



**Escuela Superior  
de Ingeniería y Tecnología**  
Universidad de La Laguna

# **Práctica 5.**

## **Autómatas Finitos en JFLAP**

Fabrizio Daniell Perilli Martín (PE203)

Computabilidad y Algoritmia

C/ Padre Herrera s/n  
38207 La Laguna  
Santa Cruz de Tenerife. España

T: 900 43 25 26

[ull.es](http://ull.es)



# Índice

Introducción	2
Ejercicios propuestos	3
- Diseño de DFAs	3
Ejercicio 1	3
Ejercicio 2	4
Ejercicio 3	5
Ejercicio 4	6
- Diseño de NFAs	8
Ejercicio 1	8
Ejercicio 2	10
Referencias	12



## Introducción

El informe a continuación tiene como finalidad introducir los fundamentos básicos de los autómatas finitos. Se comprobará y verificará el funcionamiento de algunos ejemplos y se diseñarán autómatas finitos que reconozcan ciertos lenguajes especificados en el enunciado. Para simular el comportamiento de los autómatas diseñados utilizaremos la herramienta JFLAP.

JFLAP es una herramienta interactiva para la visualización y simulación de autómatas finitos y máquinas de Turing, entre otros.



## Ejercicios propuestos

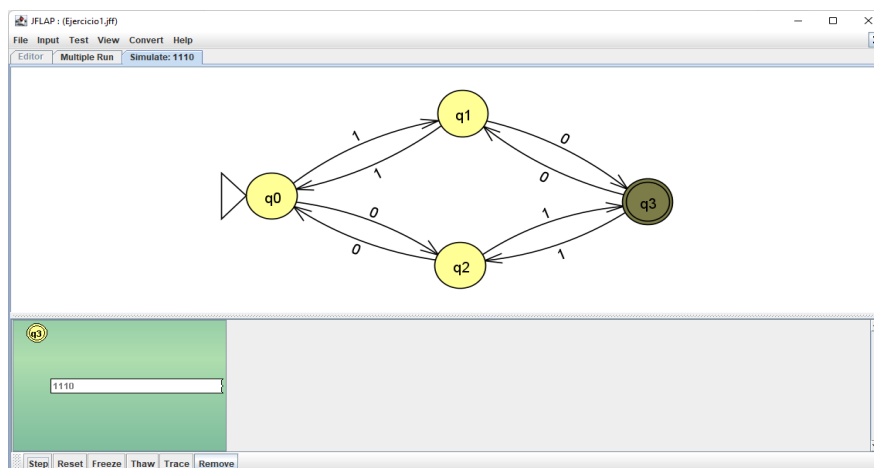
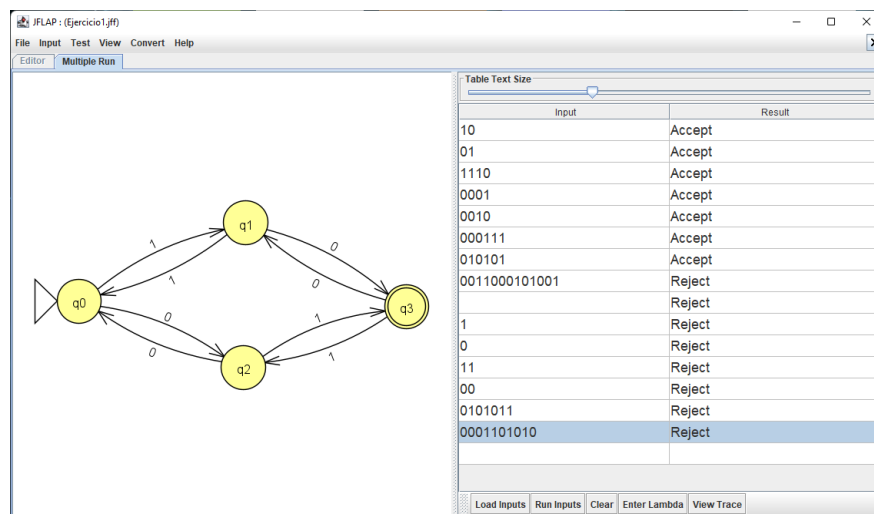
### - Diseño de DFAs

#### Ejercicio 1

Diseñar un autómata finito determinista que reconozca cadenas binarias que contengan un número impar de unos y un número impar de ceros.

