

Practica 2

Marina Gonzalez Torres

Ejercicios

Ejercicio 1

Consideremos el lenguaje sobre el alfabeto $\{a,b\}$ que sólo contiene a la cadena a.

- Construye un AFD que reconozca este lenguaje y que rechace todas aquellas cadenas que no pertenezcan al lenguaje.
- Prueba el autómata que has creado mediante la introducción de 6 cadenas.

Solución:

Construcción del lenguaje:

$$K = \{q_0, q_1, q_2\}$$

$$\Sigma = \{a, b\}$$

$$s = q_0$$

$$F = \{q_1\}$$

$$\Delta = \{(q_0, a, q_1), (q_0, b, q_2), (q_1, a, q_1), (q_1, b, q_2), (q_2, a, q_2), (q_2, b, q_2)\}$$

La prueba en JFLAP quedaría de la siguiente forma:

The screenshot shows the JFLAP interface with a Finite Automaton (FA) diagram on the left and a test table on the right.

FA Diagram: The diagram shows three states: q_0 (start state), q_1 (final state), and q_2 . Transitions are: $q_0 \xrightarrow{a} q_1$, $q_0 \xrightarrow{b} q_2$, $q_1 \xrightarrow{a} q_1$, $q_1 \xrightarrow{b} q_2$, $q_2 \xrightarrow{a} q_2$, and $q_2 \xrightarrow{b} q_2$.

Test Table:

Input	Result
bbaaba	Reject
aaba	Reject
aaaa	Accept
babaaa	Reject
aaaaaaaa	Accept
aaaaaaab	Reject

