

**TECNOLÓGICO NACIONAL DE MEXICO**

**INSTITUTO TECNOLÓGICO DE IZTAPALAPA**

**NOMBRE: FERMIN CRUZ ERIK**

**MATRICULA: 181080007**

**GRUPO: ISC-6AM**

**MATERIA: LENGUAJES Y AUTOMATAS I**

**PROFESOR: M.C. ABIEL TOMÁS PARRA HERNÁNDEZ**

**ACTIVIDADES SEMANALES**

## SEMANA 6

### GRAMÁTICA REGULAR

La gramática regular en los autómatas se refiere a una gramática que puede ser clasificada de 2 maneras: de izquierda o de derecha. Estas gramáticas pueden dar paso a los lenguajes que de igual manera llevan su nombre: lenguajes regulares.

Se utilizan principalmente para analizar el contenido que tenemos en una cadena de caracteres por medio de patrones, estos patrones son usados para identificar una determinada combinación de estos caracteres dentro de un string o cadena de tipo texto.

En la gramática regular de derecha tenemos que  $A \rightarrow a$  entonces  $a$  es un símbolo que no termina en  $N$  y termina en  $\Sigma$ .

Mientras que en la gramática de izquierda tenemos algunas reglas, que son:

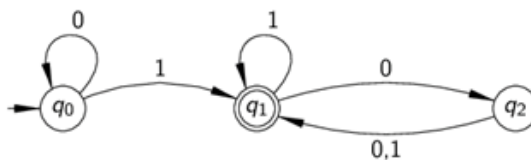
- $A \rightarrow a$ , donde  $A$  es un símbolo no-terminal en  $N$  y  $a$  uno terminal en  $\Sigma$
- $A \rightarrow Ba$ , donde  $A$  y  $B$  pertenecen a  $N$  y  $a$  pertenece a  $\Sigma$
- $A \rightarrow \epsilon$ , donde  $A$  pertenece a  $N$ .

### AUTOMATA FINITO

Se trata de un modelo computacional que realiza cálculos o instrucciones de manera automática en una entrada para producir una salida.

Este modelo está conformado por varias partes como por ejemplo un alfabeto, un conjunto de estados finitos, una función de transiciones, un estado inicial y por último, un estado final.

Recordad los autómatas finitos deterministas



- La **computación** del autómata con entrada 011 es  $(q_0, q_0, q_1, q_1)$  que me dice la secuencia de estados por los que pasa con entrada 011
- Cada entrada me da exactamente una computación. Tengo siempre como mucho **una opción** desde un estado si leo un símbolo ← Esto se llama **determinismo**

Su funcionamiento se basa en la redundancia, en una función de transición, que recibe a partir de un estado inicial una cadena de caracteres pertenecientes al alfabeto (la entrada), y que va leyendo dicha cadena a medida que el autómata se desplaza de un estado a otro, para finalmente detenerse en un estado final o de aceptación, que representa la salida.

La finalidad de los autómatas finitos es la de reconocer lenguajes regulares, que corresponden a los lenguajes formales más simples según la Jerarquía de Chomsky.

Estos tienen su origen en las máquinas electromecánicas, en las cuales el matemático ruso Andréi Márkov formalizó un proceso llamado cadena de Markov, donde la ocurrencia de cada evento dependía de una cierta probabilidad del evento anterior. Esta particularidad de recordar los estados anteriores, les sirve de mucho a los autómatas para poder tomar el siguiente estado, ya que deben conocer el anterior para saber cuál es el posterior.

## **AUTÓMATA FINITO NO DETERMINISTA**

Son autómatas que, a diferencia de los autómatas finitos deterministas, poseen al menos un estado  $q \in Q$ , tal que para un símbolo  $a \in \Sigma$  del alfabeto, existe más de una transición posible.

Estos pueden darse en cualquiera de estos dos casos:

- 1.- Que existan transiciones del tipo  $\delta(q, a) = q_1$  y  $\delta(q, a) = q_2$ , siendo  $q_1 \neq q_2$ ;
- 2.- Que existan transiciones del tipo  $\delta(q, \epsilon)$ , siendo  $q$  un estado no-final, o bien un estado final pero con transiciones hacia otros estados.

Los autómatas no deterministas pueden pasar por varios estados a la vez, generando una ramificación de las configuraciones existentes en un momento dado. Asimismo, en estos podemos aceptar la existencia de más de un nodo inicial.

Estos autómatas tienen un funcionamiento que es por decirlo así, de la siguiente manera:

La máquina comienza en el estado inicial que se le ha especificado y lee una cadena de caracteres pertenecientes al alfabeto. El autómata utiliza la función de transición de estados  $T$  para determinar el siguiente estado, usando el estado actual y el símbolo que acaba de leer o la cadena vacía. Sin embargo, "el estado siguiente de un Autómata finito no determinado no solo depende del evento de la entrada actual, sino que también depende de un número arbitrario de los eventos de entrada posterior.

Hasta que se producen estos acontecimientos posteriores no es posible determinar en qué estado se encuentra la máquina". Cuando el autómata ha terminado de leer, y pasa a un estado de aceptación, se dice que acepta la cadena, de lo contrario se dice que la cadena de caracteres es rechazada.

## AFND: Autómatas finitos *no deterministas*

Veamos un ejemplo más de AFD. Con  $\Sigma=\{0,1\}$ , queremos que acepte el lenguaje de los strings que terminan en 101.

Bosquejo (incompleto) de la solución:

