Definiciones Preliminares:

Def: Un alfabeto es un conjunto finito de símbolos

Ejemplo: $\Sigma = \{0, 1\}$

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

Def: Una cadena (string) sobre el alfabeto Σ sucesión finita de símbolos del alfabeto

Ejemplo: $w_1 = 001110$ string sobre Σ

$w_2 = bbbb$ string sobre Λ

longitud $6 = |w_1|$

longitud $4 = |w_2|$

Hay una única cadena de longitud cero, llamada la palabra vacía ε.

Si s_1, s_2 son cadenas sobre el mismo alfabeto Σ definimos su concatenación $s_1 s_2$

Ejemplo: $\Lambda = \{a, b\}$ $s_1 = aaa$, $s_2 = aba$

$s_1 s_2 = aaaaba \neq s_2 s_1 = abaaaa$

$s_1 \varepsilon = \varepsilon s_1 = s_1 \leftarrow$ ε es como el cero.

Para un alfabeto Σ definimos

$$\Sigma^* = \{ \text{cadenas sobre } \Sigma \text{ de cualquier longitud (finita)} \}$$

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

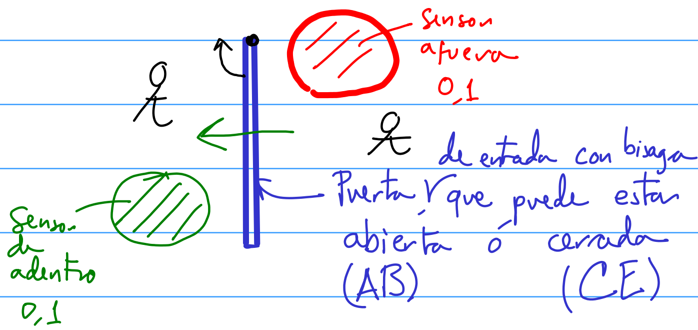
$$\Sigma^* = \{ \varepsilon, a, b, aa, ab, bb, ba, aaa, \dots \}$$

conjunto infinito salvo si $\Sigma = \emptyset$.

1) Autómatas Finitos deterministas (AFDs)

Son modelos de un sistema de cómputo con una memoria muy limitada. Vamos a ilustr lo que son mediante un

Ejemplo: El controlador de una puerta automática



De acuerdo a la información de los sensores (adentro, afuera) los inputs posibles para la máquina son

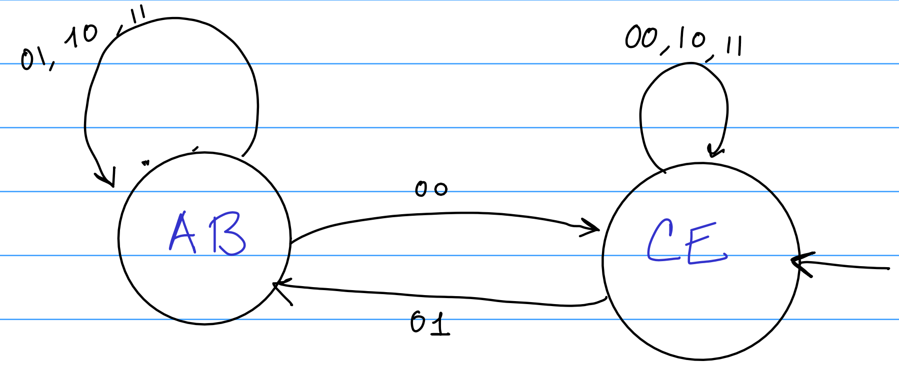
00	01	10	11
NINGUNA	AFUERA	ADENTRO	AMBAS

Cómo debe evolucionar el estado de la puerta según su input?

	00	01	10	11
AB	CE	AB	AB	AB
CE	CE	AB	CE	CE

Si la puerta esta CE y hay gente adentro y afuera NO abro para entrar golpear a la gente de adentro!

