

Apuntes del 29 de abril

Compiladores e Intérpretes

Marlon Agüero Castro - 2015027137
FRANCISCO JOSÉ TORRES ROJAS

Contenido

Repaso de la clase pasada.....	3
Errores de Sintaxis – Recuperar.....	3
Errores de Sintaxis – Reparación	4
Traducción dirigida por Sintaxis	5
Análisis Sintáctico.....	5
Avram Noam Chomsky.....	5
Lenguajes no Regulares	6
Jerarquía de Chomsky.....	6
Gramática Libre de Contexto.....	7
Definición formal	7
Derivación	8
Notación	9
Derivación de w	9
Definiciones	10
Lenguaje finito.....	11
Definiciones.....	11
Análisis de sintáctico.....	13
Definición	13
Ejemplo 1	13
Ejemplo 2.....	13
Ejemplo 3.....	14
Ejemplo 4.....	14
Ejemplo 5.....	14
Ejemplo 6.....	15
Ejemplo 7.....	15
Ejemplo 8.....	15

Repaso de la clase pasada

Con respecto al examen ya se revisaron las dos preguntas con dudas y una se le puso buena a todos (la que estaba mala) y la otra se le dio buena solo a los que marcaron la respuesta que luego de revisar era la correcta. En el foro están las nuevas notas, pero los cambios son mínimos.

Errores de Sintaxis – Recuperar

- Se recupera porque no es práctico que solo se muestre un error de los muchos que pueden existir en tiempo de compilación.
- El parse debe poder avanzar hasta un punto seguro del programa y seguir la compilación en busca de errores.

Problemas

- Saltar demasiado
- Cascada de errores por no recuperarse bien.

El profe en este punto puso una imagen que no estaba bien para ninguno del grupo, yo propongo esta.



Fin del repaso.

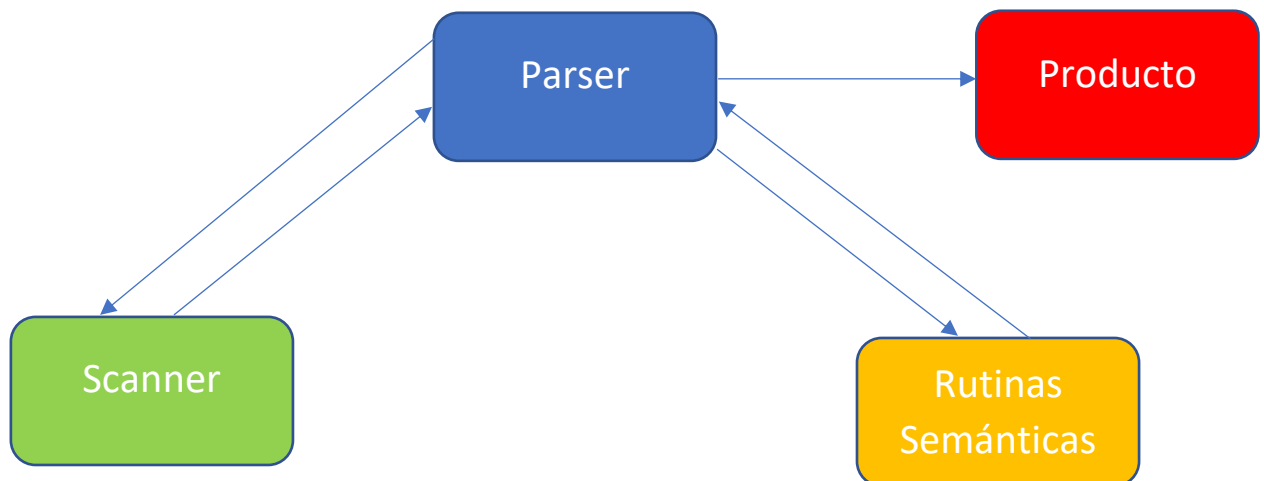
Errores de Sintaxis – Reparación

Una pregunta que se pueden hacer algunas personas, es por qué, si el parser sabe que está malo, porque no lo arregla y ya. Pero deberíamos preguntarnos, esto realmente nos beneficia. Primero, porque tendríamos que confiar en que el compilador arregle el programa realmente y no solo que lo haga superar la etapa de compilación sin nosotros saber que hizo realmente lo que necesitábamos. Segundo, si un compilador puede hacer esto es muy posible que nos quite el trabajo en muchas ocasiones.

- Para la reparación es posible que a un nivel muy simple se utilice para la recuperación.
- Los warnings pueden ser resultado de una reparación.
- Para un error pueden existir muchas reparaciones.



Traducción dirigida por Sintaxis



- Es la organización más usual.
- El parser dirige todo el proceso
 - Llama al parser cuando ocupa un token
 - En puntos apropiados llama a rutinas semánticas.
- El trabajo del parser en ocasiones llega a generar código intermedio y a veces genera lenguaje máquina.

