



TRABAJO PRÁCTICO N° 1
CADENAS, LENGUAJES Y OPERACIONES CON CADENAS
AUTÓMATAS FINITOS DETERMINISTAS Y NO DETERMINISTAS,
ACEPTACIÓN DE CADENAS

Integrantes:

- Mancuso, Augusto
- Narvaes, Franco
- Kark Verstraete, Augusto

PARTE A

Conceptos fundamentales: cadenas, lenguajes, operaciones con los lenguajes.

Ejercicio 1

Dado el alfabeto $\Sigma = \{a, b, c, d, 0, 1, 2, 3, 4\}$, obtenga dos cadenas, x e y , del alfabeto dado, indique la longitud de cada cadena y aplique las siguientes operaciones: concatenación de x e y , las potencias x_0 , x_1 , y_2 , y_3 .

Resolución:

$x = \{a, b, c, d\} \quad |x| = 4$

$y = \{0, 1, 2, 3, 4\} \quad |y| = 5$

$x_0 = \epsilon$

$x_1 = abcd$

$y_2 = 0123401234$

$y_3 = 012340123401234$

Ejercicio 2:

Dados los siguientes lenguajes, A , el conjunto de letras y B , el conjunto de dígitos, realice las siguientes operaciones $A \cup B$, $A \cap B$, $A.B$, A^3 , B^2 , B^0 , A^* , $A(A \cup B)^*$.

$A = \{a, b\} \quad |x| = 2$

$B = \{0, 1\} \quad |y| = 2$

Resolución:

$A \cup B = \{a, b, 0, 1\}$

$A \cap B = \{\epsilon\}$

$A.B = \{a0, a1, b0, b1\}$

A3=ababab

B 2=0101

A *= $\{\epsilon, a, b\}$

A (A \cup B)*= $\{aa, ab, a0, a1, ba, bb, b0, b1\}$

PARTE B

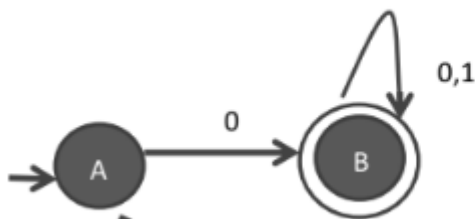
Conceptos fundamentales: autómatas finitos deterministas y no deterministas, aceptación de cadenas.

Ejercicio 1:

Obtenga el AFD y la tabla de transición.

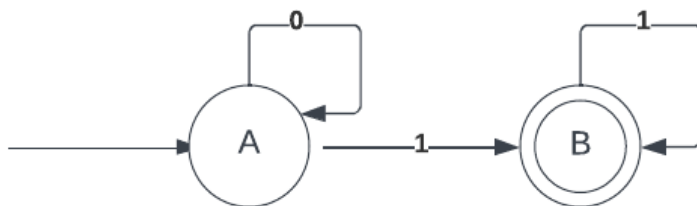
Ejemplo:

Dado el siguiente lenguaje definido en el alfabeto $\Sigma = \{0,1\}$. El AFD del conjunto de cadenas que inician en “0” y la tabla de transición son:



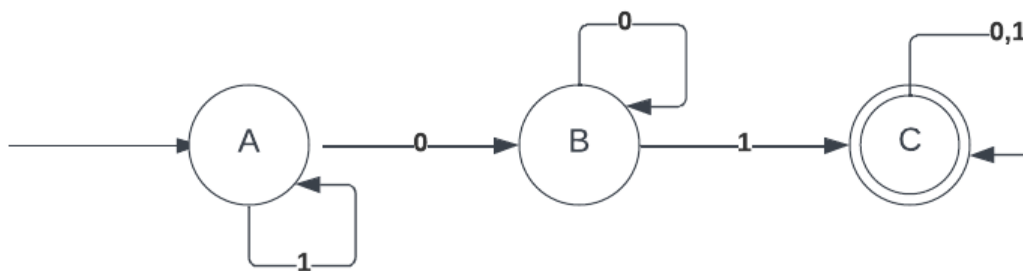
Estados	Alfabeto	
	0	1
A	B	
B	B	B

a) Del lenguaje definido en el alfabeto $\Sigma = \{0,1\}$, para el conjunto de cadenas que terminan en “1”.



Estado	Alfabeto	
	0	1
A	A	B
B		B

b) Del lenguaje definido en el alfabeto $\Sigma = \{0,1\}$, para el conjunto de cadenas que contienen a la subcadena “01”.



Estado	Alfabeto	
	0	1
A	B	A
B	B	C
C	C	C

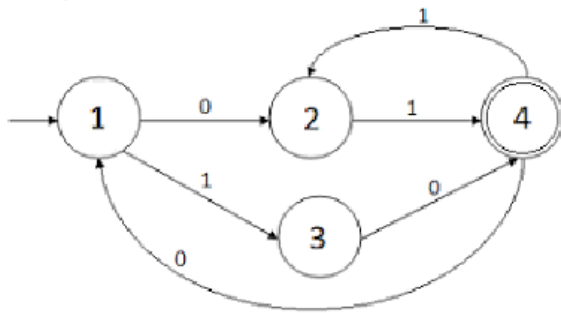
c) Del lenguaje definido en el alfabeto $\Sigma=\{a,b,c\}$, para el conjunto de cadenas que inician con la subcadena “ac”.



