

LICENCIATURA EN CIENCIAS COMPUTACIONALES
AUTÓMATAS Y COOMPILADORES

REPORTE DE PRÁCTICA 2

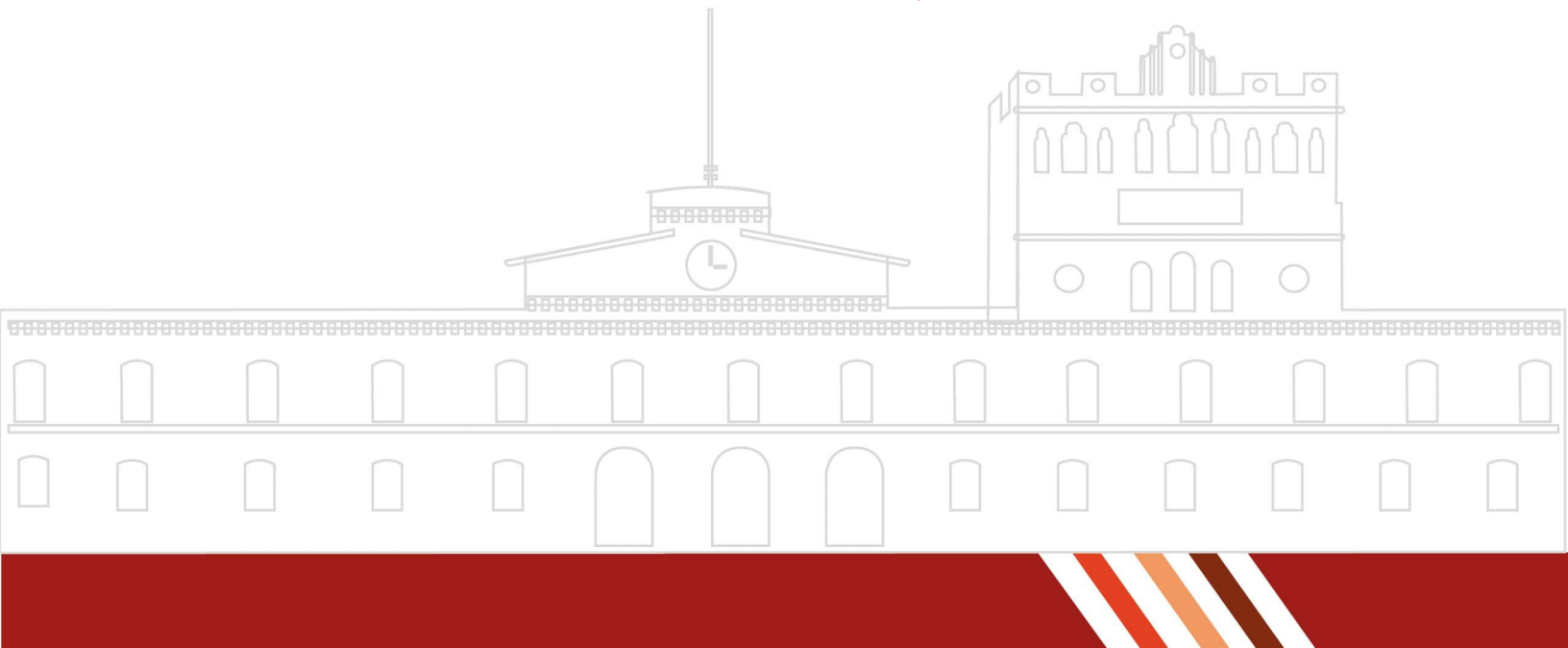
AFD y AFND

ALUMNO: CHRISTIAN LÓPEZ SOLÍS

DR. EDUARDO CORNEJO VELÁZQUEZ

26 DE FEBRERO DEL 2025

MINERAL DE LA REFORMA, PACHUCA



1. Introducción

En la materia de Automátas y Compiladores, los Autómatas Finitos Deterministas (AFD) y No Deterministas (AFND) son esenciales para entender los lenguajes formales, los cuales son fundamentales para el diseño de compiladores y sistemas que procesan texto. Mientras que un AFD tiene transiciones predecibles y únicas, un AFND permite transiciones múltiples o incluso la ausencia de transiciones, brindando mayor flexibilidad pero con menor eficiencia. Ambos modelos son clave en el reconocimiento de lenguajes regulares.

2. Objetivo

El objetivo principal es entender la teoría de lenguajes formales y cómo se representan con AFD y AFND. Se busca diferenciar entre ambos tipos de autómatas, aprender a diseñarlos para lenguajes específicos, convertir entre AFND y AFD, y aplicar este conocimiento en la creación de analizadores léxicos de compiladores.

3. Marco Teórico

Lenguajes Formales

Un lenguaje formal es un conjunto de cadenas de símbolos de un alfabeto específico, utilizado para describir sistemas y estructuras en computación. Los lenguajes formales son fundamentales para la teoría de la computación, ya que permiten representar problemas y soluciones de forma estructurada. El concepto de alfabeto y palabra es clave, donde un alfabeto es un conjunto de símbolos, y una palabra es una secuencia finita de estos símbolos.

Autómatas

Un autómata es un modelo matemático que describe un sistema mediante estados y transiciones. Los autómatas finitos deterministas (AFD) tienen una única transición para cada estado y símbolo, mientras que los autómatas finitos no deterministas (AFND) pueden tener múltiples transiciones. Los autómatas con transiciones epsilon permiten transitar sin consumir símbolos de entrada.

4. Herramientas Empleadas

YouTube

Se utilizaron recursos educativos en YouTube para comprender y profundizar en temas clave de lenguajes formales y autómatas. Los videos proporcionaron una forma accesible y visual de explicar conceptos complejos, como la definición de autómatas, las operaciones con lenguajes formales y la minimización de estados.

Editor de Textos LaTeX

Para la redacción del informe y la organización del marco teórico, se empleó LaTeX, una herramienta eficiente para la creación de documentos científicos, que permite organizar y estructurar contenido de manera clara y precisa, con un manejo adecuado de las referencias bibliográficas y el formato adecuado para la presentación de ecuaciones y gráficos.

5. Desarrollo

º Ejercicio 1:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $= \{0, 1\}$, que acepte el conjunto de palabras que inician en “0”.