Para realizar el autómata hay que tomar en cuenta las posibles combinaciones que se rechazan: par-par, par-impar, impar-par, y la que se acepta impar-impar. Se necesitan cuatro estados para evaluar las posibles combinaciones.

$$\Sigma = \{0,1\}; \quad Q = \{q_0, q_1, q_2, q_3\}; \quad F = \{q_3\}; \quad q_0 = q_0$$



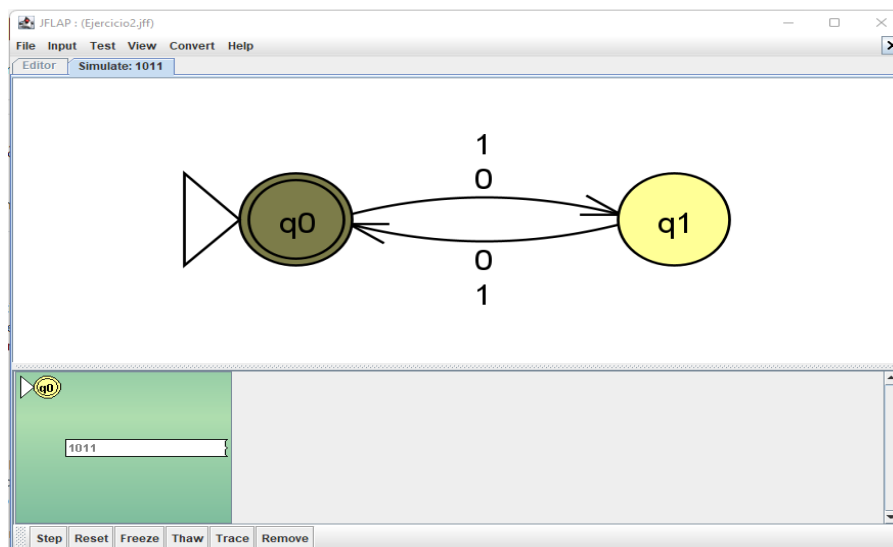
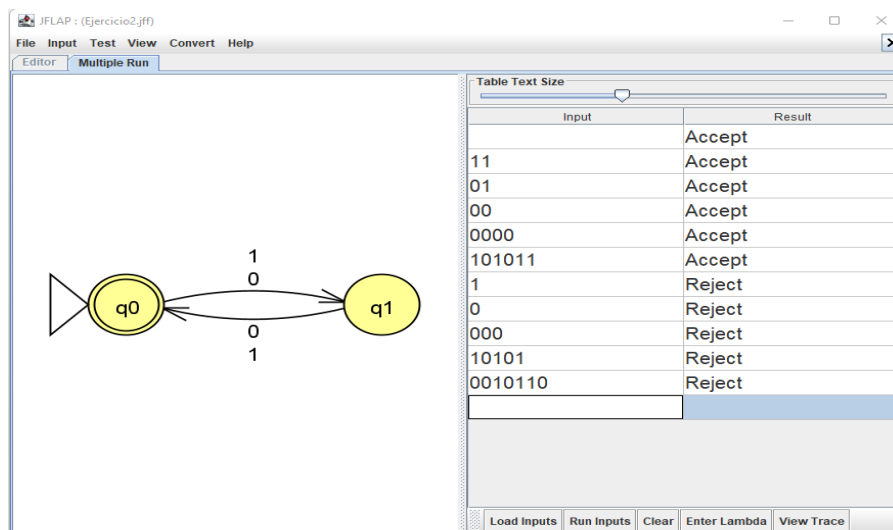


## Ejercicio 2

**Diseñar un autómata finito determinista que reconozca cadenas binarias de longitud par**

Para realizar el autómata se necesitan dos estados, el estado de arranque y un estado adicional. El estado de arranque también es el estado de aceptación ya que la cadena con longitud cero es par. Se necesitan dos transiciones hacia otro estado para rechazar los impares y otras dos desde q1 a q0 cuando vuelva otro símbolo que provoque que la cadena sea par.

