



Nombre de la práctica	AUTOMATAS FINITOS-U3			No.	2
Asignatura:	LENGUAJES Y AUTÓMATAS I	Carrera:	INGENIERÍA EN SISTEMAS COMPUTACIONALES-3501	Duración de la práctica (Hrs)	5 horas

NOMBRE DEL ALUMNO: Vanesa Hernández Martínez
GRUPO: 3501

I. Competencia(s) específica(s):

Crea y reconoce autómatas finitos en un lenguaje de programación para la solución de un problema.

Encuadre con CACEI: Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en la materia.

No. atributo	Atributos de egreso del PE que impactan en la asignatura	No. Criterio	Criterios de desempeño	No. Indicador	Indicadores
2	El estudiante diseñará esquemas de trabajo y procesos, usando metodologías congruentes en la resolución de problemas de Ingeniería en Sistemas Computacionales	CD1	Identifica metodologías y procesos empleados en la resolución de problemas	I1	Identificación y reconocimiento de distintas metodologías para la resolución de problemas
		CD2	Diseña soluciones a problemas, empleando metodologías apropiadas al área	I1	Uso de metodologías para el modelado de la solución de sistemas y aplicaciones
				I2	Diseño algorítmico (Representación de diagramas de transiciones)
3	El estudiante plantea soluciones basadas en tecnologías empleando su juicio ingenieril para valorar necesidades, recursos y resultados esperados.	CD1	Emplea los conocimientos adquiridos para el desarrollar soluciones	I1	Elección de metodologías, técnicas y/o herramientas para el desarrollo de soluciones
				I2	Uso de metodologías adecuadas para el desarrollo de proyectos
				I3	Generación de productos y/o proyectos
		CD2	Analiza y comprueba resultados	I1	Realizar pruebas a los productos obtenidos
				I2	Documentar información de las pruebas realizadas y los resultados

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

Laboratorio de cómputo y equipo de cómputo personal.

III. Material empleado:

- Equipo de cómputo
- Software para desarrollo (draw.io)

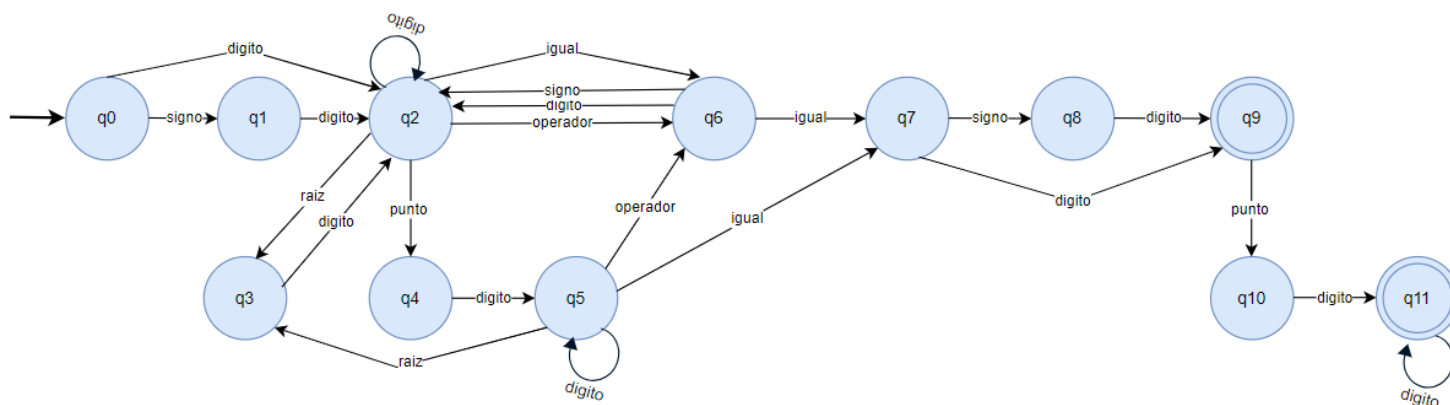
IV. Desarrollo de la práctica:

EJERCICIO 1

DESCRIPCION DEL PROBLEMA:

Genera un autómata que permita reconocer números enteros y decimales para reconocer las operaciones aritméticas básicas suma, resta, multiplicación, división, así como la raíz cuadrada. El ultimo símbolo que se genera en la cadena o cadenas deberá considerar el igual y la designación del resultado.

