Capítulo 9.

Introducción a los lenguajes formales

Continuar

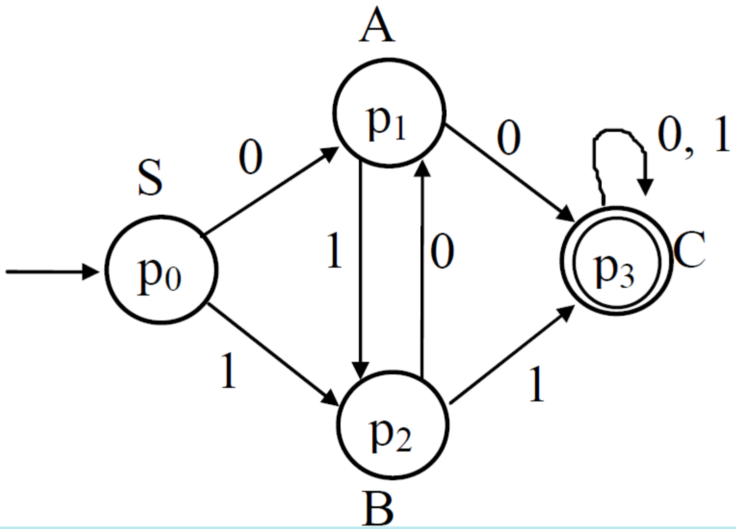
Introducción

Un lenguaje es un conjunto de símbolos y métodos para estructurar y combinar dichos símbolos. Un lenguaje también recibe el nombre del idioma y como tal consta de todos los símbolos válidos por dicho lenguaje y los métodos para estructurar correctamente cada una de las palabras, frases y oraciones. Esta clase de lenguaje recibe el nombre de lenguaje natural.

$$L(G) = \{x \in T^* | S \Rightarrow^* x\}$$

Gramáticas y lenguajes formales

Lenguaje L (G). Este tipo de lenguaje se basa en la gramática, así como en las reglas o métodos para la creación de palabras propias del lenguaje. Un lenguaje L consiste en una gramática con todos los arreglos que se pueden obtener a partir del estado inicial y las composiciones.



Estructuración de las gramáticas

Las gramáticas están integradas por varios elementos que permiten la estructuración de palabras. La gramática es el sistema que permite establecer las reglas que han de aplicarse a un lenguaje.

$$\alpha \to \beta, \ \ \alpha = \varphi A \rho \quad \text{y} \quad \beta = \varphi \omega \rho$$
$$\varphi, \omega, \rho \in (N \cup T)^* \quad \text{y} \ \ A \text{ es } S \text{ ó } A \in N$$

Clasificación de las gramáticas

Se clasifican en:

Tipo 0: Si no se pone ninguna restricción a las composiciones de G.

Tipo 1: Si para cualquier composición $d_1 \rightarrow d_2$ de la gramática G, la longitud de símbolos de la izquierda de la composición (d_1) es menor o igual a la longitud de símbolos de la derecha (d_2).

Típo 2: Si el laldo izquierdo de cada composición es un símbolo no terminal y el lado derecho consta de uno o más símbolos terminales y/o no terminales.

Tipo 3: Si el lado izquierdo de la composición es un símbolo no terminal y el lado derecho tiene uno o más símbolos, incluyendo a lo más un símbolo no terminal.

$$x \rightarrow y$$

$$x \in (NT/T)^{+}$$

$$y \in (NT/T)^{*}$$

Gramáticas regulares

Se representan principalmente por medio de autómatas finitos los cuales son elementos generadores de palabras de un lenguaje, en donde una palabra se genera comenzando en el símbolo inicial y terminando como una cadena de símbolos terminales que no contiene ningún símbolo no terminal.

- $S \rightarrow aA$
- $S \rightarrow bA$
- $A \rightarrow aB$
- $A \rightarrow bB$
- $A \rightarrow a$
- $B \rightarrow aA$
- $B \rightarrow bA$

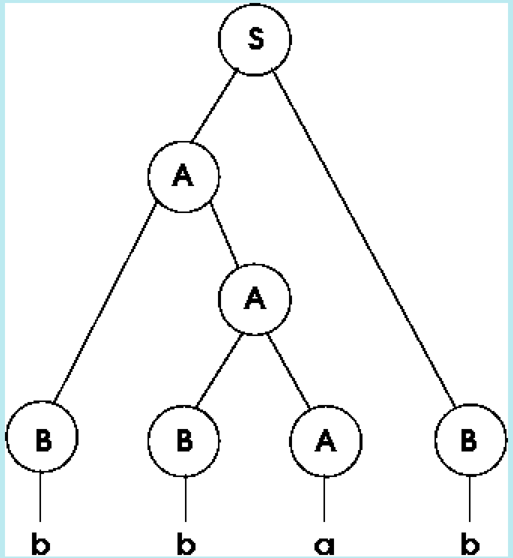
Gramáticas libres de contexto

Este tipo de gramáticas son las más usadas en la elaboración de compiladores, traductores e intérpretes, y se pueden representar por medio de árboles de derivación, representación BNF y diagramas sintácticos.

$$
\begin{aligned}
&G=(\{S, A, B, I\}, \{a, b\}, P, S)\\
&\quad P=\{\\
&\qquad S \rightarrow A \mid B\\
&\qquad A \rightarrow IaI \mid IaA\\
&\qquad B \rightarrow IbI \mid IbB\\
&\qquad I \rightarrow aIbI \mid bIaI \mid \varepsilon\\
&\qquad \}
\end{aligned}
$$

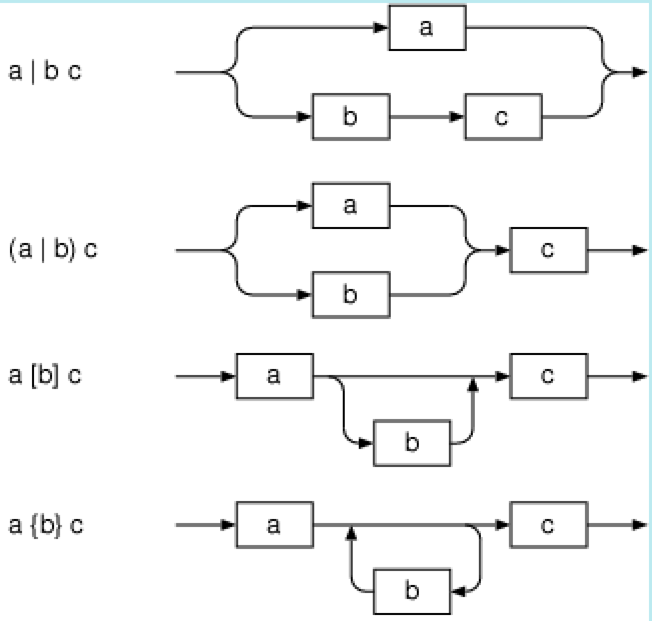
Representación mediante árboles de derivación

El procedimiento para determinar si una palabra pertenece a un lenguaje por el método de árboles de derivación es semejante al desarrollado por medio de composiciones, sólo que en este caso se estructura un árbol teniendo como raíz de ese árbol al símbolo inicial s y colocando como hijos a los signos del lado derecho de la composición.



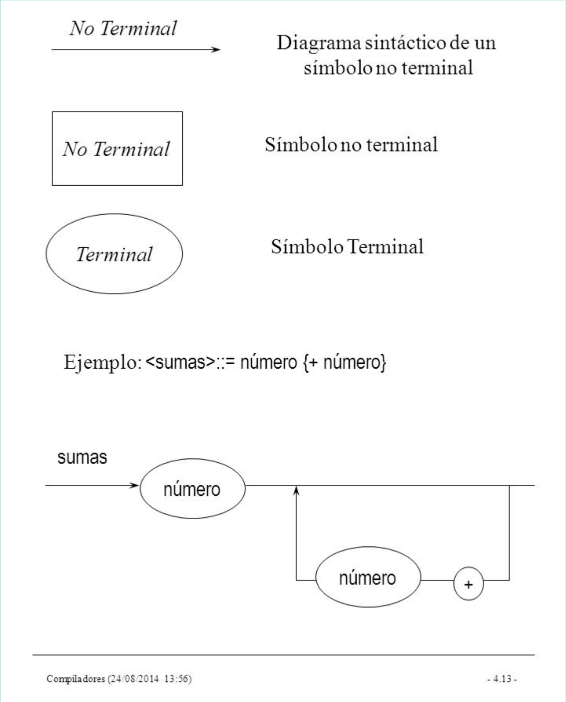
Representación BNF (Backus-Naur)

