

**ESCUELA SUPERIOR DE
CÓMPUTO**

COMPILADORES

**TAREA 01: JERARQUÍA DE
CHOMSKY**

GRUPO: 3CM7

DÍAZ MEDINA JESÚS KAIMORTS

07 de agosto del 2018

Introducción

Chomsky

Lingüista estadounidense nacido en 1928, profesor del Instituto de Tecnología de Massachussets (MIT). Aparte de como lingüista y profesor universitario, es muy conocido por su faceta de intelectual y crítico activista político.

Se le considera el fundador de la Gramática generativa transformacional. Sus análisis del lenguaje parten de las oraciones básicas que se desarrollan y terminan en una variedad de combinaciones sintácticas al aplicar una serie de reglas.

En 1956 Naom Chomsky publica "**Three models for the description of language**" un trabajo en el que se describen las propiedades de los tipos de *lenguajes formales* y correspondientes *grámaticas* en relación a su complejidad computacional. Según Chomsky, los tipos de lenguajes formales pueden dividirse en tres: a) de estados finales (***o regulares***), b) de estructura de frase (***o libres de contexto***) y c) transformaciones (***o sensibles de contexto***). Tal clasificación es hoy en día conocida como la *jerarquía de Chomsky* (JCh).

Objetivo de Chomsky

El principal objetivo de Chomsky y su jerarquía era **demostrar que los dos primeros tipos de grámaticas son incapaces de dar cuenta, de manera simple y general, de la complejidad de las lenguas naturales**. En particular, Chomsky demostró que el inglés presenta propiedades que *no pueden ser reflejadas ni por grámaticas de estados finitos ni por gramáticas de estructura de frase*.

Gramática Formal

Una gramática formal consta de un **conjunto finito de símbolos terminales** (las palabras en un lenguaje formal), **un conjunto finito de símbolos no terminales**, **un conjunto de reglas de producción** con un lado izquierdo y otro derecho, y un símbolo inicial.

Las reglas se aplican sustituyendo la parte de la izquierda por la parte de la derecha. Una derivación es una secuencia de aplicaciones de reglas.

Cada gramática *define el lenguaje formal de todas las sentencias que están formadas exclusivamente por los símbolos terminales* a los que se puede llegar mediante derivación a partir del símbolo inicial.

Jerarquía de Chomsky

Gramática Tipo 0.

Lenguajes Recursivos(Sin restricciones).

Incluyen todas las gramáticas formales. Generan todos los lenguajes que pueden ser reconocidos por una máquina de Turing. También son conocidos como **lenguajes recursivos numerables**.

Gramática Tipo 1.

Gramática sensible al contexto.

Estas son generadoras de **lenguajes dependientes del contexto**. Éstas contienen reglas de producción de forma.

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

donde

A es un no terminal.

α, β, γ son cadenas de terminales y no terminales.

α y β pueden ser vacíos, pero γ ha de ser distinto del vacío.

Se denominan gramáticas dependientes del contexto, porque, como se observa, A puede ser sustituido por γ si está acompañada de α por la izquierda y de β por la derecha.

*Estos lenguajes son todos los lenguajes que pueden ser reconocidos por una máquina de Turing no determinista.
(autómatas lineales acotados.)*

Gramática Tipo 2.

Gramática independiente de contexto (libre de contexto).

Generan los lenguajes libres de contexto. Están definidas por reglas de la forma:

$$A \rightarrow \gamma$$

donde

A es un no terminal.

γ es una cadena de terminales y no terminales.

Se denominan independientes de contexto porque A puede sustituirse por γ independientemente de las cadenas por las que esté acompañada.

Los lenguajes independientes de contexto **constituyen la base teórica para la sintaxis de la mayoría de los lenguajes de programación**. Definen la sintaxis de las declaraciones, las proposiciones, las expresiones, etc. (es decir, *la estructura de un programa*.)

Estos lenguajes son todos los lenguajes que pueden ser reconocidos por los autómatas de pila.

Gramática Tipo 3.

Gramáticas Regulares.

Generan los lenguajes regulares. Las reglas se restringen a un único no terminal en la parte izquierda y una parte derecha compuesta por un único terminal que puede estar seguido o no de un único no terminal. Es decir, normas del tipo:

$$A \rightarrow aB$$

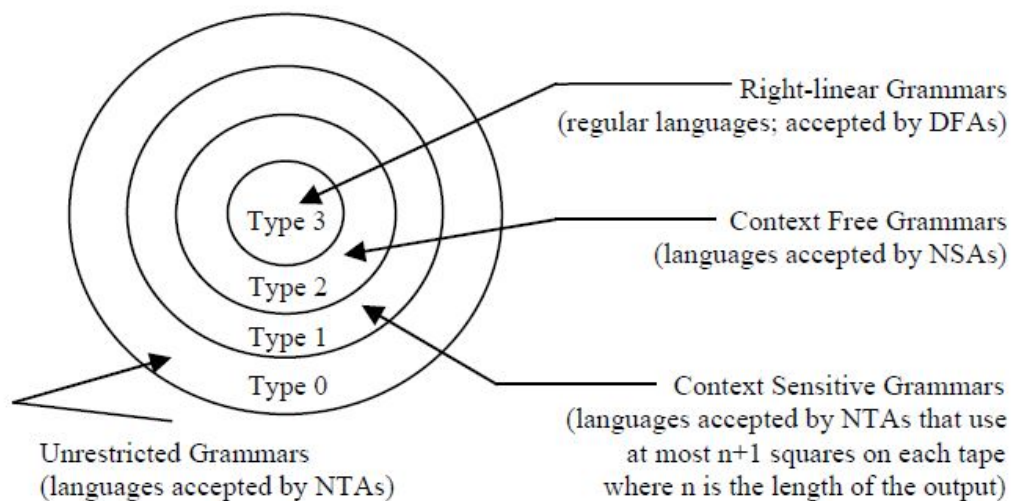
$$A \rightarrow a$$

Estos lenguajes son los que pueden ser *decididos por un autómata finito (regular)*. Los lenguajes regulares se utilizan para *definir estructura léxica de los lenguajes de programación*. Definen la sintaxis de los identificadores, número, cadenas y otros símbolos básicos del lenguaje.

[unsrt]

Referencias

- [1] ÁNGEL J.GALLEGO, *Jerarquía de Chomsky y la facultad del lenguaje: consecuencias para la variación y evolución.*, Vol.XXVII/2, pp.47-60, 2008.
- [2] MARTA GONZALO GARCÍA, «Clasificación de Lenguajes formales de Chomsky», *Teoría de Automátas y Lenguajes Formales*, págs. 1-3, 2001/2002.



Tipo 0: Lenguajes recursivos
Conjuntos de objetos formales de cualquier complejidad computacional

Tipo 2: Lenguajes libres de contexto
Conjuntos de secuencias de símbolos (o “frases”)

Tipo 1: Lenguajes sensibles al contexto
Conjuntos de conjuntos de secuencias de símbolos (o “cadenas”)

Tipo 3: Lenguajes regulares
Secuencias de símbolos

Tipo	Lenguaje	Autómata	Normas de producción de gramáticas
0	Recursivamente numerable (LRE)	Máquina de Turing (MT).	Sin restricciones
1	Dependiente del contexto (LSC)	Autómata linealmente acotado.	$\alpha A \beta \rightarrow \alpha \gamma \beta$
2	Independiente del contexto (LLC)	Autómata de pila.	$A \rightarrow \gamma$
3	Regular (LR)	Autómata finito	$A \rightarrow aB$ $A \rightarrow a$

Figura 1: Jerarquía de Chomsky