AUTOMATA FINITO NO DETERMINISTICO



DEFINICION FORMAL

AFN = $\langle S, \Sigma, p, i, F \rangle$

- $S = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}\}$
- $\Sigma = \{\text{signo, digito, punto, operador, igual, raiz}\}$
 - signo = {positivo +, negativo -}
 - digito = {0,1,2,3,4,5,6,7,8,9}
 - operador = {+, -, *, /, ^}
 - punto = {.
 - igual = {=}
 - raíz = {√}
- $p = \{(q_0, \text{signo}, q_1), (q_0, \text{digito}, q_2), (q_1, \text{digito}, q_1), (q_2, \text{digito}, q_2), (q_2, \text{operador}, q_6), (q_2, \text{punto}, q_3), (q_2, \text{raiz}, q_5), (q_2, \text{igual}, q_7), (q_3, \text{digito}, q_4), (q_4, \text{digito}, q_4), (q_4, \text{operador}, q_6), (q_4, \text{raiz}, q_5), (q_4, \text{igual}, q_7), (q_5, \text{digito}, q_2), (q_6, \text{signo}, q_2), (q_6, \text{digito}, q_2), (q_6, \text{igual}, q_7), (q_7, \text{signo}, q_8), (q_7, \text{digito}, q_9), (q_8, \text{digito}, q_9), (q_9, \text{digito}, q_9), (q_9, \text{punto}, q_{10}), (q_{10}, \text{digito}, q_{11}), (q_{11}, \text{digito}, q_{11})\}$

		signo	digito	operador	punto	raiz	igual
inicial	q0	q1	q2				
	q1		q1				
	q2		q2	q6	q3	q5	q7
	q3		q4				
	q4		q4	q6		q5	q7
	q5		q2				
	q6	q2	q2				q7
	q7	q8	q9				
	q8		q9				
final	q9		q9		q10		
	q10		q11				
final	q11		q11				