Se mencionó con anterioridad que la gramática libre de contexto se utiliza con frecuencia para la representación de lenguajes formales como C, Pascal, Basic, etc. La representación BNF es un buen ejemplos. En la gramática BNF la flecha (→) de una composición se indica con "::=".



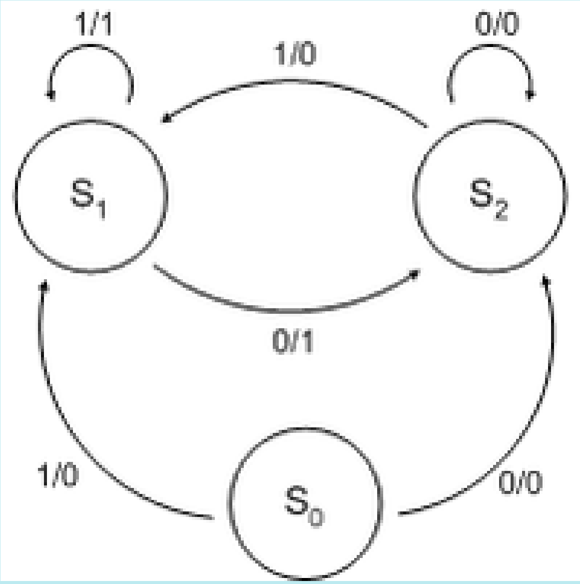
Diagramas sintácticos

Esta es una forma gráfica para representar una gramática por medio de gráficas dinámicas que permiten determinar en forma más ilustrativa si una palabra pertenece a un lenguaje; a esta gráfica se le conoce como diagrama sintáctico o diagrama de sintaxis. Toda palabra reservada de un lenguaje formal es susceptible de ser representada por medio de diagramas sintácticos.



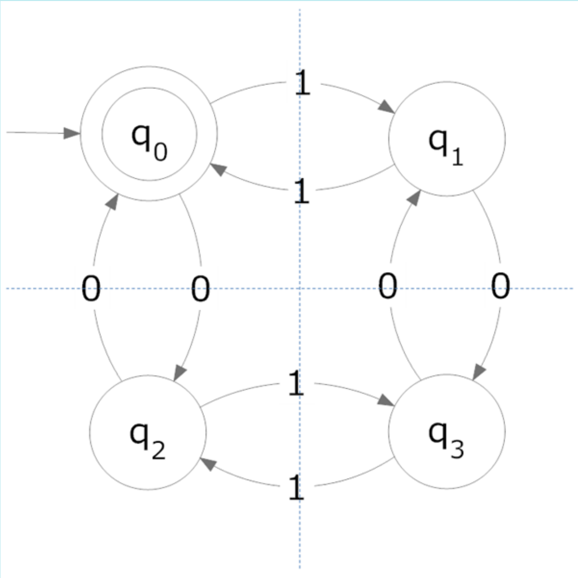
Autómatas finitos

Todo proceso que recibe una o varias entradas, que transforma estas entradas y después emite una salida recibe el nombre de sistema. Los autómatas finitos también reciben como entrada información que procesan y en función de ello emiten una salido. Un autómata finito recibe una palabra, la cual debe procesar por medio de un recorrido a través de los diferentes estados que integran el autómata.



Autómatas finitos

Todo proceso que recibe una o varias entradas, que transforma estas entradas y después emite una salida recibe el nombre de sistema. Los autómatas finitos también reciben como entrada información que procesan y en función de ello emiten una salido. Un autómata finito recibe una palabra, la cual debe procesar por medio de un recorrido a través de los diferentes estados que integran el autómata.



Terminología básica

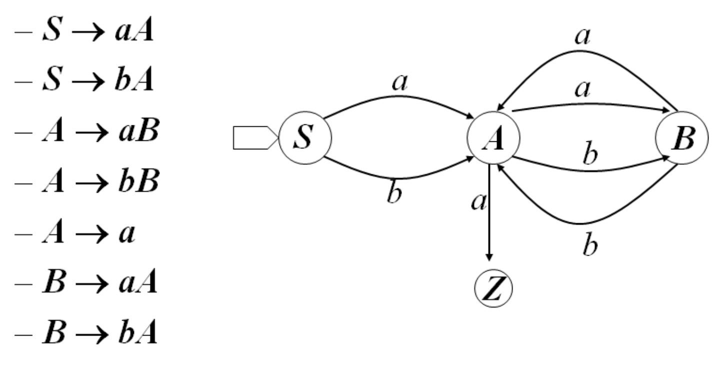
Las gramáticas regulares son parte esencial de los lenguajes regulares. Los autómatas finitos son una representación gráfica de los lenguajes regulares. Una palabra que pertenece a un lenguaje realmente es una cadena de símbolos o caracteres y por eso la relación que existe entre símbolos, cadenas, lenguajes, alfabetos y gramáticas es importante.

$$f'(x) = lim_{h\to 0}\frac{f(x+h)-f(x)}{h} = lim_{h\to 0}\frac{(x+h)^2-x^2}{h} = lim_{h\to 0}\frac{x^2+2xh+h^2-x^2}{h} =$$

$$lim_{h\to 0}\frac{2xh+h^2}{h} = lim_{h\to 0}\frac{h(2x+h)}{h} = lim_{h\to 0}(2x+h) = 2x$$

Lenguajes regulares

Con el alfabeto ∑ es posible formar una gran cantidad de lenguajes, pero no existe un método efectivo para saber cuántos de ellos son regulares. Todos los lenguajes sobre ∑ son sublenguajes del lenguaje universo ∑*, en lugar de ellos se utilizarán algunas propiedades de los lenguajes para determinar cuales de ellos son regulares.



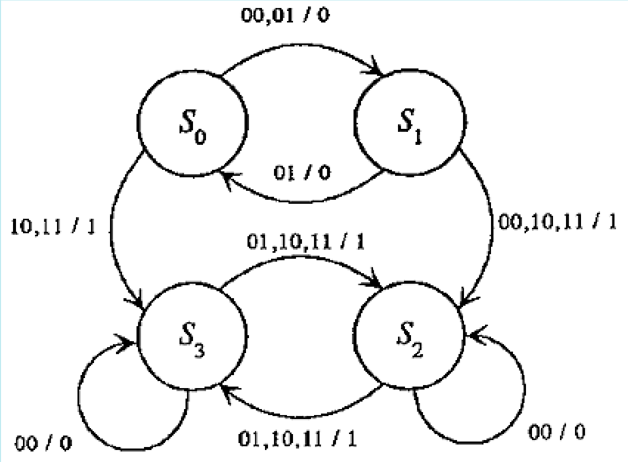
Expresiones regulares

Es una nueva forma de expresar los lenguajes regulares y tiene como finalidad facilitar la manipulación y simplificación de los mismos.

AXIOMA	DESCRIPCIÓN
$r \mid s = s \mid r$	$\mid$ es conmutativo
$r \mid (s \mid t) = (r \mid s) \mid t$	$\mid$ es asociativo
$(rs)t = r(st)$	la concatenación es asociativa
$r(s \mid t) = rs \mid rt$ $(s \mid t)r = sr \mid tr$	la concatenación distribuye sobre $\mid$
$\epsilon r = r$ $r\epsilon = r$	ϵ es el elemento identidad para la concatenación
$r^* = (r \mid \epsilon)^*$	la relación entre * y ϵ
$r^{**} = r^*$	* es idempotente