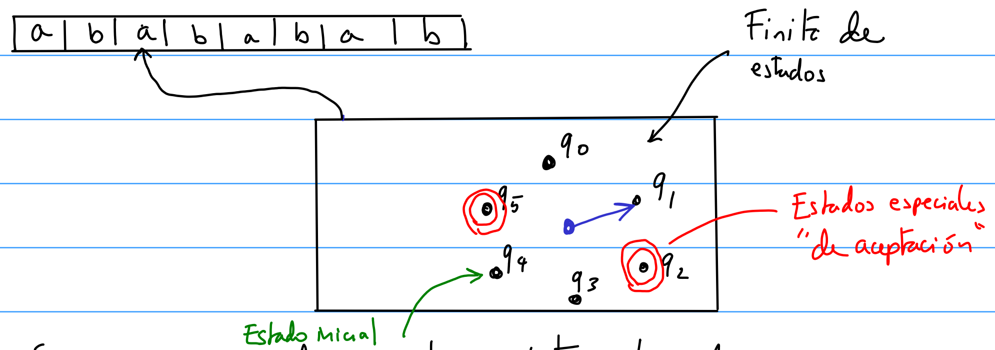
Esta información también se puede dibujar en un "diagrama de estados"



[Máquina]

Def: Un automata finito recibe como input una cadena de caracteres en una cinta y como único output produce 1 (ACEPTO) o 0 (RECHAZO) de la cadena.
El cálculo que realiza para llegar a esa conclusión utiliza solo una cantidad finita (bits) de memoria fijados de antemano.

Más precisamente



La máquina va avanzando por la cinta, leyendo un símbolo a cada paso cuando lee el símbolo pasa a un nuevo estado que depende sólamente del estado actual y del símbolo que acaba de leer. (de manera DETERMINISTA)

Al terminar la lectura de la cinta, el automata "acepta" el string si y solo si el estado que tiene al finalizar la palabra es "de aceptación" (marcados con doble círculo).

De manera precisa un autómata finito es una 5-tupla $(K, \Sigma, \delta, s, F)$ donde:

(1) K es un conjunto finito, la colección de estados de la máquina

(2) Σ es un conjunto finito, el alfabeto

(3) δ es la función de transición (determinista), que a cada $q \in K$, $a_i \in \Sigma$ asigna

$$\delta(q, a) \in K$$

"Estado siguiente cuando leo el símbolo $a \in \Sigma$ mientras estoy en estado q"

$$\delta : K \times \Sigma \longrightarrow K$$

(4) $s \in K$ es el estado inicial

(5) $F \subseteq K$ es el conjunto dado de estados de aceptación.

Con estos preliminares, podemos ahora definir el CÁLCULO (COMPUTATION) hecha por un autómata finito.

Def: Una CONFIGURACIÓN de un AFD es un elemento de $K \times \Sigma^*$ (ejemplo $(q_1, ababab)$)

estado — palabra en lo que queda de la cinta.

se pasa de una configuración a la otra en un paso

$$(q_i, w) \vdash_M (q_j, w') \iff w = aw' \text{ y } \delta(q_i, a) = q_j$$

$$(q_i, w) \vdash^* (a, z) \iff$$ se puede llegar de (q_i, w) a (a, z) mediante un número finito de pasos $\vdash^*$

IMPORTANTE PARA EL CASO ND.

Def: M acepta $w \in \Sigma^* \iff \exists f \in F : (w, s) \vdash^* (f, \varepsilon)$

Lenguaje aceptado por $M := \{w \in \Sigma^* : M \text{ acepta } w\}$

Ejemplo: Qué hace esta máquina N:

$K = \{q_0, q_1\}$ $\quad s = q_0$

$\Sigma = \{a, b\}$ $\quad F = \{q_0\}$

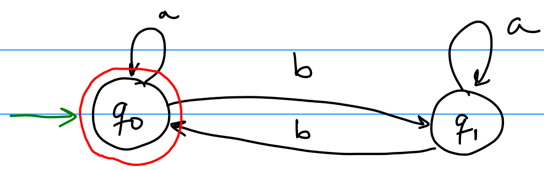
δ	a	b
q_0	q_0	q_1
q_1	q_1	q_0

Lema: El lenguaje aceptado por N son las palabras con un número par de b's.

Dem: $(s, w) \vdash^* \begin{cases} (q_0, \varepsilon), & \text{si } w \text{ tiene una cantidad par de } b\text{'s} \\ (q_1, \varepsilon), & \text{si } w \text{ tiene una cantidad impar de } b\text{'s} \end{cases}$

demostrar por inducción en longitud de w.

Diagrama de estados



Ejercicio: Diseñe un autómata M cuyo lenguaje $L(M) = \{w \in \{a,b\}^* : w \text{ no contiene el string } bbb\}$