$$\Sigma = \{0,1\}; \quad Q = \{q_0, q_1\}; \quad F = \{q_0\}; \quad q_0 = q_0$$







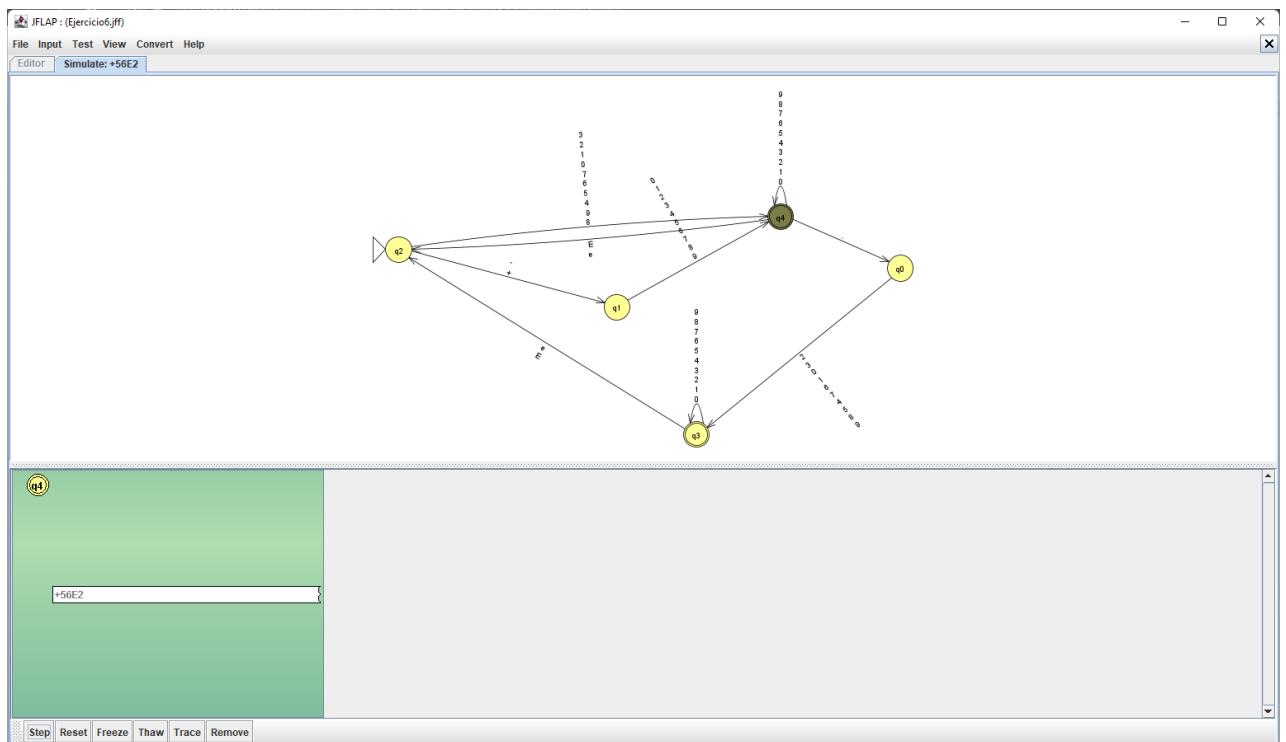
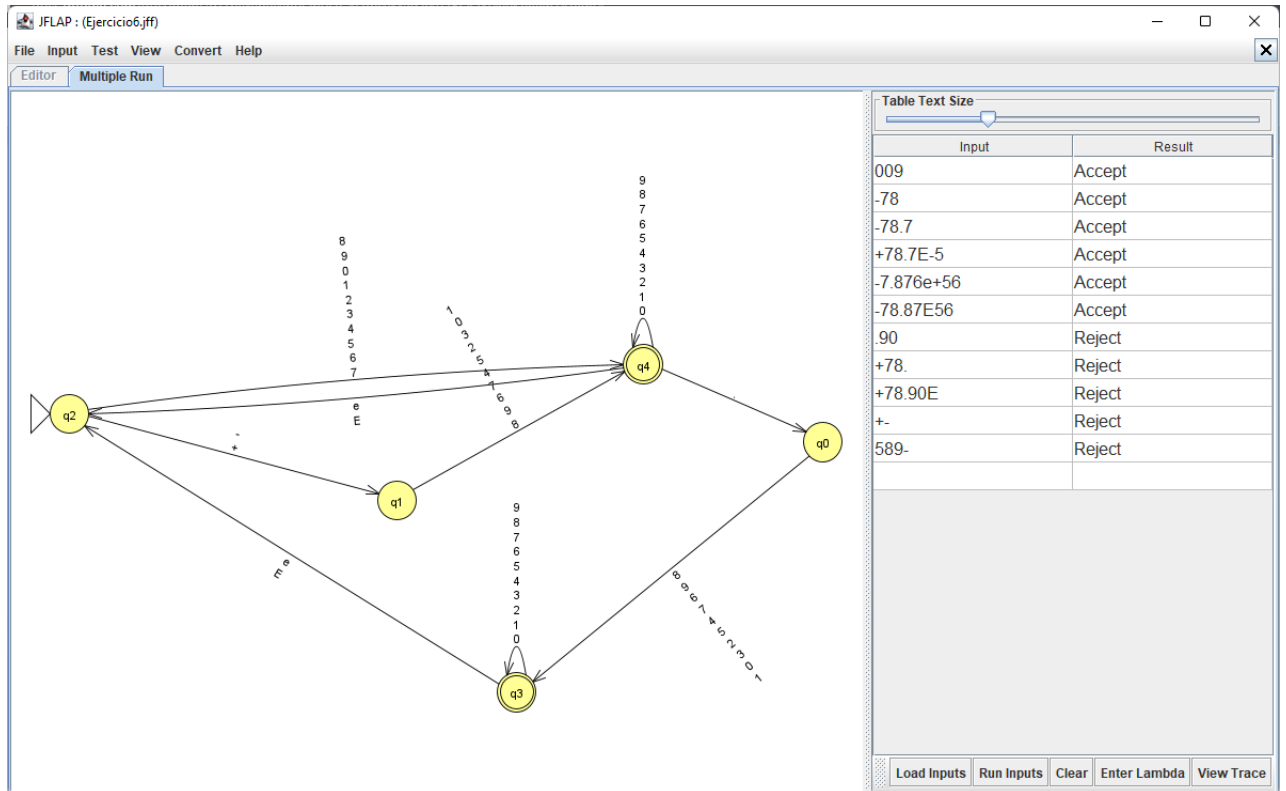
## Ejercicio 4

Diseñar un autómata finito determinista que acepte números reales. El alfabeto que usa el autómata se define como  $\Sigma = \{+, -, ., E, e, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$  y las cadenas a aceptar se definen de la siguiente forma:

- La cadena comienza opcionalmente por el símbolo “+” o “-”
- A continuación la cadena contiene uno o varios símbolos en el rango [0- 9]
- Posteriormente, y de forma opcional, aparece en la cadena el símbolo “.”. Si aparece este símbolo, la cadena debe continuar con uno o más símbolos entre el rango [0 - 9].
- Opcionalmente la cadena puede ir seguida del símbolo “E” o “e” para indicar un número en notación científica. En este caso, tendrá que ir precedido de un símbolo “+” o “-” (opcional) y una cadena de uno o más símbolos entre el rango [0-9] que representan el exponente.
- Algunas cadenas aceptadas por este autómata serán: 009, -78, -78.7, +78.7E-5, -7.876e+56, -78.87E56, etc.
- Algunas cadenas no aceptadas por este autómata serán: .90, +78., +78.90E,etc.

Para diseñar el autómata se tienen que tomar en cuenta las posibles combinaciones de aceptación de los números reales y gestionar los casos de rechazo por el autómata, se necesitarán al menos 5 estados para evaluar toda la casuística.