Estado	Alfabeto		
	a	b	c
A	B		
B			C
C	C	C	C

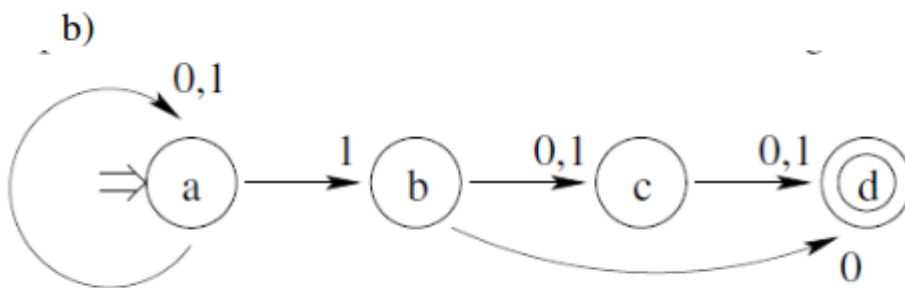
Ejercicio 2:

Indique en cada caso si es un AFD (autómata finito determinista) o un AFN (autómata finito no determinista).



a)

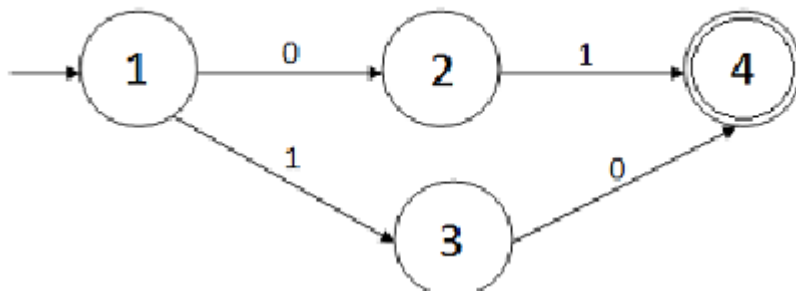
Este caso es determinista



determinista

Este caso es NO

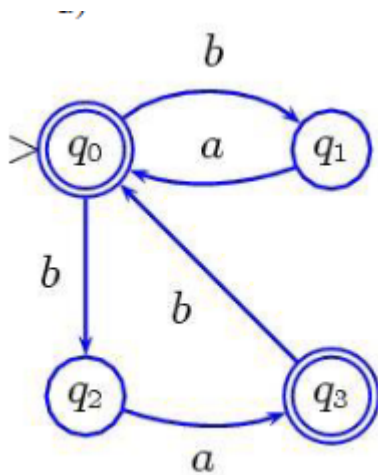
c)



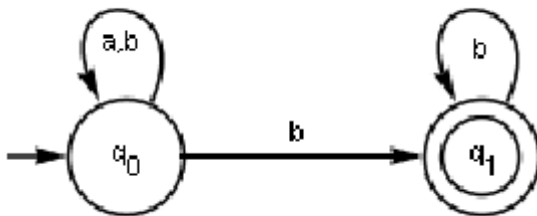
determinista

Este caso es

d)



Este caso es NO determinista



Este caso es NO determinista

Ejercicio 3:

Implementar mediante el lenguaje Python un autómata que reconozca números con signo o sin signo y en formato exponencial, usando una matriz de transición. Para implementar la matriz de transición en Python, se pueden utilizar listas. `tabla = [[2,1,1,"",""],[2,"","","",""],[2,"",3,5,8],["","","",""],[4,"","","",""],[4,"","","5,8],[7,6,6,"",""],[7,"","","",""],[7,"","","",8],['acepta','acepta','acepta','acepta','acepta','acepta']]`

```
class Automata:
```

```
    def __init__(self):
```

```
        self.table = [[2, 1, 1, "", "", ""],
                       [2, "", "", "", "", ""],
                       [2, "", "", 3, 5, 8],
                       [4, "", "", "", "", ""],
                       [4, "", "", "", 5, 8],
                       [7, 6, 6, "", "", ""],
                       [7, "", "", "", "", ""],
                       [7, "", "", "", 8],
                       ['acepta', 'acepta', 'acepta', 'acepta', 'acepta', 'acepta']]
```

```
        self.state = 0
```

```

self.number = input('Ingrese un número: ')
for i in range(len(self.number)):
    if self.state == 0:
        if self.number[i] == '+' or self.number[i] == '-':
            self.state = 1
            continue

        elif self.number[i].isdigit():

            self.state = 2
            continue
        else:
            print('No es un número')
            return

    if self.state == 1:
        if self.number[i].isdigit():

            self.state = 2
            continue
        else:
            print('No es un número')
            return

    if self.state == 2:
        if self.number[i].isdigit():

            self.state = 2
            continue
        elif self.number[i] == '.':

            self.state = 3
            continue
        elif self.number[i] == 'e' or self.number[i] == 'E':

            self.state = 5
            continue
        elif self.number[i] == '$':

            self.state = 8

        else:
            print('No es un número')
            return

```

```
if self.state == 3:
    if self.number[i].isdigit():

        self.state = 4
        continue
    else:
        print('No es un número')
        return

if self.state == 4:
    if self.number[i].isdigit():

        self.state = 4
        continue
    elif self.number[i] == 'e' or self.number[i] == 'E':

        self.state = 5
        continue
    elif self.number[i] == '$':

        self.state = 8

    else:
        print('No es un número')
        return

if self.state == 5:
    if self.number[i] == '+' or self.number[i] == '-':

        self.state = 6
        continue
    elif self.number[i].isdigit():

        self.state = 7
        continue
    else:
        print('No es un número')
        return

if self.state == 6:
    if self.number[i].isdigit():

        self.state = 7
```

```
        continue
    else:
        print('No es un número')
        return

if self.state == 7:
    if self.number[i].isdigit():

        self.state = 7
        continue
    elif self.number[i] == '$':

        self.state = 8

    else:
        print('No es un número')
        return

if self.state == 8:
    print('El número es correcto')
    return

if __name__ == '__main__':
    automata = Automata()
```