Análisis Sintáctico

Gramáticas libres de contexto.

Avram Noam Chomsky

Lingüista y filósofo estadounidense, introducido en la lingüística por su padre, quien era especialista en lingüista hebrea. Obtuvo su doctorado en 1955 en la Universidad de Pensilvania. A partir de 1961 se convierte en profesor en el MIT. Donde ahora se considera emérito.

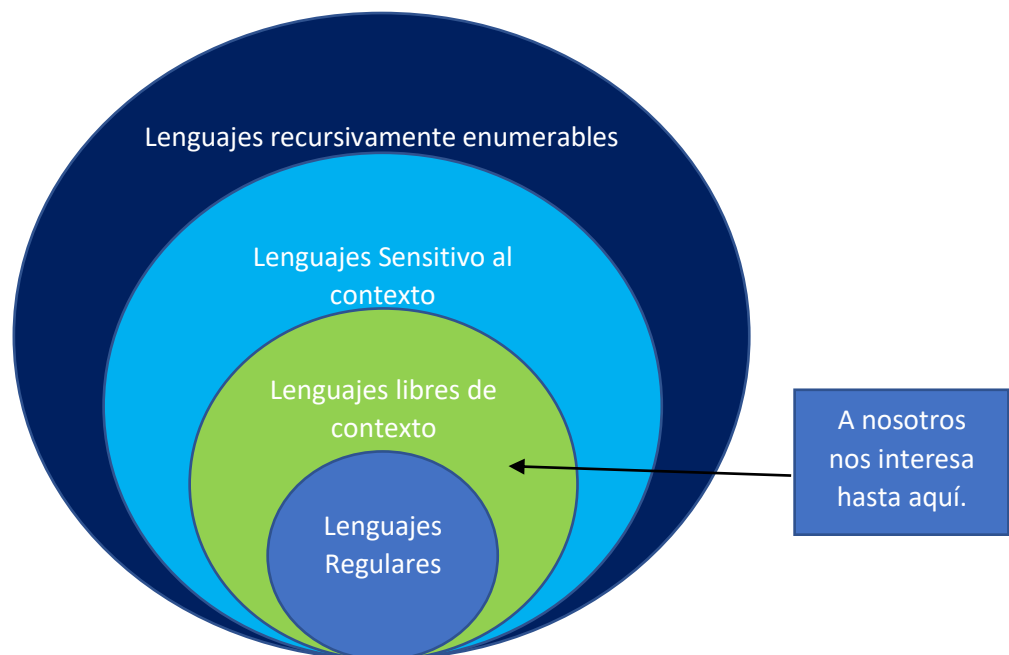
Considerado padre de la lingüística, es el autor vivo más citado. Es una de las figuras culturales más importantes de nuestros tiempos. Intervino en la política con sus críticas al imperialismo estadounidense, desde el inicio de la guerra de Vietnam.



Lenguajes no Regulares

El pumping lemma nos permite demostrar que algunos lenguajes no son regulares. En algunas ocasiones podíamos suponerlos al intentarlos hacer.

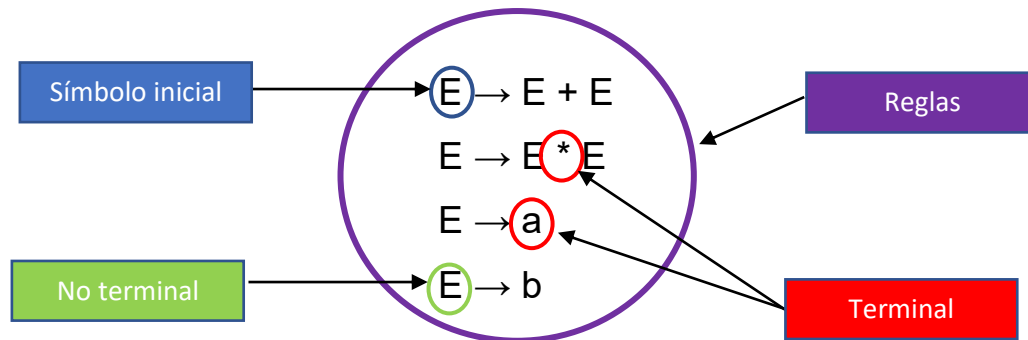
Jerarquía de Chomsky



La imagen anterior quiere explicar que los lenguajes libres de contexto contienen a los lenguajes regulares, por lo tanto, todo lenguaje regular es un lenguaje libre de contexto, pero no todos los lenguajes libres de contexto son regulares. Y esto con cada nivel.

Gramática Libre de Contexto

Los siguiente es un ejemplo de gramática libre de contexto.



El símbolo inicial se debe definir, los no terminales están a la izquierda, aunque también podrían estar al lado derecho. Los terminales están al lado derecho y solo ahí pueden estar.

Definición formal

Una gramática libre de contexto, desde ahora una CFG, es lo siguiente:

G es un cuarteto de la forma $G = (V, \Sigma, P, S)$ donde:

- V es un conjunto finito de no terminales
- Σ es un conjunto finito de terminales
- P es un conjunto finito de reglas
- $S \in V$ es el símbolo inicial

El S es un terminal y puede ser lo que quiera, mientras se defina que es el inicial, se le puede poner V_0 .

- $V \cap \Sigma = \emptyset$

Los terminales y los no terminales son conjuntos disjuntos.

- $P \subseteq V \times (V \cup \Sigma)^*$

Es el producto cartesiano del conjunto de no terminales con las hileras de terminales y no terminales de cualquier longitud, incluidos ϵ .

Es básicamente que siempre hay un no terminal (solo uno) y luego una hilera de ya sea terminales o no terminales incluyendo ϵ a la derecha. Eso es una regla.

Derivación

Sean α , β y γ hileras conformadas de terminales y no terminales

$(V \cup \Sigma)^*$. Si se tiene una hilera w de la forma $\alpha A \beta$ y existe una regla de la gramática de la forma $A \rightarrow \gamma$, entonces w se puede reescribir como $\alpha \gamma \beta$. Es libre de contexto porque solo me importa quién es A , α y β , no me importa para nada quienes son.

- Podemos decir que w deriva a la hilera $\alpha \gamma \beta$ y se denota como $w \Rightarrow \alpha \gamma \beta$.
- Se puede derivar mientras haya terminales.
- Para aplicar una regla solo interesa el no terminal a expandir, el resto no.

En este momento solo nos interesa los lenguajes libres de contexto, los lenguajes sensitivos al contexto no los queremos, por eso todos deben decir.



Notación

- Un paso de derivación:
 - $w \Rightarrow v$
- Uno o más pasos de derivación:
 - $w \Rightarrow^+ v$
- Cero o más pasos de derivación:
 - $w \Rightarrow^* v$

Una hilera con cero pasos de derivación es la misma hilera.

Derivación de w

Sea $G = (V, \Sigma, P, S)$ una CFG y sea $w \in (V \cup \Sigma)^*$.

El conjunto de hileras derivables de w se define recursivamente como:

1. w es derivable de w. Es decir, toda hilera deriva a sí misma.
2. Si $v = \alpha A \beta$ es derivable desde w y $A \rightarrow \gamma \in P$ entonces $\alpha \gamma \beta$ es derivable desde w. Todas las hileras derivables se generan partiendo desde un no terminal y usando las reglas de la gramática para generar todas las hileras posibles.
3. Todas y únicamente las hileras construidas aplicando el paso 2 son derivables desde w.

En resumen, se tiene $w = \{v \mid w \Rightarrow^* v\}$

Ejemplo- Sea G definida como:

$$E \rightarrow E + E$$

$$E \rightarrow E * E$$

$$E \rightarrow a$$

$$E \rightarrow b$$

- Hileras derivables de abba.

$\{abba\}$

- Hileras derivables de $*EaE$

$\{ *EaE, *aaE, *aaa, *E+EaE, *E*EaE, *E+E+EaE*E, \dots \}$

En este caso esta se puede hacer de forma compacta de la siguiente forma: $\{w \mid *EaE \Rightarrow^* w\}$.

Definiciones

Sea $G = (V, \Sigma, P, S)$ una CFG.

1. Una hilera $w \in (V \cup \Sigma)^*$ es una forma sentencial de G , si y solo si $S \Rightarrow^* w$. Cualquier hilera partiendo desde el símbolo inicial, que se pueda derivar usando las reglas de la gramática.

2. Una hilera $w \in \Sigma^*$ es una sentencia de G , si y solo si $S \Rightarrow^* w$. Cualquier hilera partiendo desde el símbolo inicial, que se puede derivar usando las reglas de la gramática, pero que solo contiene terminales.

*** Toda sentencia es una forma sentencial, pero no toda forma sentencial es una sentencia.

3. El lenguaje de G , denotado como $L(G)$ es el conjunto $\{v \mid v \in \Sigma^* \text{ y } S \Rightarrow^* v\}$.

4. El lenguaje de G es el conjunto de todas las sentencias, es decir, no puede haber no terminales.

Ejemplo- Sea G definida como:

$E \rightarrow E + E$

$E \rightarrow E * E$

$E \rightarrow a$

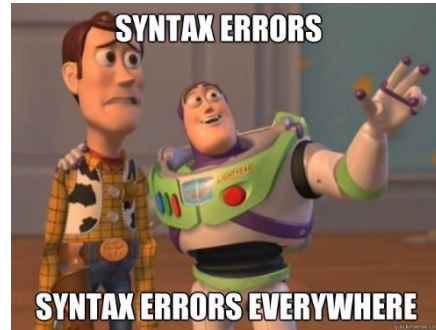
$E \rightarrow b$

- Formas sentenciales de G:
 $\{E, E+E, E+E^*E, E^*E, a, a+a, b, \dots\}$
- Lenguaje de G:
 $\{a, b, a+a, a+b, a^*a, b+a^*b, b+b+b+b+b+b, \dots\}$

Puede existir lenguajes vacíos.