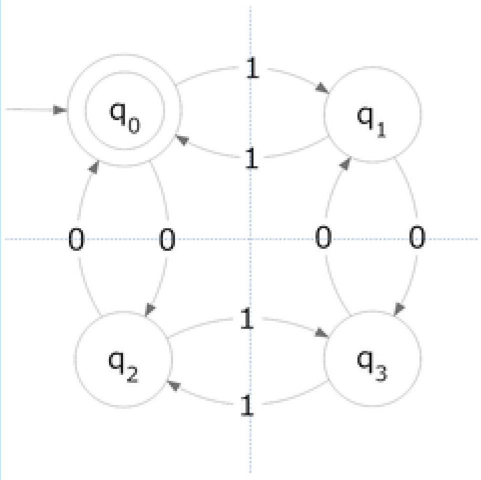
Diagrama de transición

En los diagramas de transición; los estados se representan por medio de un círculo con el nombre del estado dentro de ella. Los estados de aceptación o finales se distinguen porque tienen doble círculo, las transiciones se representan por aristas y se etiquetan con un símbolo del alfabeto. El estado inicial se distingue porque se hace incidir sobre él una flecha.



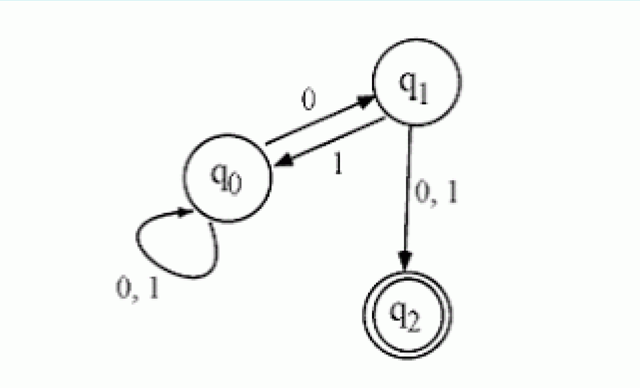
Autómatas finitos determinísticos

Se dice que un autómata finito es determinístico si por medio de la función de transición es posible determinar claramente cuál es el estado siguiente. El autómata anteriormente visto es un AFD ya que cuando se está en un estado cualquiera y se tiene un símbolo del alfabeto es posible determinar claramente cuál es el estado siguiente.



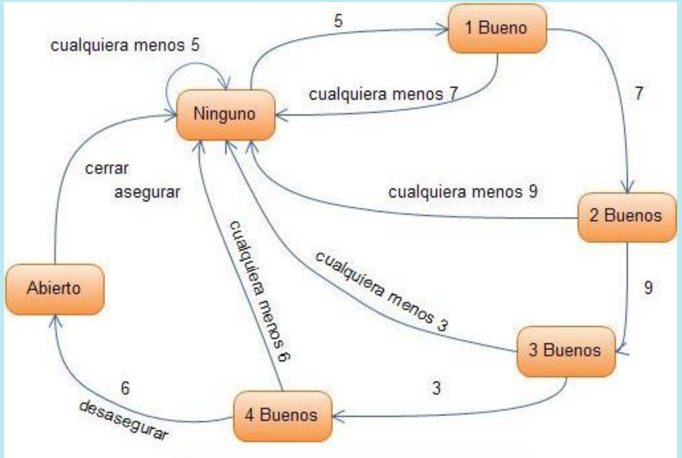
Autómatas finitos no determinísticos

La diferencia fundamental entre un autómata finito determinístico y un autómata finito no determinístico, es que en éste último la función de estado siguiente, no conduce a un estado único determinado.