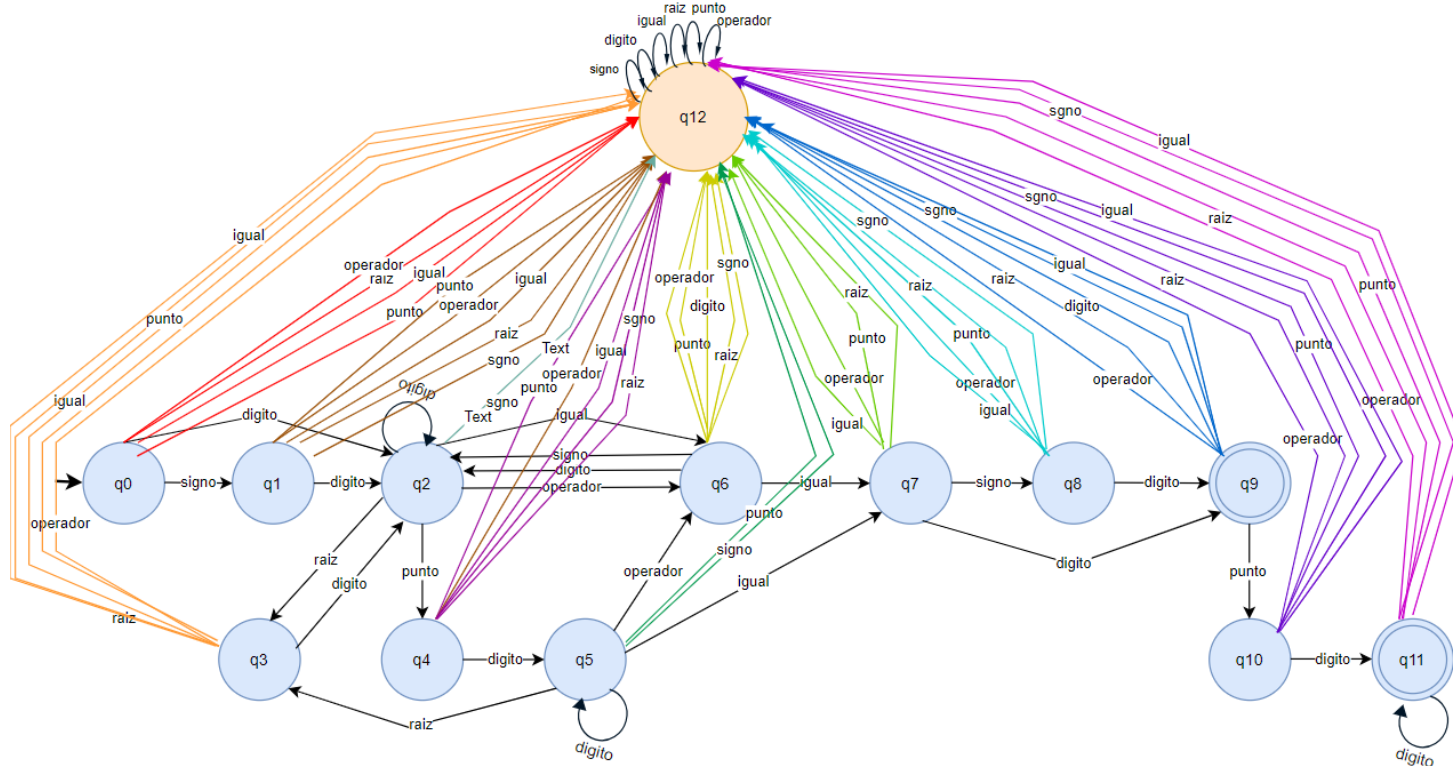
- $i = \{q0\}$
- $F = \{q9, q11\}$

LENGUAJE POR COMPRESION

$(\text{signo digito}^+ \text{ punto digito}^+ (\text{operador} \mid \text{raiz}) \text{ signo digito}^+ \text{ punto digito}^+ (\text{operador} \mid \text{igual}))^* \text{ signo digito}^+ \text{ punto digito}^+$