Tupla:

- El alfabeto: $\Sigma = \{0, 1\}$
- Lenguaje: $L = \{\text{conjunto de palabras que inician en } 0\}$
- Conjunto de estados: $Q = \{a, b\}$
- Función de transición:
 - $f(a, 0) \rightarrow b$
 - $f(b, 0) \rightarrow b$
 - $f(b, 1) \rightarrow b$
- Estado inicial: $q_0 = \{a\}$
- Estados finales: $F = \{b\}$

Tabla de Transiciones:

Table 1: Tabla de transiciones.

	0	1
a	b	
b	b	b

Diagrama de Transiciones:

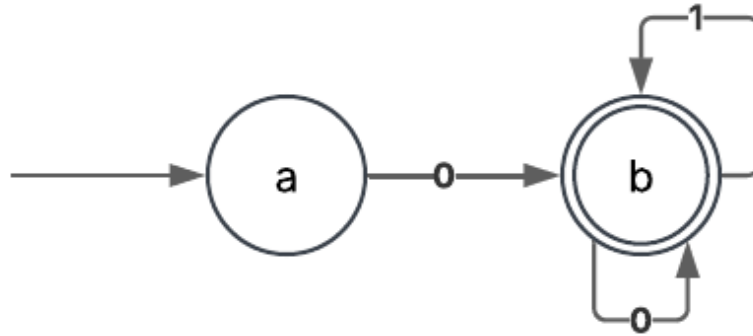


Figure 1: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

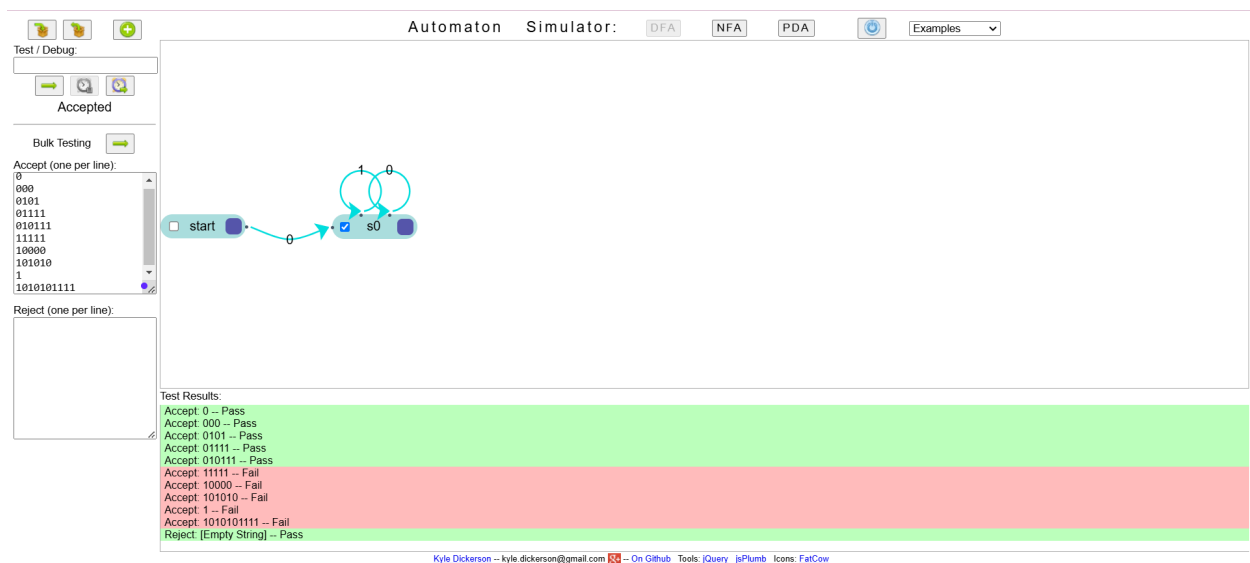


Figure 2: Simulador en Automaton.

º Ejercicio 2:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $= \{0, 1\}$, que acepte el conjunto de palabras que inician en “0”.

Tupla:

- El alfabeto: $\Sigma = \{0, 1\}$
- Lenguaje: $L = \{\text{conjunto de palabras que terminan en } 1\}$
- Conjunto de estados: $Q = \{a, b\}$
- Función de transición:
 - $f(a, 0) \rightarrow a$
 - $f(a, 1) \rightarrow b$
 - $f(b, 0) \rightarrow a$
 - $f(b, 1) \rightarrow b$
- Estado inicial: $q_0 = \{a\}$
- Estados finales: $F = \{b\}$

Tabla de Transiciones:

Table 2: Tabla de transiciones.

	0	1
a	a	b
b	a	b

Diagrama de Transiciones:

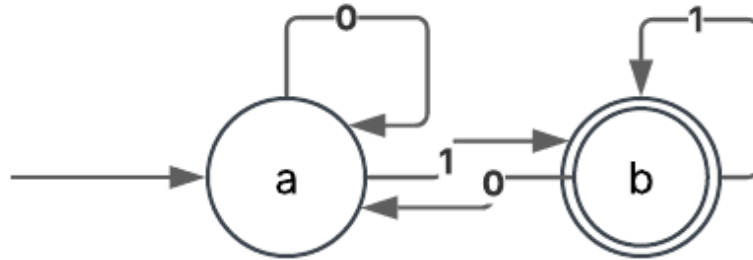


Figure 3: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):



Figure 4: Simulador en Automaton.

9 Ejercicio 3:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $= \{0, 1\}$, que acepte el conjunto de palabras que contienen la subcadena "01".

Tupla:

- El alfabeto: $\Sigma = \{0, 1\}$
- Lenguaje: $L = \{\text{conjunto de palabras que contienen la subcadena "01"}\}$
- Conjunto de estados: $Q = \{a, b, c\}$
- Función de transición:
 - $f(a, 0) \rightarrow b$
 - $f(a, 1) \rightarrow a$
 - $f(b, 0) \rightarrow b$
 - $f(b, 1) \rightarrow c$
 - $f(c, 0) \rightarrow c$
 - $f(c, 1) \rightarrow c$
- Estado inicial: $q_0 = \{a\}$
- Estados finales: $F = \{c\}$

Tabla de Transiciones:

Table 3: Tabla de transiciones.

	0	1
a	b	a
b	b	c
c	c	c

Diagrama de Transiciones:

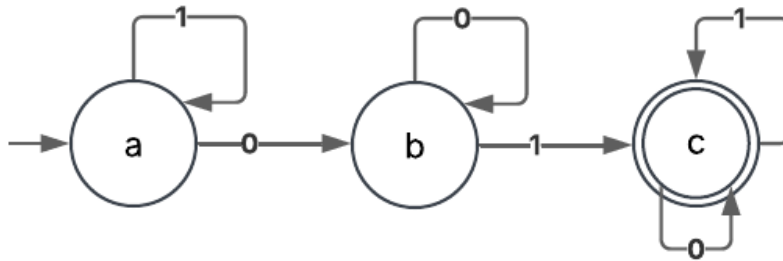


Figure 5: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

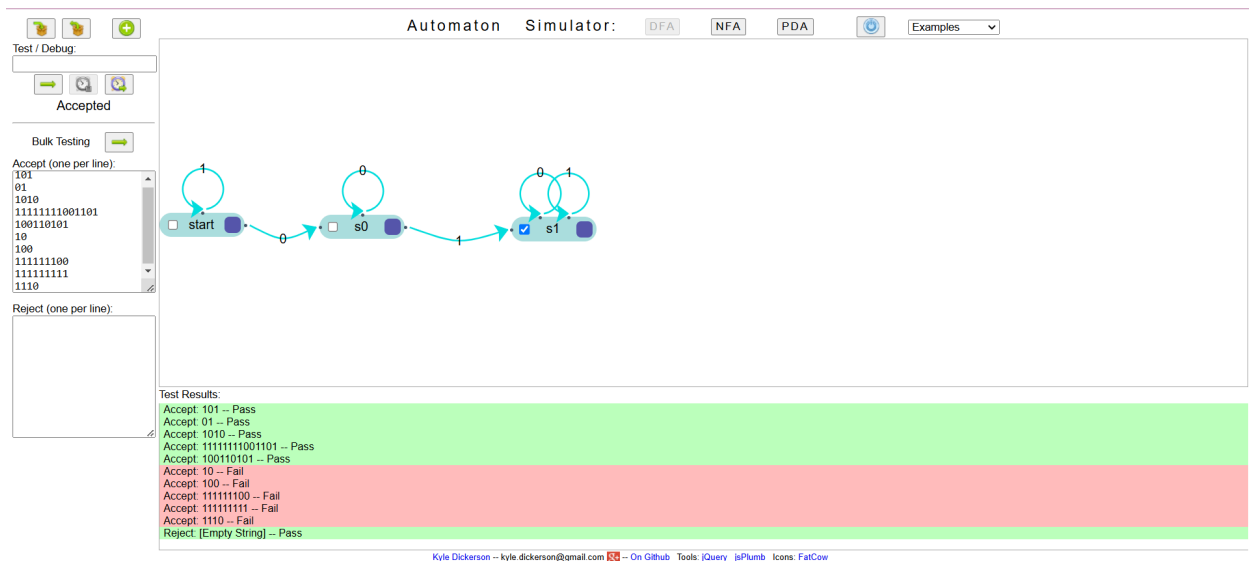


Figure 6: Simulador en Automaton.

9 Ejercicio 4:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $= \{0, 1\}$, que acepte el conjunto de palabras que no contienen la subcadena "01".

Tupla:

- El alfabeto: $\Sigma = \{0, 1\}$
- Lenguaje: $L = \{\text{conjunto de palabras que no contienen la subcadena "01"}\}$
- Conjunto de estados: $Q = \{a, b, c\}$
- Función de transición:
 - $f(a, 0) \rightarrow b$
 - $f(a, 1) \rightarrow a$
 - $f(b, 0) \rightarrow b$
 - $f(b, 1) \rightarrow c$
 - $f(c, 0) \rightarrow c$
 - $f(c, 1) \rightarrow c$
- Estado inicial: $q_0 = \{a\}$
- Estados finales: $F = \{a, b\}$

Tabla de Transiciones:

Table 4: Tabla de transiciones.

	0	1
a	b	a
b	b	c
c	c	c

Diagrama de Transiciones:

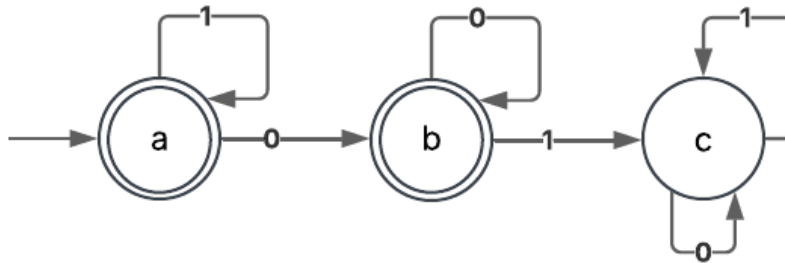


Figure 7: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

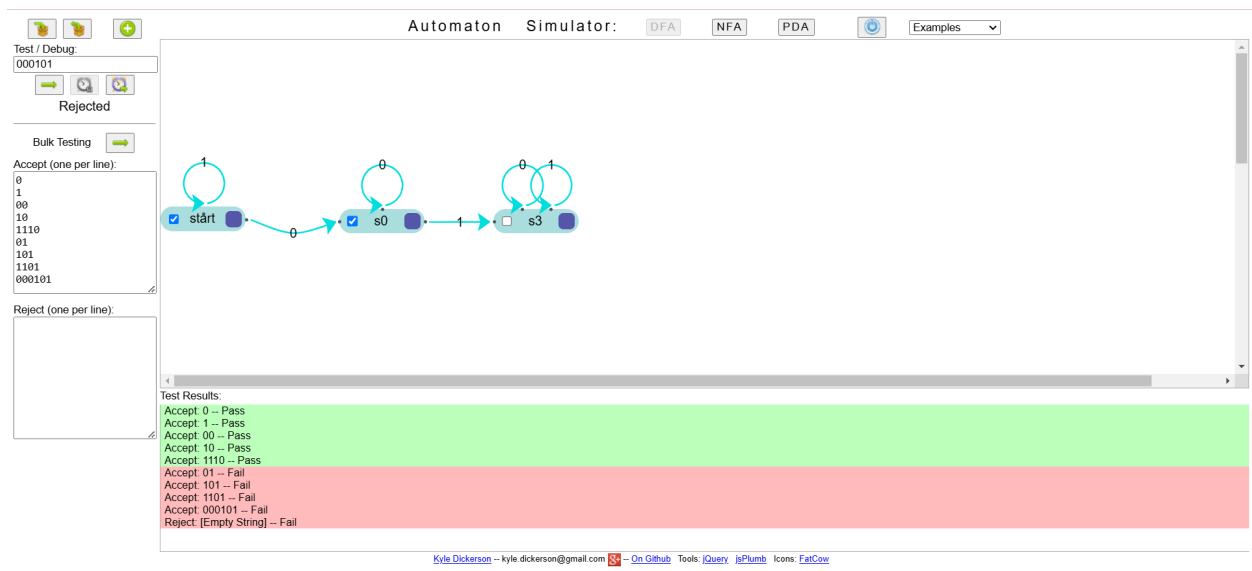


Figure 8: Simulador en Automaton.

