

# Práctica 2

Araceli Ruiz Vallecillo

24 de octubre de 2022

## Ejercicio 1

En este primer ejercicio vamos a construir un DFA que represente al lenguaje que solo contiene el **string** 'a' sobre el alfabeto  $\{a,b\}$ .

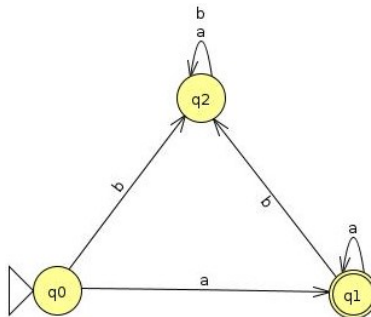
Sabemos que un autómata determinista es una 5-tupla  $(K, \Sigma, \delta, s, F)$

En este caso concreto nuestra 5-tupla se quedaría así  $(\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \{q_1\})$

Nuestra **función de transición** ( $\delta$ ) sería la siguiente:

$\delta(q, \sigma)$	a	b
$q_0$	$q_1$	$q_2$
$q_1$	$q_1$	$q_2$
$q_2$	$q_2$	$q_2$

Una vez conocidos todos los parámetros necesarios para construir nuestro autómata determinista vamos a usar el programa **JFLAP** para construir este:



Podemos comprobar, con la siguiente prueba, que el autómata generado es correcto:

Table Text Size	
Input	Result
a	Accept
aaa	Accept
abbbbbaaa	Reject
bbbaba	Reject
bbb	Reject
aaaaaaaaaaaa	Accept
	Reject

## Ejercicio 2

Para este ejercicio vamos a usar **Octave**, vamos a coger el autómata generado en el ejercicio anterior, poner su configuración y comprobar si es correcto. En primer lugar metemos su configuración en el archivo **finiteautomata.json**, como podemos ver en esta imagen:

```
{
  "name" : "Ejercicio_2",
  "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"]
    ]
  }
}
```

Una vez introducido, comprobamos nuestro autómata en **Octave** y podemos ver que nuestro autómata nos dice correctamente si las cadenas que le introducimos pertenecen al lenguaje o no. A continuación podemos ver una demostración:

```
octave:> finiteautomata("Ejercicio_2", "ab", "LaTeX")
warning: strmatch is obsolete; use strncmp or stricmp instead
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, a, q_2), (q_2, b, q_2)} )S
Sw = abS
S(q_0, ab) \vdash (q_1, b) \vdash (q_2, \vdash \epsilon)S
x \notin L(N)
octave:> finiteautomata("Ejercicio_2", "aaaaaaaa", "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, a, q_2), (q_2, b, q_2)} )S
Sw = aaaaaaaaaS
S(q_0, aaaaaaaaa) \vdash (q_1, aaaaaaaaa) \vdash (q_1, aaaaaa) \vdash (q_1, aaaaa) \vdash (q_1, aaaa) \vdash (q_1, aaa) \vdash (q_1, aa) \vdash (q_1, a) \vdash (q_1, \vdash \epsilon)S
x \notin L(N)
octave:> finiteautomata("Ejercicio_2", "ababababab", "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, a, q_2), (q_2, b, q_2)} )S
Sw = abababababS
S(q_0, ababababab) \vdash (q_1, babababab) \vdash (q_2, abababab) \vdash (q_2, bababab) \vdash (q_2, ababab) \vdash (q_2, babab) \vdash (q_2, abab) \vdash (q_2, bab) \vdash (q_2, ab) \vdash (q_2, b) \vdash (q_2, \vdash \epsilon)S
x \notin L(N)
octave:> q[]
```