$\Sigma = \{+, -, ., E, e, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ ;  $Q = \{q_0, q_1, q_2, q_3, q_4\}$ ;  
 $F = \{q_3, q_4\}$ ;  $q_0 = q_2$







## - Diseño de NFAs

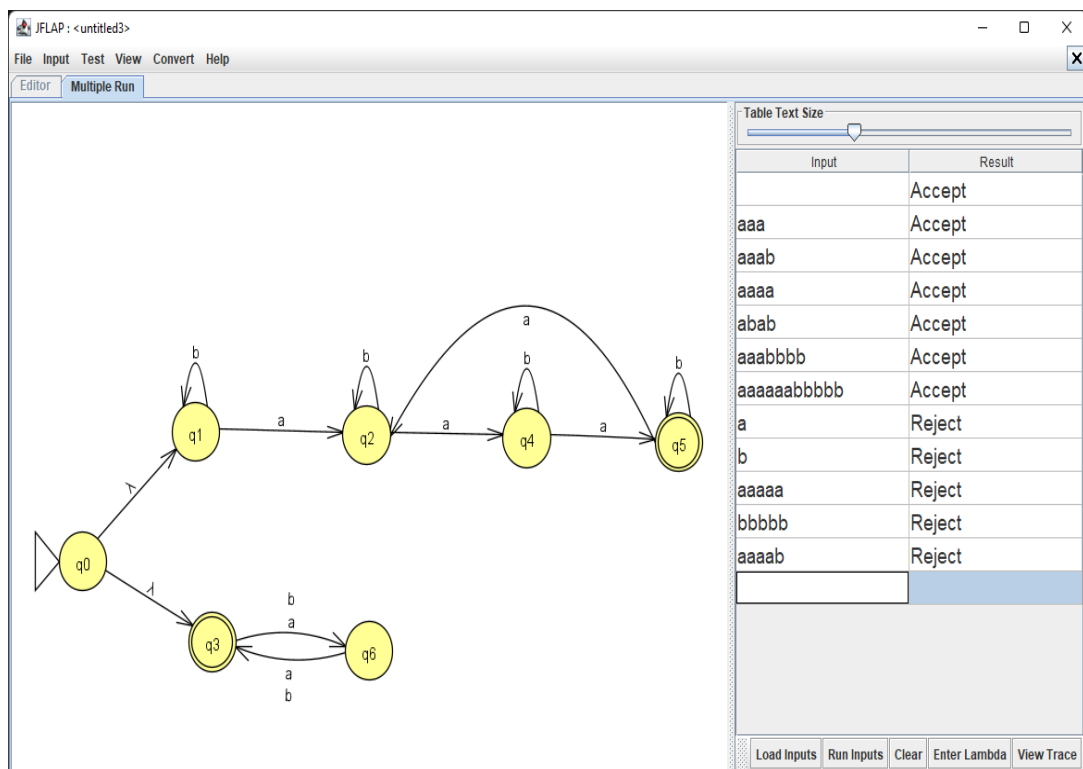
### Ejercicio 1

**Diseñar un autómata finito no determinista que reconozca cadenas sobre el alfabeto  $\Sigma = \{a, b\}$  que tengan un número de a 'es múltiplo de tres o longitud par. A partir del NFA diseñado, obtenga un DFA mínimo equivalente.**

Para realizar el diseño del autómata se tiene que tomar en cuenta dos casos de aceptación separados, por tanto se pretende hacer uso de un NFA con epsilon transiciones para evaluar los casos de aceptación que al menos se cumplan una vez.

