Universidad Nacional de Cajamarca
"Norte de la Universidad Peruana"

UNIDAD I:
LENGUAJES Y GRAMATICAS

ING. SANDRA RODRÍGUEZ AVILA

Resultado de aprendizaje 1 (RA-1): Describe aspectos fundamentales de la teoría de lenguajes formales y compiladores teniendo en cuenta los antecedentes históricos, conceptos básicos y principales aplicaciones.

Resultado de aprendizaje 2 (RA-2): Desarrolla problemas de determinación de lenguajes y diseño de gramáticas formales basados en la jerarquía de Chomsky y algoritmos de simplificación de gramáticas.

CONTENIDO

INTRODUCCION

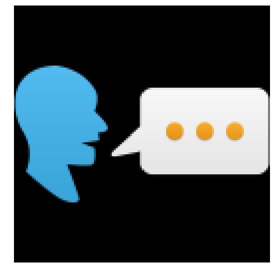
Tenemos experiencia con:

- Lenguajes naturales y gramáticas.
- Lenguajes de programación.

Sabemos que:

- Las palabras de un lenguaje se forman con los símbolos de un alfabeto
- Las gramáticas define la estructura de las frases y las palabras.

LENGUAJES Y GRAMATICAS



Noam Chomsky:

- *Teoría de lenguajes formales* (1950)
- Herramienta para los lenguajes naturales y los lenguajes de computadoras (1963)

LENGUAJES: DEFINICIONES BASICAS

- Alfabeto: conjunto no vacío y finito de símbolos.
  Ejemplos:
  - Alfabeto castellano

    V= {a, b, c, …,ñ, …, z}
  - Alfabeto para números en base 10

    V= {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

- Palabras (tiras o cadenas) : secuencia o concatenación de símbolos de un alfabeto.

  Ejemplos: caja, universidad, curso, 3, 2020, etc.

  x=caja y=marca

LENGUAJES: DEFINICIONES BASICAS

- Palabra o tira mínima o nula: λ
- Longitud de una palabra: Número de letras

  Ejemplos

  si x=caja entonces $|x| = |caja| = 4$

  $|\lambda| = 0$

- Lenguaje Universal: conjunto infinito de todas las palabras sobre un alfabeto

  Ejemplo: si $V=\{a\}$

  Lenguaje universal = $V^*=\{\lambda, a, aa, aaa, \ldots\}$

OPERACIONES CON PALABRAS

- Concatenación:

  Ejemplo: x=caja   y=marca ⇒ xy = cajamarca

  Propiedades:

  - Operación cerrada
  - Asociativa: $(xy)z=x(yz)$
  - Elemento neutro: λ

    $x\lambda=\lambda x=x$
  - $|xy| = |x| + |y|$
  - Conmutativa: No se cumple

OPERACIONES CON PALABRAS

- Potencia i-ésima: x^{i}

  Ejemplo: si $x=ab \Rightarrow x^3 = ababab$

  Propiedades:
  - $x^{i+j} = x^i x^j$
  - $|x^i| = i|x|$
  - $x^0 = \lambda$

OPERACIONES CON PALABRAS

- Inversa de una palabra: x^{-1}

Ejemplo: si $x=ab \Rightarrow x^{-1} = ba$

Si $y=010 \Rightarrow y^{-1} = 010$

Propiedad:

- $|x|=|x^{-1}|$

OPERACIONES CON LENGUAJES

- Lenguaje: subconjunto del lenguaje universal definido sobre un alfabeto

  Sea $V=\{0,1\} \Rightarrow L1=\{x/\ |x|=2\}=\{00, 01,10,11\}$

- Lenguaje vacío: $L_{\emptyset}= \emptyset = \{\ \}$

- Lenguaje λ : $L= \{\lambda\}$

OPERACIONES CON LENGUAJES

- Unión: definido sobre un mismo Alfabeto

$$L=L1 \cup L2=L1+L2=\{x/\ x \in L1 \lor x \in L2\}$$

Propiedades:

- Operación cerrada
- Asociativa: $(L1+L2)+L3=L1+(L2+L3)$
- Conmutativa: $L1+L2=L2+L1$
- Elemento neutro: $\emptyset$

$$(L1+\emptyset)=L1$$

- Idempotencia: $(L+L)=L$

OPERACIONES CON LENGUAJES

- Concatenación: definido sobre un mismo Alfabeto

$$L=L1L2=\{xy \;/\; x\in L1 \wedge y\in L2\}$$

Propiedades:

- Operación cerrada
- Asociativa: $(L1L2)L3=L1(L2L3)$

- Conmutativa: No cumple
- Elemento neutro: $(L_\lambda=\{\lambda\})$

$$L_\lambda L=LL_\lambda=L$$

OPERACIONES CON LENGUAJES

- Potencia i-ésima: L^i

$$L^i = L \ldots L \text{ (i veces)}$$

Propiedades:

- $L^{i+j} = L^i L^j$
- $L^0 = L_\lambda = \{\lambda\}$

OPERACIONES CON LENGUAJES

- Cierre o clausura: L^*

$$L^*=L^0 U L^1 \ldots L^{\infty}$$

Propiedades:

- $L^{i+j}=L^iL^j$
- $L^0=L_\lambda=\{\lambda\}$

OPERACIONES CON LENGUAJES

- Clausura positiva: L^+

$$L^+=L^1 U L^2 \ldots L^{\infty}$$

Propiedades:

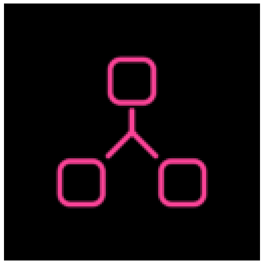
- $L^*=L^+ \cup \{\lambda\}$
- $L^+=L^*L=LL^*$
- Lenguaje universal=V^*

OPERACIONES CON LENGUAJES

- Inversa de un lenguaje: L^{-1}

$$L^{-1}=\{\ x^{-1}/x \in L\ \}$$

GRAMATICA FORMAL



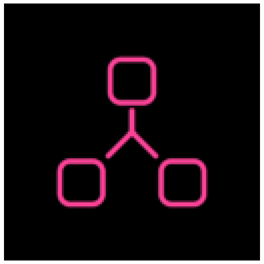
- Gramática:
  - Describe la estructura de las frases y palabras de un lenguaje a través de reglas.
  - Lenguajes naturales

El estudiante juega fútbol

  - Lenguajes artificiales para computadora

If (a>b) c=0 else c=1;

GRAMATICA FORMAL – DEFINICIONES PREVIAS



- Regla o producción: o regla de derivación o sintaxis

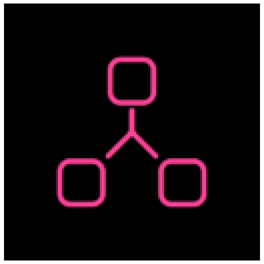
  <oración> := <sujeto> <predicado>

  <sent IF> → if <condicion> <sent> else <sent>

  : = o → : “se define como”

- P: conjunto de reglas de producción definidas sobre un Alfabeto.

GRAMATICA FORMAL – DEFINICIONES PREVIAS



- Derivación directa: Sea un conjunto de reglas P definidas sobre un alfabeto.

  P: conjunto de reglas

  $S \rightarrow xAz$

  $A \rightarrow y$

$$S \Rightarrow xAz \Rightarrow xyz$$

EJEMPLO LENGUAJE NATURAL Y GRAMATICA

- Reglas gramaticales (P):

1. <oración> := <sujeto> <predicado>
2. <sujeto> := <articulo> <sustantivo>
3. <predicado> := <verbo>
4. <predicado> := <verbo> <complemento>
5. <articulo> := el | la
6. <sustantivo> := estudiante |asignatura
7. <verbo> := juega | es
8. <complemento>:= fútbol | interesante

EJEMPLO LENGUAJE NATURAL Y GRAMATICA

- Árbol sintáctico:

<oración> 1

<sujeto> 2 <predicado> 4

<articulo> <sustantivo> <verbo> <complemento>

5 6 7 8

el estudiante juega fútbol

EJEMPLO LENGUAJE DE PROGRAMACION

- Reglas gramaticales (P):

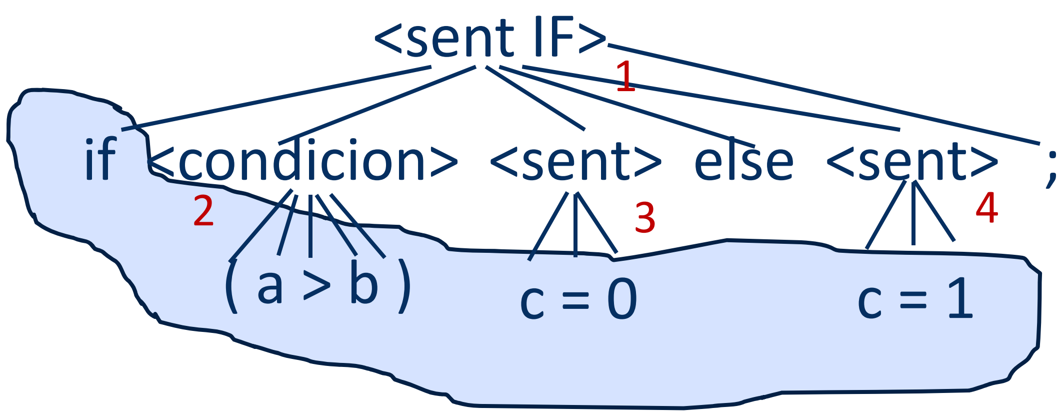
1. <sent IF> → if <condicion> <sent> else <sent>;
2. <condicion> → (a>b)
3. <sent> → c=0
4. <sent> → c=1