CONVERSION A AUTOMATA FINITO DETERMINISTICO

- ✓ Si se puede convertir, pero modificándolo estructuralmente





DEFINICION FORMAL UNA VEZ CONVERTIDO

AFN = $\langle S, \Sigma, \mathcal{S}, i, F \rangle$

- $S = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}\}$
- $\Sigma = \{\text{signo, digito, punto, operador, igual, raiz}\}$
 - signo = {positivo + , negativo -}
 - digito = {0,1,2,3,4,5,6,7,8,9}
 - operador = {+, -, *, /, ^}
 - punto = {.}
 - igual = {=}
 - raíz = {√}
- $\mathcal{S} = \{ (q_0, \text{signo}, q_1), (q_0, \text{digito}, q_2), (q_0, \text{operador}, q_{12}), (q_0, \text{punto}, q_{12}), (q_0, \text{raiz}, q_{12}), (q_0, \text{igual}, q_{12}), (q_1, \text{digito}, q_1), (q_1, \text{signo}, q_{12}), (q_1, \text{operador}, q_{12}), (q_1, \text{punto}, q_{12}), (q_1, \text{raiz}, q_{12}), (q_1, \text{igual}, q_{12}), (q_2, \text{signo}, q_{12}), (q_2, \text{digito}, q_2), (q_2, \text{operador}, q_6), (q_2, \text{punto}, q_3), (q_2, \text{raiz}, q_5), (q_2, \text{igual}, q_7), (q_3, \text{digito}, q_4), (q_3, \text{signo}, q_{12}), (q_3, \text{operador}, q_{12}), (q_3, \text{punto}, q_{12}), (q_3, \text{raiz}, q_{12}), (q_3, \text{igual}, q_{12}), (q_4, \text{digito}, q_4), (q_4, \text{operador}, q_6), (q_4, \text{raiz}, q_5), (q_4, \text{igual}, q_7), (q_4, \text{signo}, q_{12}), (q_4, \text{punto}, q_{12}), (q_5, \text{digito}, q_2), (q_5, \text{signo}, q_{12}), (q_5, \text{operador}, q_{12}), (q_5, \text{punto}, q_{12}), (q_5, \text{raiz}, q_{12}), (q_5, \text{igual}, q_{12}), (q_6, \text{signo}, q_2), (q_6, \text{digito}, q_2), (q_6, \text{igual}, q_7), (q_6, \text{operador}, q_{12}), (q_6, \text{punto}, q_{12}), (q_6, \text{raiz}, q_{12}), (q_7, \text{signo}, q_8), (q_7, \text{digito}, q_9), (q_7, \text{operador}, q_{12}), (q_7, \text{punto}, q_{12}), (q_7, \text{raiz}, q_{12}), (q_7, \text{igual}, q_{12}), (q_8, \text{digito}, q_9), (q_8, \text{signo}, q_{12}), (q_8, \text{operador}, q_{12}), (q_8, \text{punto}, q_{12}), (q_8, \text{raiz}, q_{12}), (q_8, \text{igual}, q_{12}), (q_9, \text{digito}, q_9), (q_9, \text{punto}, q_{10}), (q_9, \text{signo}, q_{12}), (q_9, \text{operador}, q_{12}), (q_9, \text{raiz}, q_{12}), (q_9, \text{igual}, q_{12}), (q_{10}, \text{digito}, q_{11}), (q_{10}, \text{signo}, q_{12}), (q_{10}, \text{operador}, q_{12}), (q_{10}, \text{punto}, q_{12}), (q_{10}, \text{raiz}, q_{12}), (q_{10}, \text{igual}, q_{12}), (q_{11}, \text{digito}, q_{11}), (q_{11}, \text{signo}, q_{12}), (q_{11}, \text{operador}, q_{12}), (q_{11}, \text{punto}, q_{12}), (q_{11}, \text{raiz}, q_{12}), (q_{11}, \text{igual}, q_{12}), (q_{12}, \text{signo}, q_{12}), (q_{12}, \text{digito}, q_{12}), (q_{12}, \text{operador}, q_6), (q_{12}, \text{punto}, q_3), (q_{12}, \text{raiz}, q_5), (q_{12}, \text{igual}, q_7), \}$

		signo	digito	operador	punto	raiz	igual
inicial	q0	q1	q2	q12	q12	q12	q12
	q1	q12	q1	q12	q12	q12	q12
	q2	q12	q2	q6	q3	q5	q7
	q3	q12	q4	q12	q12	q12	q12
	q4	q12	q4	q6	q12	q5	q7
	q5	q12	q2	q12	q12	q12	q12
	q6	q2	q2	q12	q12	q12	q7
	q7	q8	q9	q12	q12	q12	q12
	q8	q12	q9	q12	q12	q12	q12
final	q9	q12	q9	q12	q10	q12	q12
	q10	q12	q11	q12	q12	q12	q12
final	q11	q12	q11	q12	q12	q12	q12
	q12	q12	q11	q12	q12	q12	q12