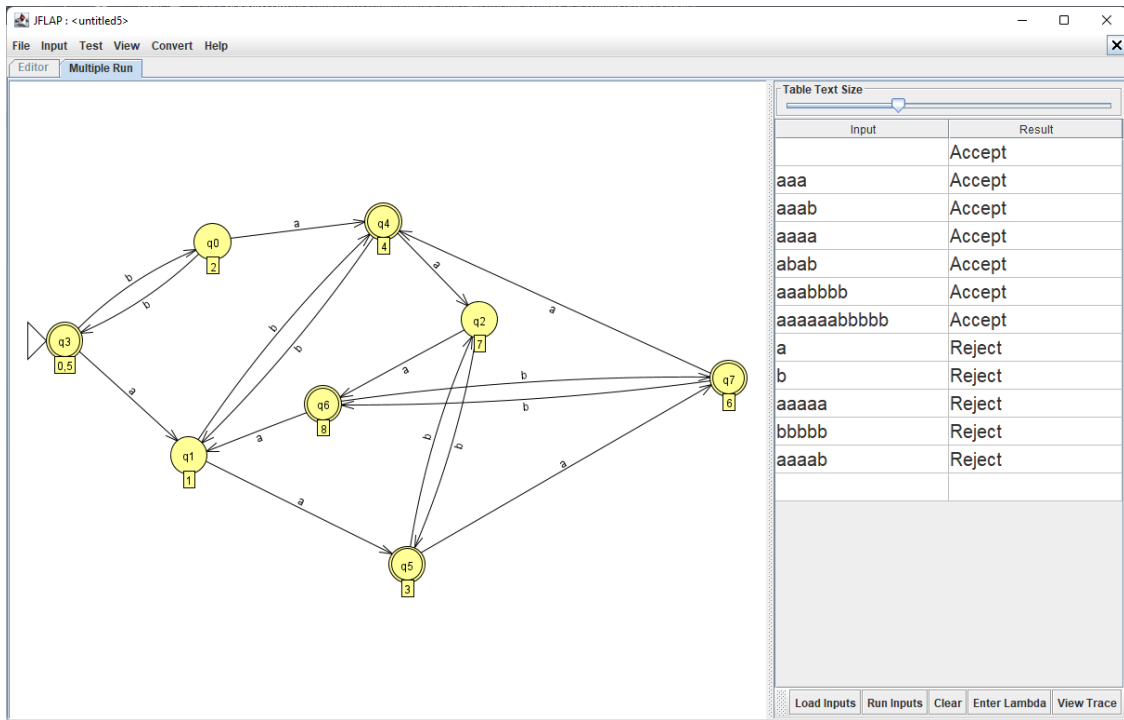
$\Sigma = \{a,b\}$ ;  $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$ ;  $F = \{q_3, q_5\}$ ;  $q_0 = q_0$

### NFA





## DFA (mínimo)





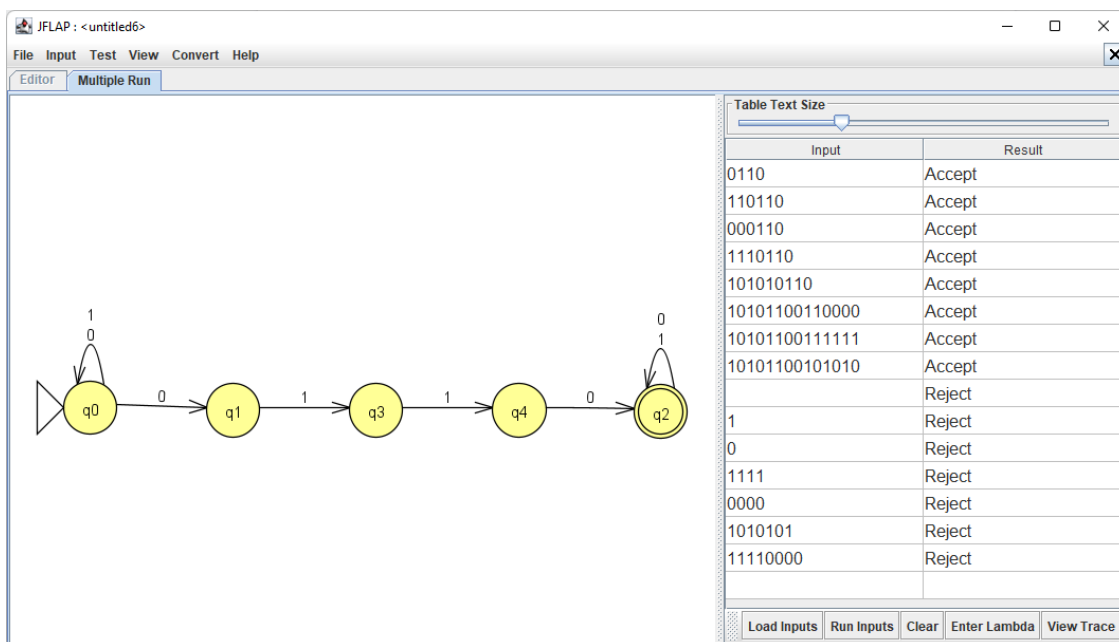
## Ejercicio 2

**Diseñar un autómata finito no determinista que reconozca cadenas sobre el alfabeto  $\Sigma = \{0, 1\}$  tales que contengan la subcadena 0110. A partir del NFA diseñado, obtenga un DFA mínimo equivalente**