9 Ejercicio 5:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $= \{a, b, c\}$, que acepte el conjunto de palabras que inician con la subcadena “ac” o terminan con la subcadena “ab”.

Tupla:

- El alfabeto: $\Sigma = \{a, b, c\}$
- Lenguaje: $L = \{\text{conjunto de palabras que inician con la subcadena "ac" o terminan con la subcadena "ab"}\}$
- Conjunto de estados: $Q = \{1, 2, 3, 4, 5, 6\}$
- Función de transición:
 - $f(1, a) \rightarrow 2$
 - $f(1, b) \rightarrow 1$
 - $f(1, c) \rightarrow 1$
 - $f(2, a) \rightarrow 5$
 - $f(2, b) \rightarrow 4$
 - $f(2, c) \rightarrow 3$
 - $f(3, a) \rightarrow 3$
 - $f(3, b) \rightarrow 3$
 - $f(3, c) \rightarrow 3$
 - $f(4, a) \rightarrow 5$
 - $f(4, b) \rightarrow 6$
 - $f(4, c) \rightarrow 6$
 - $f(5, a) \rightarrow 5$
 - $f(5, b) \rightarrow 4$
 - $f(5, c) \rightarrow 6$
 - $f(6, a) \rightarrow 5$
 - $f(6, b) \rightarrow 6$
 - $f(6, c) \rightarrow 6$
- Estado inicial: $q_0 = \{1\}$
- Estados finales: $F = \{3, 4\}$

Tabla de Transiciones:

Table 5: Tabla de transiciones.

	a	b	c
1	2	1	1
2	5	4	3
3	3	3	3
4	5	6	6
5	5	4	6
6	5	6	6

Diagrama de Transiciones:

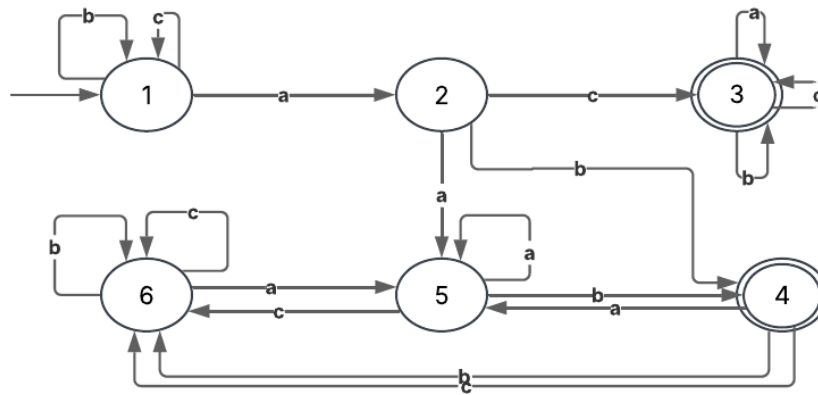


Figure 9: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

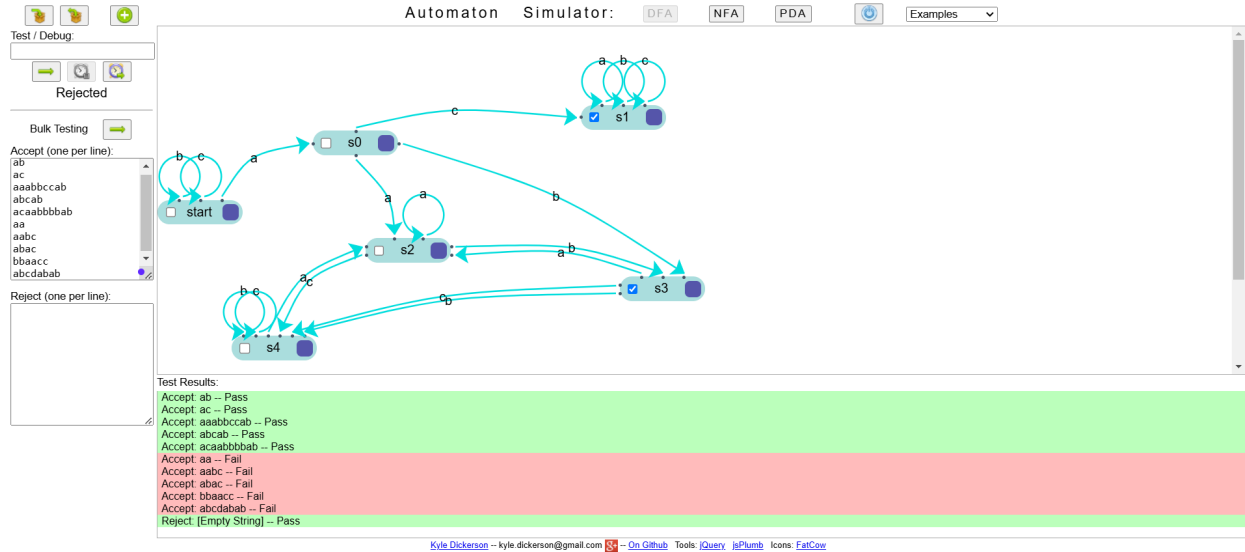


Figure 10: Simulador en Automaton.

◻ Ejercicio 6:

Planteamiento: Obtenga un Automata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $\Sigma = \{a, b, c\}$, que acepte el conjunto de palabras que inician con la subcadena "ac" y no terminan con la subcadena "ab".

