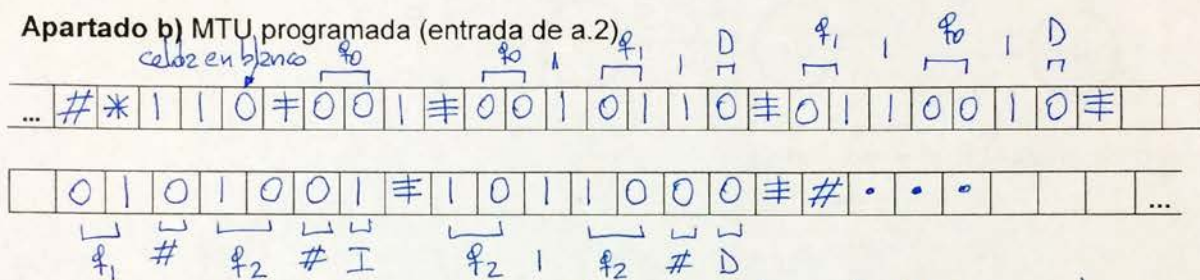
Apartado a) a.1) y a.2)

$$a.1) q_0 \vdash 1 q_1 \vdash 1 q_0 \# P_{22}$$

$$a.2) q_0 \vdash 1 q_1 \vdash 1 q_0 \vdash 1 q_1 \# \vdash 1 q_2 \vdash 1 \# q_2 \# P_{22}$$

Función: $f(w) = w - w \bmod 2$ (borra el último 1 de bs w impares)
le restz 1 a bs impares

Apartado b) MTU programada (entrada de a.2)



El * se sitúa sobre la primera celda que está leyendo M (que contiene un 1)
Registro inicial: $\neq 001 \neq$ Estado inicial de M (q_0) y símbolo que lee M (1).

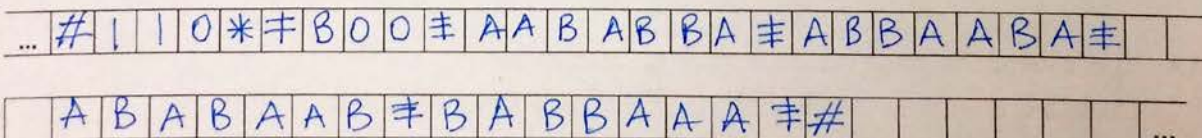
4 registros: uno por cada estado diferente que puede ejecutar M.

Apartado c) Mód. Simulador (escribid sólo la parte de la cinta que cambia)



La máquina M ejecuta el movimiento $f(q_0, 1) = (q_1, 1, D)$. Por tanto, la MTU desplaza el * una celda a la derecha. En la celda donde estaba el * vuelve a escribir un 1.
El control pasa de q_0 a q_1 , con lo que el contenido del registro inicial se actualiza.
Al recordar el *, el mód. simulador se encuentra con un 1, que se memoriza transitando de q_{20} a q_{22}

Apartado d) MTU para



- La MTU se para en el estado q_5 porque el módulo localizador (tras haber marcado y rechazado previamente todos los registros) está memorizando y buscando un "1" en el siguiente registro (que no existe y aparece el #)
- El * está sobre la celda que M lee cuando se para.