Errores de sintaxis

- aa
- ++
- **
- bb



Lenguaje finito

$$S \rightarrow a$$

$$S \rightarrow b$$

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

Definiciones

Una derivación es leftmost (más a la izquierda), si siempre se reemplaza el no terminal que este más a la izquierda de la hilera actual.

Ejemplo- Sea G definida como:

$$E \rightarrow E + E$$

$$E \rightarrow E * E$$

$$E \rightarrow a$$

$$E \rightarrow b$$

Respuesta:

- E
- $\Rightarrow E^*E$
- $\Rightarrow E+E^*E$
- $\Rightarrow b+E^*E$
- $\Rightarrow b+E^*E^*E$
- $\Rightarrow b+a^*E^*E$
- $\Rightarrow b+a^*a^*E$
- $\Rightarrow b+a^*a^*b$

Una derivación rightmost (más a la derecha), si siempre se reemplaza el no terminal que este más a la derecha de la hilera actual.

Ejemplo- Sea G definida como:

$$E \rightarrow E + E$$

$$E \rightarrow E * E$$

$$E \rightarrow a$$

$$E \rightarrow b$$

Respuesta:

- E
- $\Rightarrow E+E$
- $\Rightarrow E+E^*E$
- $\Rightarrow E+E^*E+E$
- $\Rightarrow E+E^*E+b$
- $\Rightarrow E+E^*E+E+b+b$
- $\Rightarrow E+E^*a+b+b$
- $\Rightarrow E+a^*a+b+b$
- $\Rightarrow b+a^*a+b+b$

Análisis de sintáctico

Definición

Un conjunto de hileras es un lenguaje libre de contexto si es completamente generado por una gramática libre de contexto $G = (V, \Sigma, P, S)$, además G no genera hileras que no pertenezcan al lenguaje.

Ejemplo 1

Presente una CFG que genere el lenguaje de hileras de 1 o más “a’s”.

Primera opción:

$$A \rightarrow aA$$

$$A \rightarrow a$$

Segunda opción:

$$E \rightarrow EE$$

$$E \rightarrow a$$

Tercera opción más compacta:

$$A \rightarrow a|aA$$

ó

$$E \rightarrow a|EE$$



Ejemplo 2

Presente una CFG que genere el lenguaje de hileras de 0 o más “a’s”.

Respuesta

$$A \rightarrow \varepsilon|aA$$



Ejemplo 3

Presente una CFG que genere el lenguaje de hileras sobre $\Sigma = \{0,1\}$ que terminen en 1.

Primera opción:

$$S \rightarrow A1$$

$$A \rightarrow 0A$$

$$A \rightarrow 1A$$

$$A \rightarrow \varepsilon$$



Segunda opción:

$$E \rightarrow 1$$

$$E \rightarrow 0E$$

$$E \rightarrow 1E$$



Ejemplo 4

Presente una CFG que genere el lenguaje de hileras sobre $\Sigma = \{0,1\}$ que inicie en 0 y termine en 1.

Respuesta

$$S \rightarrow 0A1$$

$$A \rightarrow 0A$$

$$A \rightarrow 1A$$

$$A \rightarrow \varepsilon$$



Ejemplo 5

Presente una CFG que genere el lenguaje $\{a^n b^m \mid n > 0, m > 0\}$

$$S \rightarrow AB$$

$$A \rightarrow aA \mid a$$

$$B \rightarrow bB \mid b$$



Ejemplo 6

Presente una CFG que genere el lenguaje $\{a^n b^m \mid n \geq 0, m \geq 0\}$

$$S \rightarrow AB$$

$$A \rightarrow aB \mid \varepsilon$$

$$B \rightarrow bB \mid \varepsilon$$



Ejemplo 7

Sea G definida como:

$$S \rightarrow aSa$$

$$S \rightarrow aBa$$

$$B \rightarrow bB$$

$$B \rightarrow b$$

¿Cuál es el lenguaje de G?

$$\{a^n b^m a^n, n > 0, m > 0\}$$



Ejemplo 8

Presente una CFG que genere el lenguaje $\{a^n b^m c^m d^{2n}, n \geq 0, m > 0\}$

$$S \rightarrow aSdd$$

$$S \rightarrow B$$

$$B \rightarrow bBc$$

$$B \rightarrow bc$$