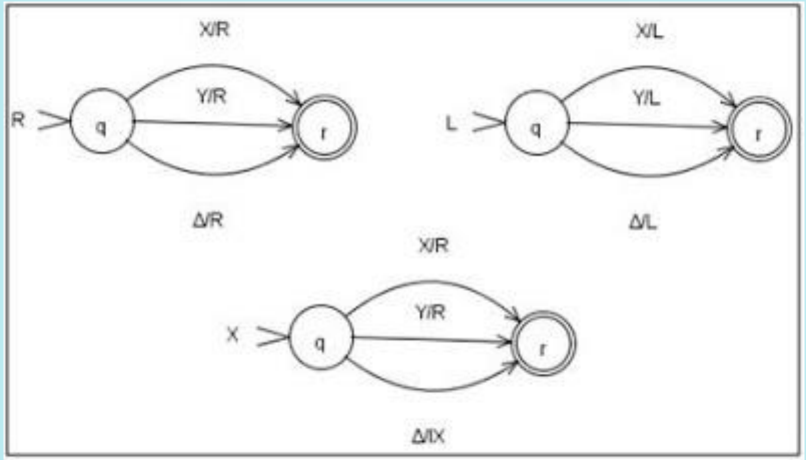
Máquinas de estado finito

Una máquina de estado finito es una forma de representar los autómatas finitos, en donde no existen estados aceptados y donde los símbolos de salida se colocan juntamente con los símbolos de entrada en cada una de las aristas de la máquina.



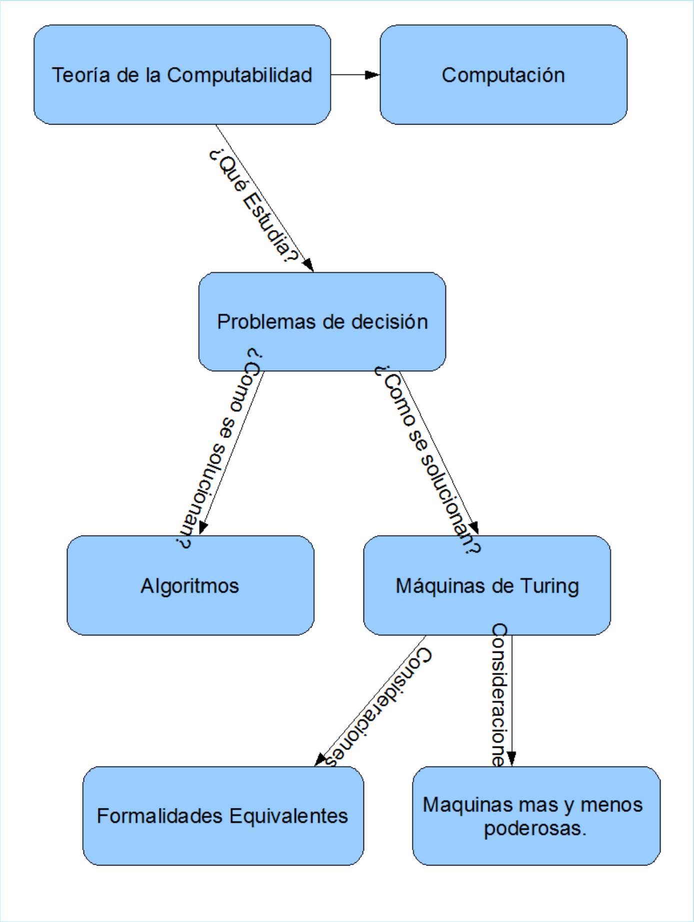
Máquinas de Turing

Una máquina de Turing consiste en una cinta que se extiende de manera infinita, en donde se escribe o se lee información por medio de una cabeza de lectura-escritura.



Teoría de la computabilidad

Es la parte de la computación que analiza y determina los problemas que pueden resolverse por medio de un algoritmo o bien por alguna de las MT.



Teoría de la complejidad

La complejidad es la cantidad de recursos necesarios para resolver un problema como son tiempo y espacio. El tiempo es el número de pasos de ejecución de un algoritmo para resolver un problema y el espacio la cantidad de memoria utilizada para resolver dicho problema.

1. $f(x) = 3x^4 - 2x^3 + 8x^2 + 7x + 1$
2. $f(x) = -6x^3 - 2x + 1$
3. $f(x) = 6x^4 - 7x^3 - 7x + 9$
4. $g(x) = x^5 - 3x^2 + 1$
5. $h(x) = 6x^6 - 7x^3 - 8x^2 + 9x - 1$

Aplicación de los lenguajes formales

Las máquinas de estado finito y los AFD admiten lenguajes regulares, lenguajes simples que normalmente se observan funcionando en circuitos lógicos de control sencillos, en donde las operaciones a realizar, están completamente determinadas por la información de entrada.