

Practica 2

Por Ruben Cazorla Rodriguez

October 29, 2022

Lenguaje

Dado un lenguaje $L=\{a,b\}$ que solo contenga “a”

1 Ejercicio 1

1.1 Enunciado

Construye un DFA que reconozca el lenguaje y rechace las cadenas que no cumplan con la condicion, y prueba el automata con 6 cadenas.

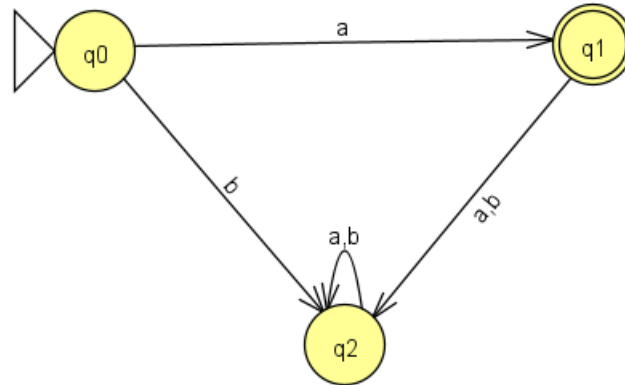
1.2 Solucion

1.2.1 Construccion

Para crear el DFA hay que definir $(K, \Sigma, \delta, s, F)$ donde:

- $K=\{q_0, q_1, q_2\}$, el conjunto de estados no vacios
- $\Sigma=\{a,b\}$, el alfabeto
- $\delta=\{(q_0, a, q_1), (q_0, b, q_2), (q_1, a, q_2), (q_1, b, q_2), (q_2, a, q_2), (q_2, b, q_2)\}$, funciones de transicion
- $s=q_0$, el estado inicial
- $F=\{q_1\}$, el conjunto de estados iniciales

Una vez definido el DFA, utilizando JFLAP, podemos esquematizarlo graficamente:



1.2.2 Pruebas

Utilizando la opción “multiple run” podemos probar las 6 cadenas para probar si el automata funciona correctamente:

- a : acepta
- aaaa: rechaza
- ab: rechaza
- bbbaaa: rechaza
- b: rechaza
- baba: rechaza

2 Ejercicio 2

2.1 Enunciado

Abre en Octave el archivo “finiteautomata.m”, Especifica en “finiteautomata.json” el lenguaje del ejercicio 1 y prueba que funciona correctamente.

2.2 Solucion

Primero abrimos “finiteautomata.json” y definimos el lenguaje en un json.

```

{
  "name" : "practica2",
  "representation" : {
    "K" : ["q0", "q1", "q2"],

```

```

"A" : ["a","b"],
"s" : "q0",
"F" : ["q1"],
"t" : [["q0", "a", "q1"],
["q0", "b", "q2"],
["q1", "a", "q2"],
["q1", "b", "q2"],
["q2", "a", "q2"],
["q2", "a", "q2"]] }
}

```

Y una vez definido, abrimos “*finiteautomata.m*” en Octave, y utilizando el comando *finiteautomata(“nombre_automata”, “String”)* mostrara el camino que sigue el automata analizando la cadena introducida y mostrara si pertenece al lenguaje mediante una variable llamada “ans”, la cual sera 1 si el lenguaje acepta y 0 si lo rechaza.

Sabiendo esto, y utilizando los ejemplos del ejercicio anterior, sus resultados son los mismos que en el anterior ejercicio:

- a : acepta
- aaaa: rechaza
- ab: rechaza
- bbbaaa: rechaza
- b: rechaza
- baba: rechaza