N={<sent IF>, <condicion>, <sent> }

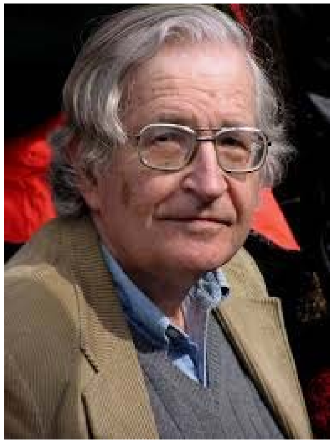
T= {if, else, ;, (, a, >, b,), c, =, 0, 1}

EJEMPLO LENGUAJE DE PROGRAMACIÓN

- Árbol sintáctico:



NOCIÓN DE GRAMÁTICA



- Chomsky, 1959.
- Definición formal

$$G = (N, T, P, S)$$

N: Vocabulario No terminal

T: Vocabulario Terminal (tira o palabra terminal)

P: conjunto de reglas de derivación

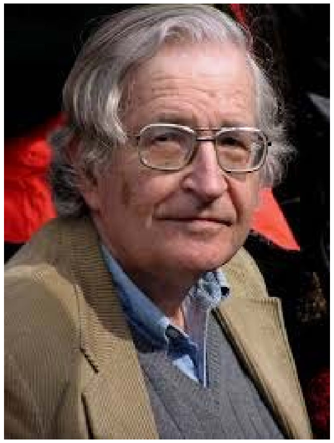
tira1 → tira2

tira1 := tira2

tira1 se define como tira2

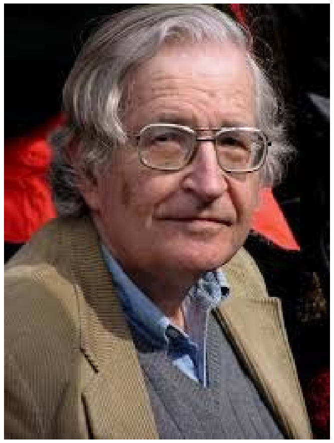
S: símbolo inicial o axioma

NOCIÓN DE GRAMÁTICA



- $N \cap T = \{\emptyset\}$
- $N \cup T = V$
- Las reglas P se aplican desde el símbolo inicial S para obtener las cadenas, palabras o tiras del lenguaje.

NOCIÓN DE GRAMÁTICA-EJEMPLOS



Ejemplos:

- G1=(N, T, P, S)

donde: N={S} T={a, b}

P: S→ab S→aSb

- G2=(N, T, P, S)

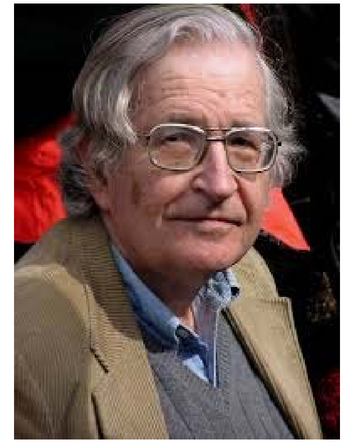
donde: N={S, A, B} T={a, b, c, d}

P: S→ASB A→b

A→aaBB S→d

A→aA B→dcd

NOCIÓN DE GRAMÁTICA-EJEMPLOS



- G3=(N, T, P, S)