Tupla:

- El alfabeto: $\Sigma = \{a, b, c\}$
- Lenguaje: $L = \{ \text{conjunto de palabras que inician con la subcadena "ac" y no terminan con la subcadena "ab"} \}$
- Conjunto de estados: $Q = \{1, 2, 3, 4, 5\}$
- Función de transición:
 - $f(1, a) \rightarrow 2$
 - $f(2, c) \rightarrow 2$
 - $f(3, a) \rightarrow 4$
 - $f(3, b) \rightarrow 3$
 - $f(3, c) \rightarrow 3$
 - $f(4, a) \rightarrow 4$
 - $f(4, b) \rightarrow 5$
 - $f(4, c) \rightarrow 3$
 - $f(5, a) \rightarrow 4$
 - $f(5, b) \rightarrow 3$
 - $f(5, c) \rightarrow 3$
- Estado inicial: $q_0 = \{1\}$
- Estados finales: $F = \{3, 4\}$

Tabla de Transiciones:

Table 6: Tabla de transiciones.

	a	b	c
1	2		
2			3
3	4	3	3
4	4	5	3
5	4	3	3

Diagrama de Transiciones:

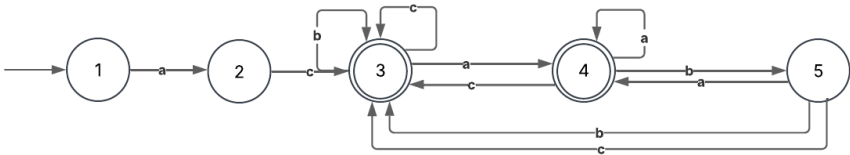


Figure 11: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

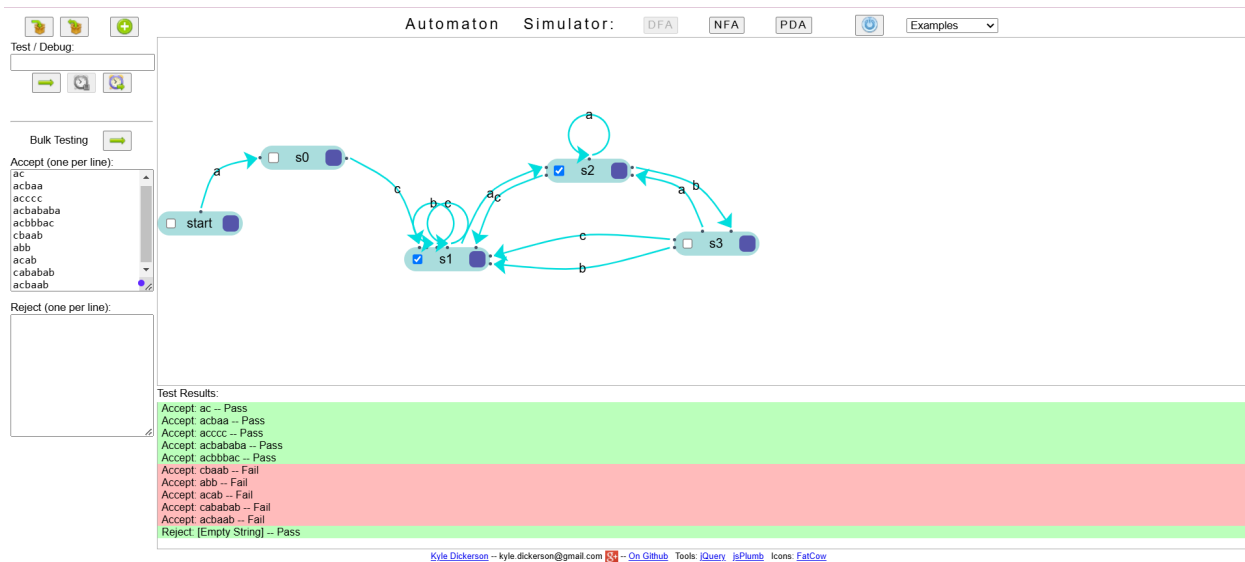


Figure 12: Simulador en Automaton.

◦ Ejercicio 7:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $\Sigma = \{a, b, c\}$, que acepte el conjunto de palabras que inician con la subcadena “ac” o no terminan con la subcadena “ab”.

Tupla:

- El alfabeto: $\Sigma = \{a, b, c\}$
- Lenguaje: $L = \{\text{conjunto de palabras que inician con la subcadena "ac" o no terminan con la subcadena "ab"}\}$
- Conjunto de estados: $Q = \{1, 2, 3, 4, 5, 6\}$
- Función de transición:
 - $f(1, a) \rightarrow 2$
 - $f(1, b) \rightarrow 6$
 - $f(1, c) \rightarrow 6$
 - $f(2, a) \rightarrow 5$
 - $f(2, b) \rightarrow 4$

- $f(2, c) \rightarrow 3$
- $f(3, a) \rightarrow 3$
- $f(3, b) \rightarrow 3$
- $f(3, c) \rightarrow 3$
- $f(4, a) \rightarrow 5$
- $f(4, b) \rightarrow 6$
- $f(4, c) \rightarrow 6$
- $f(5, a) \rightarrow 5$
- $f(5, b) \rightarrow 4$
- $f(5, c) \rightarrow 6$
- $f(6, a) \rightarrow 5$
- $f(6, b) \rightarrow 6$
- $f(6, c) \rightarrow 6$
- Estado inicial: $q_0 = \{1\}$
- Estados finales: $F = \{1, 2, 3, 5, 6\}$

Tabla de Transiciones:

Table 7: Tabla de transiciones.

	a	b	c
1	2	6	6
2	5	4	3
3	3	3	3
4	5	6	6
5	5	4	6
6	5	6	6

Diagrama de Transiciones:

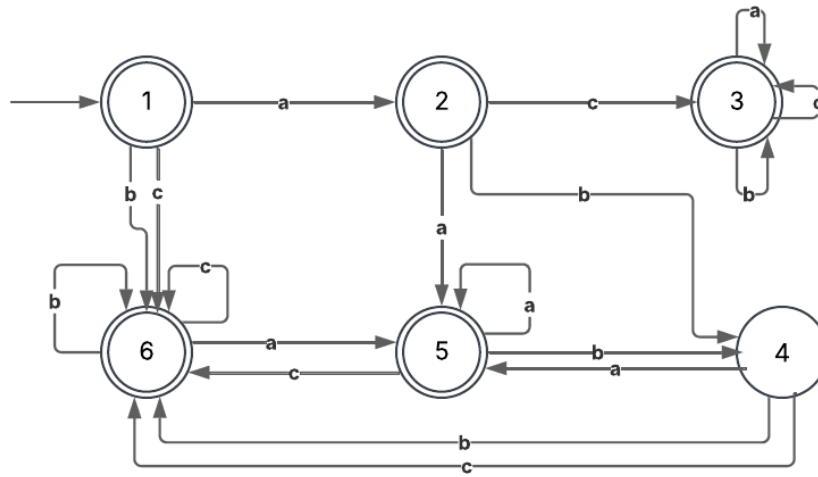


Figure 13: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

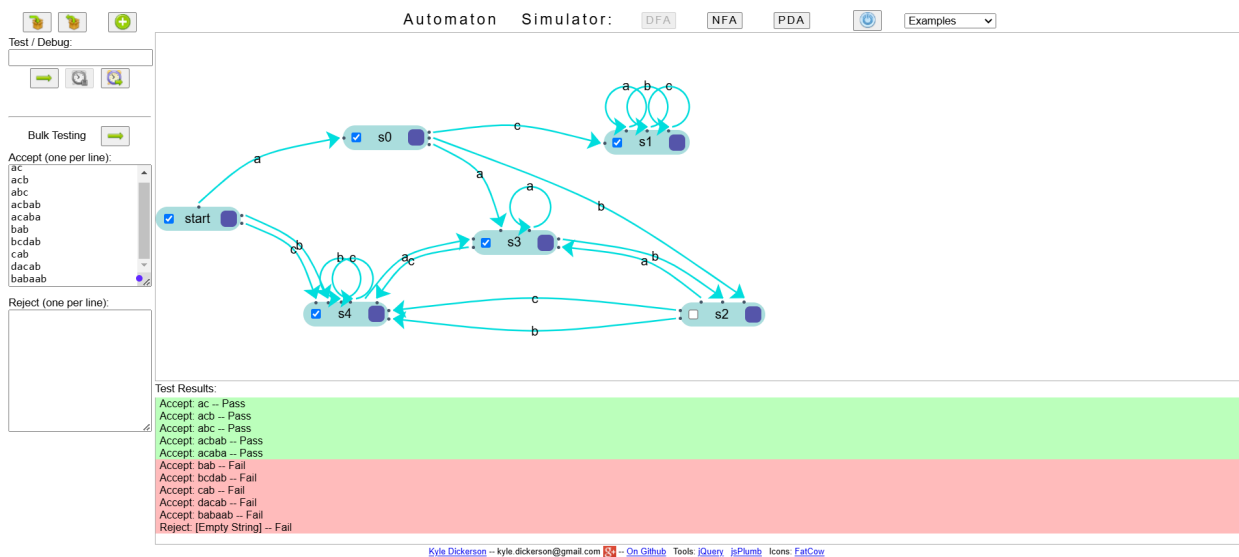


Figure 14: Simulador en Automaton.

9 Ejercicio 8:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $= \{a, b, c\}$, que acepte el conjunto de palabras que no inician con la subcadena “ac” y no terminan con la subcadena “ab”.

Tupla:

- El alfabeto: $\Sigma = \{a, b, c\}$
- Lenguaje: $L = \{\text{conjunto de palabras que no inician con la subcadena "ac" y no terminan con la subcadena "ab"}\}$
- Conjunto de estados: $Q = \{1, 2, 3, 4, 5, 6\}$
- Función de transición:
 - $f(1, a) \rightarrow 2$
 - $f(1, b) \rightarrow 6$
 - $f(1, c) \rightarrow 6$
 - $f(2, a) \rightarrow 5$
 - $f(2, b) \rightarrow 4$
 - $f(2, c) \rightarrow 3$
 - $f(3, a) \rightarrow 3$
 - $f(3, b) \rightarrow 3$
 - $f(3, c) \rightarrow 3$
 - $f(4, a) \rightarrow 5$
 - $f(4, b) \rightarrow 6$
 - $f(4, c) \rightarrow 6$
 - $f(5, a) \rightarrow 5$
 - $f(5, b) \rightarrow 4$
 - $f(5, c) \rightarrow 6$
 - $f(6, a) \rightarrow 5$
 - $f(6, b) \rightarrow 6$
 - $f(6, c) \rightarrow 6$
- Estado inicial: $q_0 = \{1\}$
- Estados finales: $F = \{1, 2, 5, 6\}$

Tabla de Transiciones:

Table 8: Tabla de transiciones.

	a	b	c
1	2	6	6
2	5	4	3
3	3	3	3
4	5	6	6
5	5	4	6
6	5	6	6

Diagrama de Transiciones:

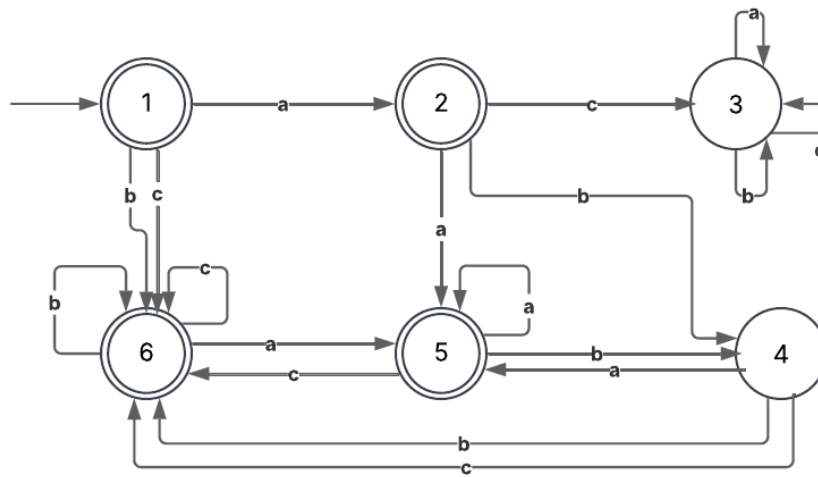


Figure 15: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

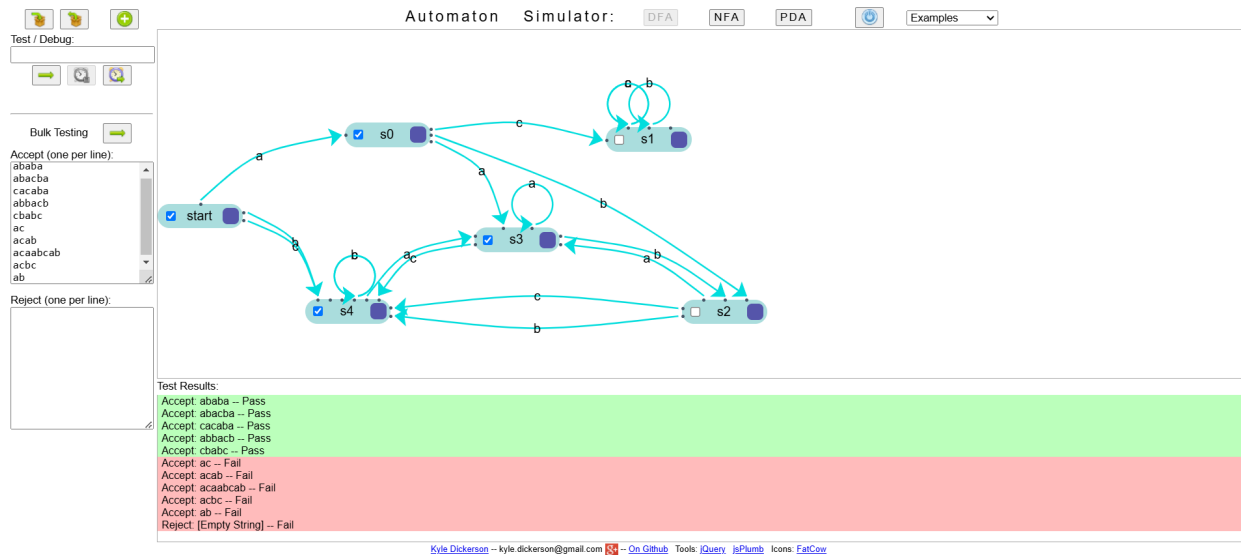


Figure 16: Simulador en Automaton.

9 Ejercicio 9:

Planteamiento: Obtenga un Autómata Finito Determinista (AFND) dado el lenguaje definido en el alfabeto $\Sigma = \{0, 1\}$, que acepte el conjunto de palabras que no contienen la subcadena "01".

Tupla:

- El alfabeto: $\Sigma = \{0, 1\}$
- Lenguaje: $L = \{\text{conjuntodepalabrasquenoclientenenlasubcadena "01"}\}$
- Conjunto de estados: $Q = \{a, b, c\}$
- Función de transición:
 - $f(a, 0) \rightarrow b$
 - $f(a, 1) \rightarrow a$
 - $f(b, 0) \rightarrow b$
 - $f(b, 1) \rightarrow c$
 - $f(c, 0) \rightarrow c$
 - $f(c, 1) \rightarrow c$

- Estado inicial: $q_0 = \{a\}$
- Estados finales: $F = \{a, b\}$

Tabla de Transiciones:

Table 9: Tabla de transiciones.

	0	1
a	b	a
b	b	c
c	c	c

Diagrama de Transiciones:

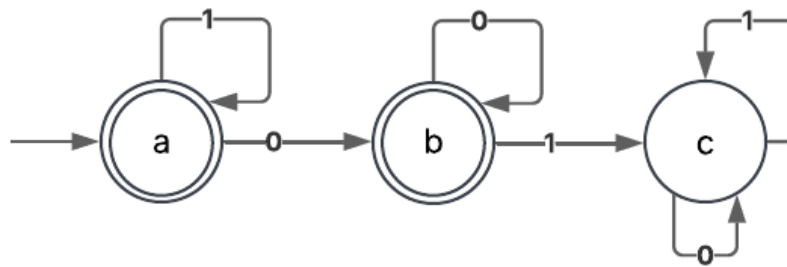


Figure 17: Diagrama de transiciones.

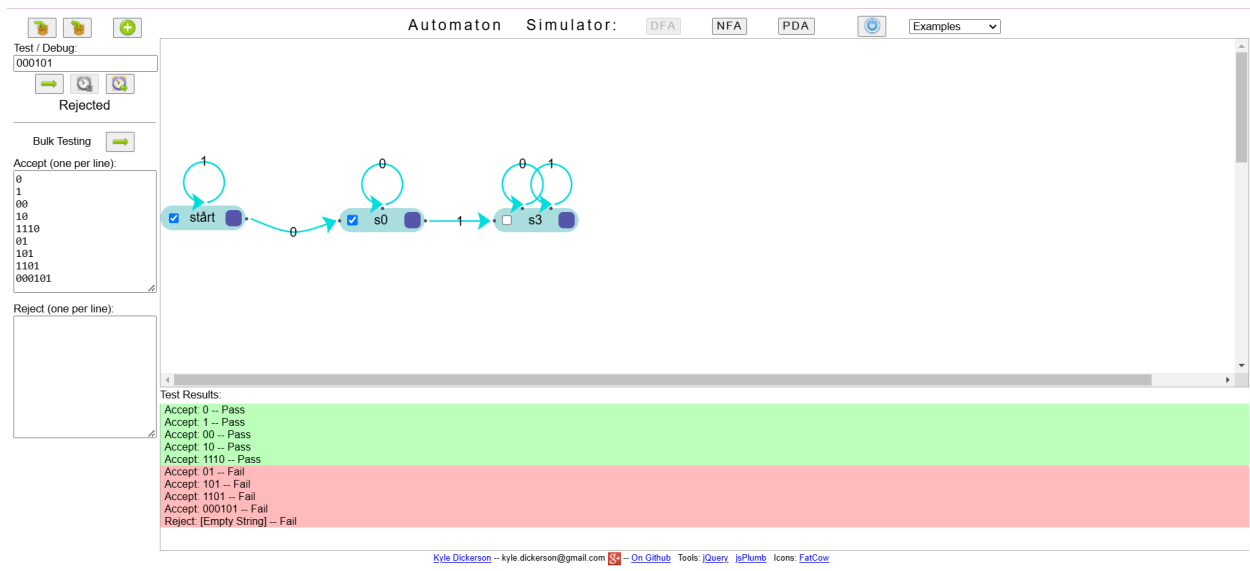


Figure 18: Simulador en Automaton.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

º Ejercicio 10:

Planteamiento: Obtenga un Autómata Finito Determinista (AFD) dado el lenguaje definido en el alfabeto $\Sigma = \{a, b, c\}$, que acepte el conjunto de palabras que inician con la subcadena "ac" y terminan con la subcadena "ab".

