

Tarea 4

Informática Teórica

Matías Peñaloza 202373037-8

2024-2

Concepto	Tiempo [min]
Revisión	60
Desarrollo	90
Informe	90

1 Enunciado

Directamente construya autómatas de pila que acepten los siguientes lenguajes, ya sea por estado final o por pila vacía. Explique sus construcciones.

1. $L_1 = \{a^i b^j c^k : i + j = k\}$
(50 puntos)
2. $L_2 = \{\omega \in \{a, b\}^* : \#_a(\omega) \neq \#_b(\omega)\}$, donde $\#_a(\omega)$ es el número de a en ω .
Pista: Contabilice la diferencia del número de a y b vistas, por ejemplo con A en la pila las a sobrantes y con B las b sobrantes.
(50 puntos)
3. ¿Son deterministas sus autómatas?
(20 puntos)

2 Desarrollo

2.1 PDA M_1 para L_1

Para la estructura partiremos de la idea de aceptar los caracteres en el orden a, b, c y aceptar que i, j o k podrían ser 0 en algunos casos. Resultando en el siguiente automata:



Figure 1: Automata orden a, b, c

Para la pila (con el simbolo inicial \mathbb{Z}_0) podemos insertar una X cada vez que veamos una a o una b y luego quitar las X cada vez que vemos una c , lo que nos llevara a tener \mathbb{Z}_0 para el estado final de aceptacion solamente si la cantidad de a 's mas las de b 's son igual a la cantidad de c 's.

Por lo tanto transformamos las transiciones con a a las siguientes:

- $a, X/XX$
- $a, \mathbb{Z}_0/\mathbb{Z}_0X$

las transiciones con b a:

- $b, X/XX$
- $b, \mathbb{Z}_0/\mathbb{Z}_0X$

las transiciones con c a:

- $c, X/\varepsilon$

y por ultimo las transiciones con ε no deben afectar la pila:

- $\varepsilon, X/X$
- $\varepsilon, \mathbb{Z}_0/\mathbb{Z}_0$

Resultando en el siguiente automata de pila M_1 :

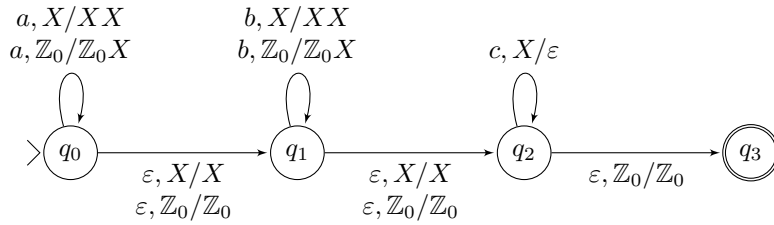


Figure 2: PDA M_1

2.2 PDA M_2 para L_2

Para la estructura partiremos del automata que acepta $L(a^*b^*)$ con una transición ε :

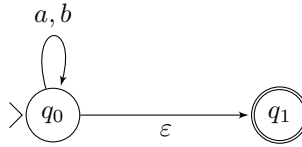


Figure 3: Automata a^*b^*

Para la pila utilizaremos la pista y agregaremos A 's para las a 's sobrantes y B 's para las b 's sobrantes, además deberemos de quitar una A si encontramos una b (que compensa a la a sobrante) y viceversa. Así entonces transformamos las transiciones con a a las siguientes:

- $a, \mathbb{Z}_0 / \mathbb{Z}_0 A$
- $a, A / AA$
- $a, B / \varepsilon$

las transiciones con b a:

- $b, \mathbb{Z}_0 / \mathbb{Z}_0 B$
- $b, B / BB$
- $b, A / \varepsilon$

y las transiciones con ε tendrán que llevarnos al estado de aceptación únicamente si existe una A o una B (a o b sobrante) al tope de la pila, de esta manera no se aceptarán cantidades iguales de a 's y b 's:

- $\varepsilon, A / A$
- $\varepsilon, B / B$

Finalmente nuestro PDA M_2 resultante:

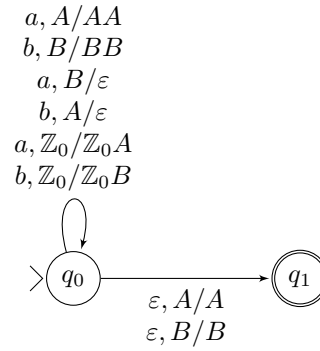


Figure 4: PDA M_2

2.3 ¿PDA's M_1 y M_2 deterministas?

Para saber si los PDA's son deterministas basta con fijarse en sus estructuras con las cuales los armamos. Para el PDA M_1 podemos ver que el automata con el cual lo construimos (ver Figura 2.1) es un NFA ya que el automata tiene que decidir entre leer un caracter o leer ε , esto mismo se aplica para la estructura con la cual armamos al PDA M_2 (ver Figura 2.2). Por lo tanto ninguno de nuestros PDA's son deterministas.