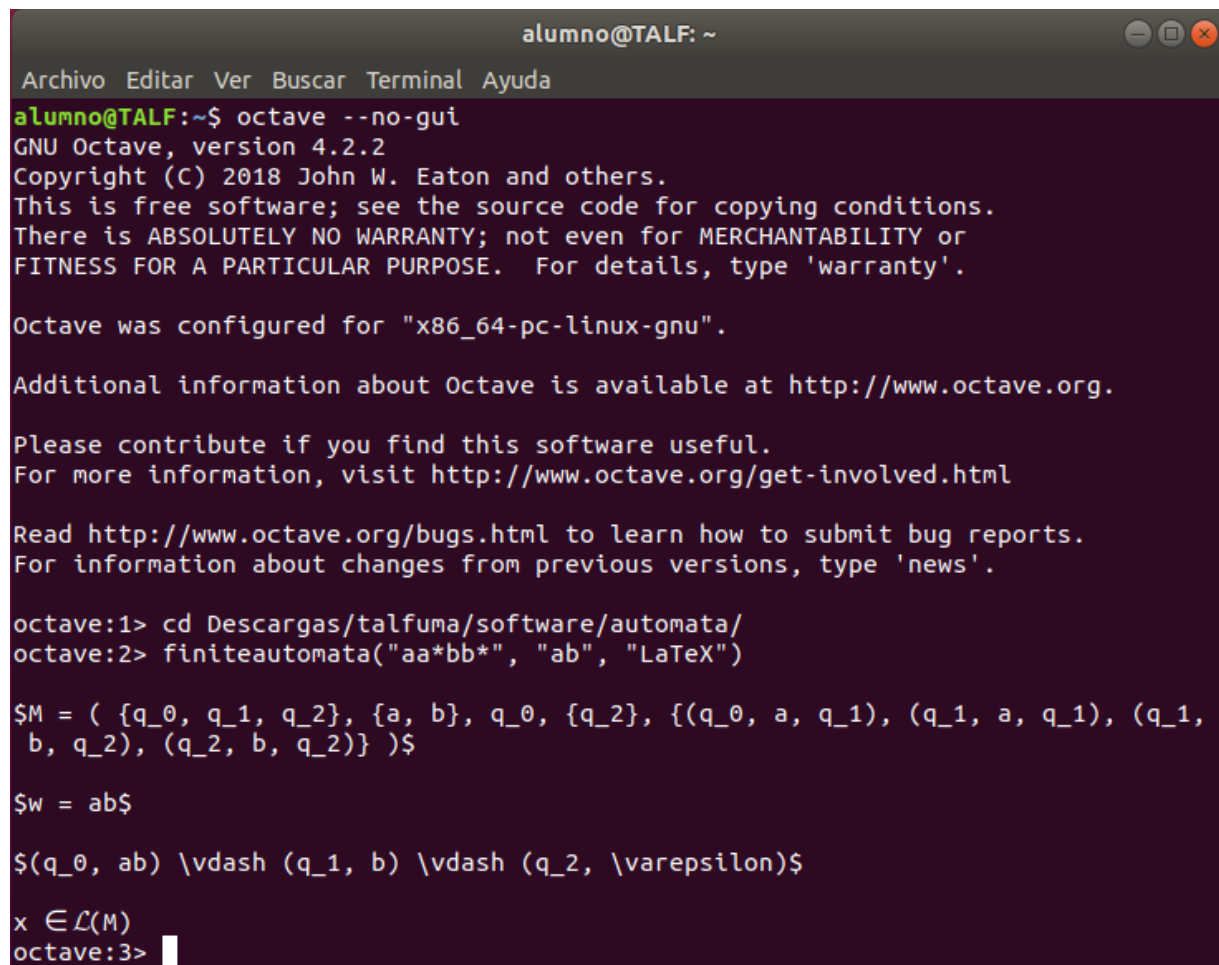
Ejercicio 2

Automata finito en Octave:

- Abre el script de Octave *finiteautomata.m* y pruebalo con el ejemplo dado en el repositorio GitHub.
- Especifica en *finiteautomata.json* el autómata creado en la actividad 1 y pruebalo con el script.

Solución:

Probar el automata de GitHub:



```
alumno@TALF: ~
Archivo Editar Ver Buscar Terminal Ayuda
alumno@TALF:~$ octave --no-gui
GNU Octave, version 4.2.2
Copyright (C) 2018 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-pc-linux-gnu".

Additional information about Octave is available at http://www.octave.org.

Please contribute if you find this software useful.
For more information, visit http://www.octave.org/get-involved.html

Read http://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.

octave:1> cd Descargas/talfuma/software/automata/
octave:2> finiteautomata("aa*bb*", "ab", "LaTeX")

$M = ( {q_0, q_1, q_2}, {a, b}, q_0, {q_2}, {(q_0, a, q_1), (q_1, a, q_1), (q_1,
b, q_2), (q_2, b, q_2)} )$

$w = ab$

$(q_0, ab) \vdash (q_1, b) \vdash (q_2, \varepsilon)$

 $x \in \mathcal{L}(M)$ 
octave:3> 
```

Especificar en el json el automata del ejercicio 1:

```
{
  "name" : "aa*",
  "representation" : {
    "K" : ["q0", "q1", "q2"],
    "A" : ["a", "b"],
    "s" : "q0",
    "F" : ["q1"],
    "t" : [
      ["q0", "a", "q1"],
      ["q0", "b", "q2"],
      ["q1", "a", "q1"],
      ["q1", "b", "q2"],
      ["q2", "a", "q2"],
      ["q2", "b", "q2"]
    ]
  }
}
```

Prueba:

```
alumno@TALF: ~
Archivo Editar Ver Buscar Terminal Ayuda
octave:4> finiteautomata("aa*", "ab", "LaTeX")

SM = ( {q_0, q_1, q_2}, {a, b}, q_0, {q_1}, {(q_0, a, q_1), (q_0, b, q_2), (q_1, a, q_1),
(q_1, b, q_2), (q_2, b, q_2), (q_2, a, q_2)} )$

Sw = ab$

$(q_0, ab) \vdash (q_1, b) \vdash (q_2, \varepsilon)$

 $x \notin \mathcal{L}(M)$ 
octave:5> finiteautomata("aa*", "aaaa", "LaTeX")

SM = ( {q_0, q_1, q_2}, {a, b}, q_0, {q_1}, {(q_0, a, q_1), (q_0, b, q_2), (q_1, a, q_1),
(q_1, b, q_2), (q_2, b, q_2), (q_2, a, q_2)} )$

Sw = aaaa$

$(q_0, aaaa) \vdash (q_1, aaa) \vdash (q_1, aa) \vdash (q_1, a) \vdash (q_1, \varepsilon)$

 $x \in \mathcal{L}(M)$ 
octave:6>
```