

Unidad IV

Lenguajes y Autómatas

4.1. Máquinas de estado finito.

Se denomina *máquina de estados* a un modelo de comportamiento de un sistema con entradas y salidas, en donde las salidas dependen no sólo de las señales de entradas actuales sino también de las anteriores.

Las máquinas de estados se definen como un conjunto de [estados](#) que sirve de intermediario en esta relación de entradas y salidas, haciendo que el historial de señales de entrada determine, para cada instante, un estado para la máquina, de forma tal que la salida depende únicamente del estado y las entradas actuales.

Una máquina de estados se denomina *máquina de estados finitos* (*FSM* por *finite state machine*) si el conjunto de estados de la máquina es finito, este es el único tipo de máquinas de estados que podemos modelar en un computador en la actualidad; debido a esto se suelen utilizar los términos *máquina de estados* y *máquina de estados finitos* de forma intercambiable. Sin embargo un ejemplo de una *máquina de estados infinitos* sería un [computador cuántico](#) esto es debido a que los [Qubit](#) que utilizaría este tipo de computadores toma valores continuos, en contraposición los [bits](#) toman valores discretos (0 ó 1). Otro buen ejemplo de una máquina de estados infinitos es una [Máquina universal de Turing](#) la cual se puede definir teóricamente con una "cinta" o memoria infinita.

La representación de una máquina de estados se realiza mediante un [Diagrama de estados](#), sin embargo también es posible utilizar un [Diagrama de flujo](#).

Es posible clasificar las máquinas de estados en *aceptoras* o *transductoras*:

- Aceptoras (también llamadas reconocedoras o discriminadoras): Son aquellas en donde la salida es binaria (sí/no), depende únicamente del estado y existe un estado inicial. Puede decirse, entonces, que cuando la máquina produce una salida "positiva" (es decir, un "sí"), es porque ha "reconocido" o "aceptado" la secuencia de entrada. En las máquinas de estados aceptoras, los estados con salida "positiva" se denominan *estados finales*.
- Transductoras: Son las más generales, que convierten una secuencia de señales de entrada en una secuencia de salida, pudiendo ésta ser binaria o más compleja, depender de la entrada actual (no sólo del estado) y pudiendo también prescindirse de un estado inicial.

La bibliografía a veces llama [autómata finito](#) a las aceptoras, mientras que en otros casos se emplea *autómata* como sinónimo de *máquina de estados* sin importar su tipo.

Las aceptoras son los de mayor interés en la Teoría de la Computación, más precisamente en la [Teoría de autómatas](#), siendo éstas ramas de la matemática. Las transductoras, en cambio, lo son en la electrónica digital y la computación práctica. Es por eso que, por lo general, en los textos sobre matemática y ciencias de la computación se suele hablar de *autómatas* (y se refieren a las aceptoras) mientras que los de electrónica y computación práctica hablan de *máquinas de estados* (y se refieren a los transductoras).

En UML (Lenguaje Unificado de Modelado), dice que una máquina de estado es aquel comportamiento que permite hacer un seguimiento de la vida de un objeto en el transcurso de un tiempo finito.

4.2. Lenguajes regulares.

Un **lenguaje regular** es un tipo de [lenguaje formal](#) que satisface las siguientes propiedades:

Los lenguajes más sencillos que se considerarán son los lenguajes regulares, es decir, los que se pueden generar a partir de los lenguajes básicos, con la aplicación de las operaciones de unión, concatenación y * de Kleene un número finito de veces.

Puede ser reconocido por:

- un [autómata finito determinista](#)
- un [autómata finito no determinista](#)
- un [autómata de pila](#)
- un autómata finito alterno
- una [máquina de Turing](#) de solo lectura

Es generado por:

- una [gramática regular](#)
- una gramática de prefijos

Es descrito por:

- una [expresión regular](#)

4.3. Lenguajes no regulares