- $i = \{q_0\}$
- $F = \{q_9, q_{11}\}$

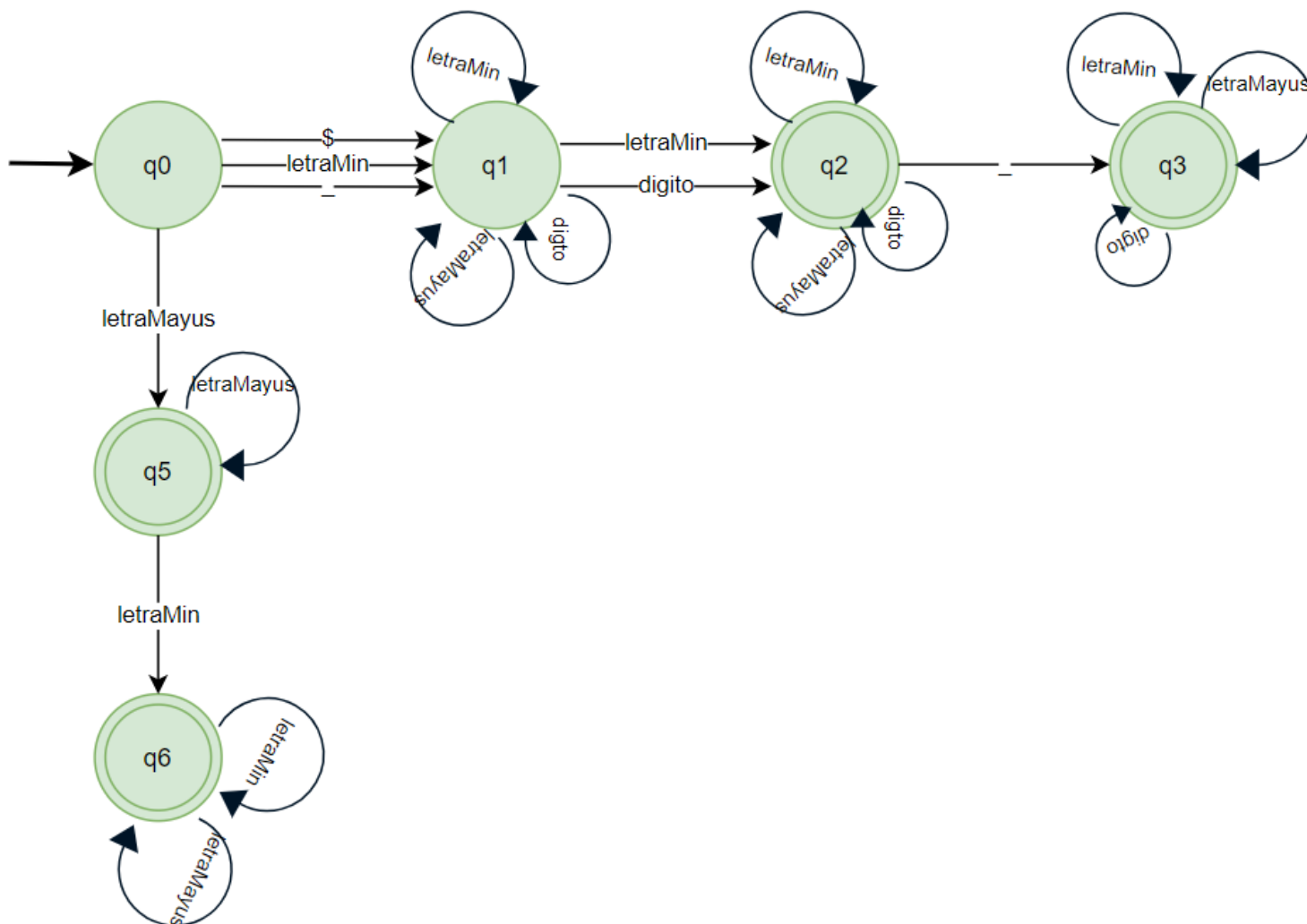
EJERCICIO 2

.....

DESCRIPCION DEL PROBLEMA:

Generar un autómata finito no determinístico que reconozca nombres de variables, constantes y clases.

AUTOMATA FINITO NO DETERMINISTICO



DEFINICION FORMAL

AFN = $\langle S, \Sigma, p, i, F \rangle$

- $S = \{ q0, q1, q2, q3, q4, q5 \}$
- $\Sigma = \{ \text{letraMin}, \text{letraMayus}, \text{digito}, \text{carácter}, \text{guion bajo} \}$
 - $\text{letraMin} = \{ [a-z] \}$
 - $\text{letraMayus} = \{ [A-Z] \}$
 - $\text{digito} = \{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 \}$
 - $\text{carácter} = \{ \$ \}$
 - $\text{guion bajo} = \{ _ \}$



- $P = \{ (q0, \text{letraMin}, q1), (q0, \text{letraMayus}, q4), (q0, \text{carácter}, q1), (q0, \text{guion bajo}, q1), (q1, \text{letraMin}, q1), (q1, \text{letraMin}, q2), (q1, \text{letraMayus}, q1), (q1, \text{digito}, q1), (q1, \text{digito}, q2), (q2, \text{letraMin}, q2), (q2, \text{letraMayus}, q2), (q2, \text{digito}, q2), (q2, \text{guion bajo}, q3), (q3, \text{letraMin}, q3), (q3, \text{letraMayus}, q3), (q3, \text{digito}, q3), (q4, \text{letraMin}, q5), (q4, \text{letraMayus}, q4), (q5, \text{letraMin}, q5), (q5, \text{letraMayus}, q5) \}$

		letraMin	letraMayus	digito	carácter	guion bajo
inicial	q0	q1	q4		q1	q1
	q1	q1,q2	q1	q1,q2		
final	q2	q2	q2	q2		q3
final	q3	q3	q3	q3		
final	q4	q5	q4			
final	q5	q5	q5			