Tupla:

- El alfabeto: $\Sigma = \{a, b, c\}$
- Lenguaje: $L = \{\text{conjunto de palabras que inician con la subcadena "ac" y terminan con la subcadena "ab"}\}$
- Conjunto de estados: $Q = \{1, 2, 3, 4, 5\}$
- Función de transición:
 - $f(1, a) \rightarrow 2$
 - $f(2, c) \rightarrow 3$
 - $f(3, a) \rightarrow 4$
 - $f(3, b) \rightarrow 3$
 - $f(3, c) \rightarrow 3$
 - $f(4, a) \rightarrow 4$

- $f(4, b) \rightarrow 5$
- $f(4, c) \rightarrow 3$
- $f(5, a) \rightarrow 4$
- $f(5, b) \rightarrow 3$
- $f(5, c) \rightarrow 3$
- Estado inicial: $q_0 = \{1\}$
- Estados finales: $F = \{5\}$

Tabla de Transiciones:

Table 10: Tabla de transiciones.

	a	b	c
1	2		
2			3
3	4	3	3
4	4	5	3
5	4	3	3

Diagrama de Transiciones:

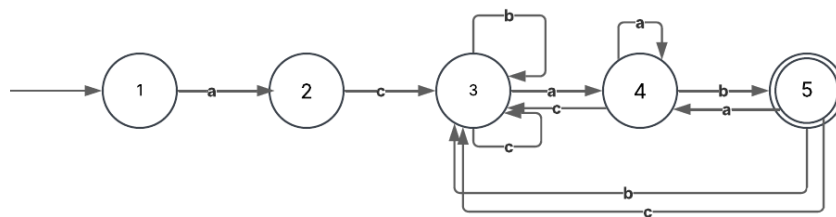


Figure 19: Diagrama de transiciones.

Simulador en Automaton Simulador y palabras aceptadas (5) y rechazadas (5):

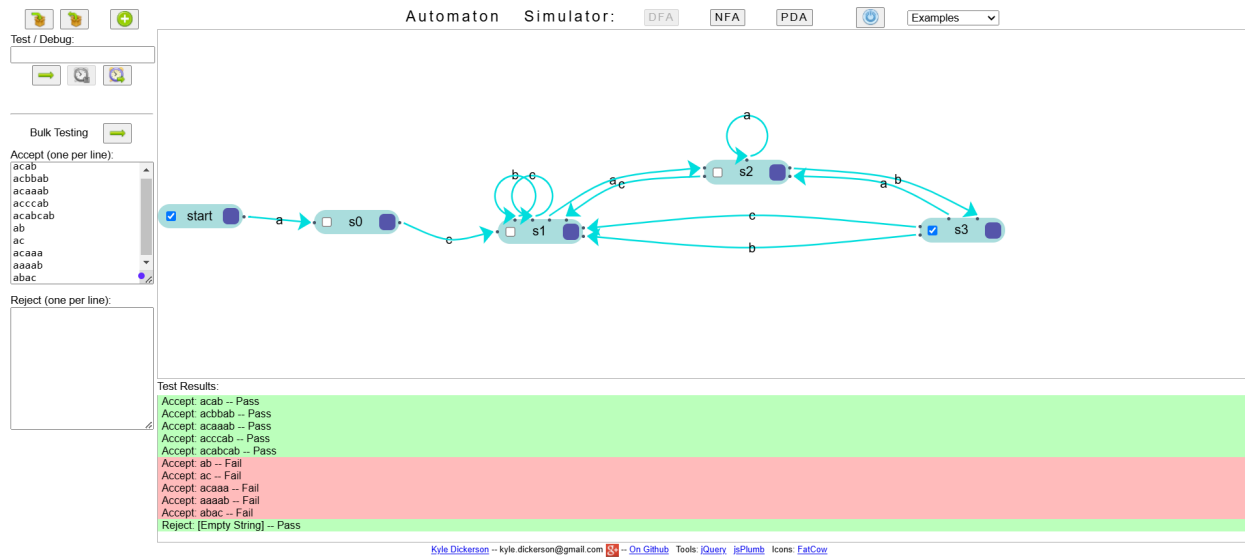


Figure 20: Simulador en Automaton.

6. Conclusión

El estudio de los Autómatas Finitos Deterministas (AFD) y No Deterministas (AFND) es crucial para el entendimiento de los lenguajes formales y su aplicación en el diseño de compiladores y sistemas de procesamiento de texto. Estos modelos teóricos nos permiten representar y reconocer lenguajes regulares, que son la base de muchos sistemas computacionales modernos. Aunque los AFD y AFND son conceptualmente similares, sus diferencias en cuanto a determinismo y la forma en que procesan las cadenas hacen que cada uno tenga ventajas y desventajas dependiendo del contexto y la aplicación. Además, la habilidad de convertir entre un AFND y un AFD amplía la flexibilidad en el diseño de autómatas, permitiendo optimizar el proceso de análisis y ejecución en compiladores y otros sistemas de procesamiento de lenguajes formales.

7. Referencias Bibliográficas

References

- [1] Codemath. (2023, 28 noviembre). *Lenguajes Formales desde CERO — Palabra, Alfabeto y Clausura de Kleene [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=UdVL-84rXc>
- [2] Codemath. (2023, 4 diciembre). *Operaciones con Palabras — Lenguajes Formales II [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=MXDl4TsEZ0>
- [3] Codemath. (2023b, diciembre 15). *Operaciones con Lenguajes y Aplicaciones — Lenguajes Formales III [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=uU-fNuwbmZg>
- [4] Codemath. (2024, 29 enero). *Descubre los autómatas: el corazón de la computación [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=pMIwci0kMv0>
- [5] Codemath. (2024b, febrero 4). *Qué es un Autómata Finito Determinista (AFD) [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=d9aEE-uLmNE>
- [6] Codemath. (2024c, abril 23). *Qué es un Autómata Finito No Determinista (AFND) [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=dIgKBNUagIE>
- [7] Codemath. (2024d, abril 29). *Convertir un Autómata NO Determinista (AFND) a Determinista (AFD) [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=hzJ8CNdPElc>
- [8] Codemath. (2024e, mayo 5). *Qué es un Autómata con Transiciones Epsilon [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=71P3daDZWlQ>
- [9] Codemath. (2024f, mayo 11). *Convertir un AFND con Transiciones a un AFND [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=1yKBT8gWN-Y>
- [10] Codemath. (2024g, mayo 27). *Pattern Matching con Autómatas: Mejora tus Algoritmos [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=22XqyZLhKPg>
- [11] Codemath. (2024h, junio 22). *Clases de Equivalencia en Autómatas y Lenguajes Formales [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=JuTuMe8Q58c>
- [12] Codemath. (2024i, julio 1). *Demostrar que un Lenguaje es Regular - Teorema de Myhill-Nerode [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=gYOvIrljRBwg>
- [13] Codemath. (2024j, julio 8). *Demostrar que un Lenguaje NO es Regular - Teorema de Myhill-Nerode [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=FPWPcQ20g0o>
- [14] Codemath. (2025, 13 febrero). *Minimización de estados de un autómata explicada desde cero [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=gd6uyNXsqcw>