En este ejercicio se busca inicialmente el caso de aceptación donde aparezca la cadena 0110, a diferencia del ejercicio anterior no se hará uso de un NFA con epsilon transiciones.

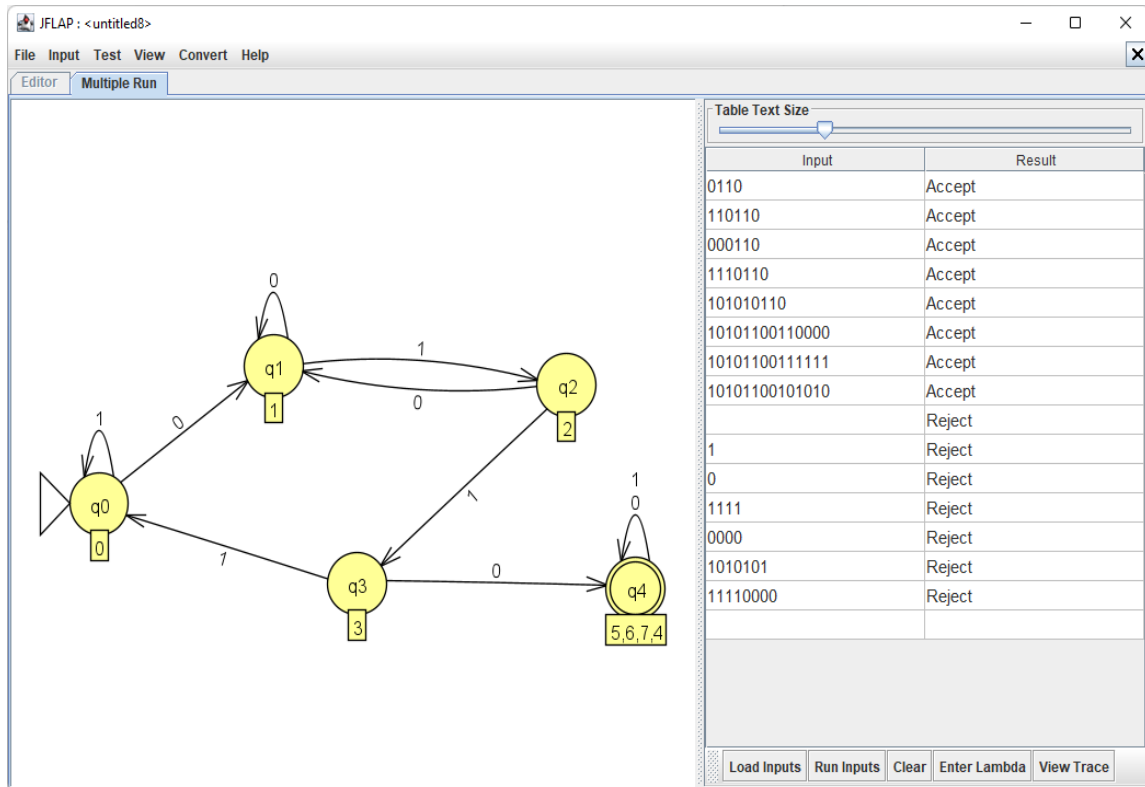
$$\Sigma = \{0,1\}; \quad Q = \{q_0, q_1, q_2, q_3, q_4\}; \quad F = \{q_2\}; \quad q_0 = q_0$$

### NFA





## DFA (Mínimo)





## Referencias

- <https://www.jflap.org>
- <https://www.youtube.com/watch?v=EzWxFMfqeFs>
- <https://www.youtube.com/watch?v=audjcmz7ons>