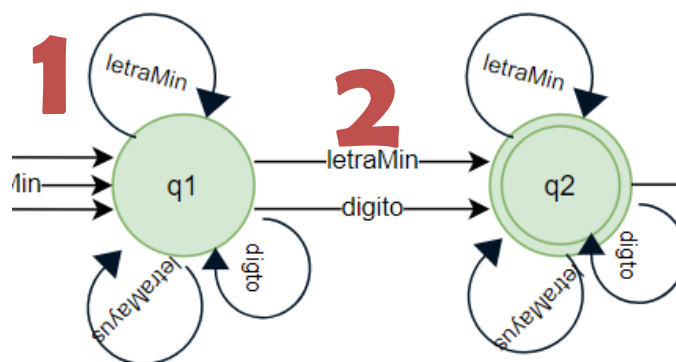
- $i = \{ q0 \}$
- $F = \{ q2, q3, q4, q5 \}$

LENGUAJE POR COMPRENSION

$(\$ | \text{letraMin} | _) \text{letraMin}^* \text{digito}^* \text{letraMayus}^* (\text{letraMin} | \text{digito}) \text{letraMin}^* \text{digito}^* \text{letraMayus}^*) | (\$ | \text{letraMin} | _) \text{letraMin}^* \text{digito}^* \text{letraMayus}^* (\text{letraMin} | \text{digito}) \text{letraMin}^* \text{digito}^* \text{letraMayus}^* \text{letraMin}^* \text{digito}^* \text{letraMayus}^*) | (\text{letraMayus}^+) | (\text{letraMayus}^+ (\text{letraMin}^+ \text{letraMayus}^* \text{letraMin}^*)$

CONVERSION A AUTOMATA FINITO DETERMINISTICO

- ✓ No se puede convertir porque afecta el lenguaje generado



	letraMin
q0	q1
q1	q1,q2
q2	q2

Desde la creación de la tabla nos podemos dar cuenta que de la entidad q0 salen dos flechas con la misma etiqueta de letraMin, y aunque se intento ver si se podía acomodar el diagrama de una manera distinta para que esto no ocurriera, no se pudo modificar su estructura sin afectar su lenguaje por lo tanto este AFN no se puede convertir un un AFD.

DEFINICION FORMAL UNA VEZ CONVERTIDO

V. Conclusiones:

Los autómatas finitos son modelos matemáticos fundamentales en la teoría de lenguajes formales y autómatas, utilizados para representar y procesar cadenas dentro de un lenguaje. Se clasifican principalmente en autómatas finitos determinísticos (AFD) y no determinísticos (AFN), cada uno con características únicas que los hacen valiosos en diferentes contextos.

Los **autómatas finitos determinísticos (AFD)** se caracterizan por tener una única transición para cada símbolo del alfabeto desde cualquier estado, lo que implica que, para una entrada dada, el autómata siempre sigue un solo camino definido. Esta característica los hace eficientes y simples de implementar en dispositivos o algoritmos, pero a veces requieren más estados para representar un mismo lenguaje.

Por otro lado, los **autómatas finitos no determinísticos (AFN)** permiten múltiples transiciones o incluso ninguna para un símbolo dado, lo que les otorga mayor flexibilidad y simplicidad al definir ciertos lenguajes. Sin embargo, esta flexibilidad trae complejidad, ya que en cada paso el autómata puede seguir varios caminos al mismo tiempo.

En cuanto a sus aplicaciones, los autómatas finitos, ya sean determinísticos o no, son esenciales en áreas como el reconocimiento de patrones, la validación de cadenas (como en expresiones regulares), la construcción de parsers en compiladores y la definición de protocolos de comunicación. La conversión de AFN a AFD es especialmente importante en estos contextos, ya que permite pasar de un diseño teórico y flexible a uno implementable de manera eficiente.