donde: N={<numero>, <digito>}

T={0,1,2,3,4,5,6,7,8,9}

P: <numero>:= <digito><numero>

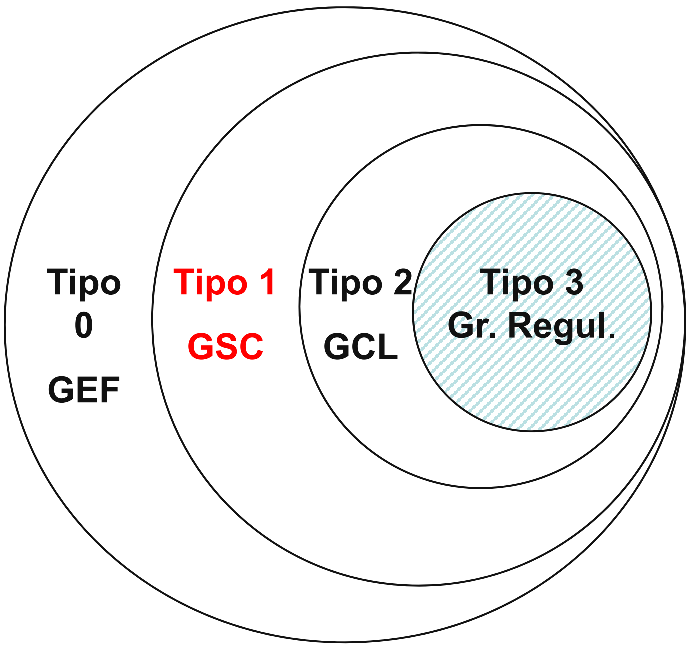
<numero>:= <digito>

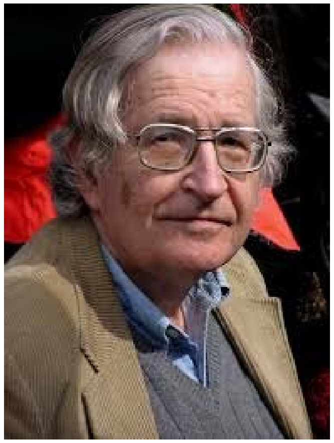
<digito>:= 0|1|2|3|4|5|6|7|8|9

S ={<numero>}

JERARQUIA DE LAS GRAMATICAS

- Chomsky definió cuatro tipos de gramáticas





JERARQUIA DE LAS GRAMATICAS

A. Gramática de tipo 0 o con estructura de frase:

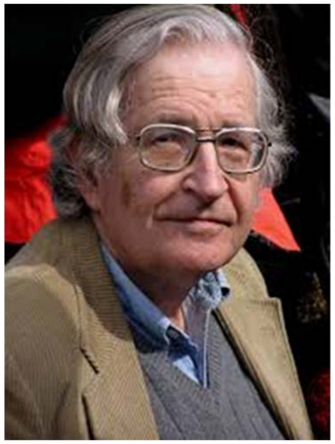
Las reglas P tienen la forma:

$$\alpha \rightarrow \beta$$

siendo α ε $(N \cup T)^+$ y β ε $(N \cup T)^*$

Ejemplos de reglas:

$S \rightarrow \lambda$ $AXb \rightarrow YdE$ $X \rightarrow ZYE$



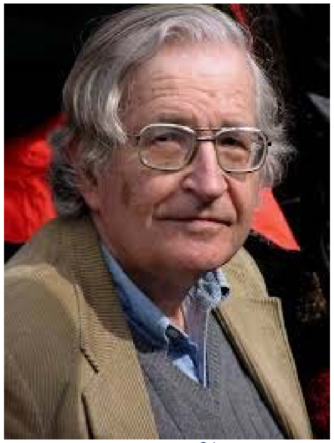
JERARQUIA DE LAS GRAMATICAS

B. Gramática de tipo 1 o sensibles al contexto:

Las reglas P tienen la forma:

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

siendo $A \varepsilon N$, α y $\beta \varepsilon (N \cup T)^*$ y $\gamma \varepsilon (N \cup T)^+$



Ejemplos de reglas:

$\lambda S \lambda \rightarrow \lambda X \lambda$ $\quad Y \rightarrow d \quad dEF \rightarrow deF$

$eE \rightarrow ee$

JERARQUIA DE LAS GRAMATICAS

c. Gramática de tipo 2 o con de contexto libre:

Las reglas P tienen la forma:

$$A \rightarrow \alpha$$

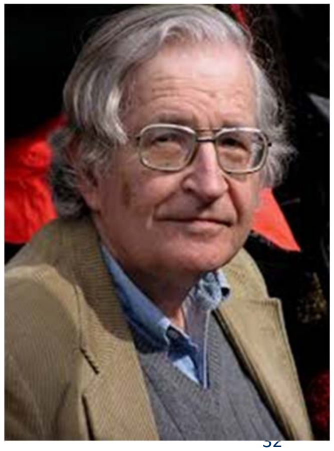
siendo $A \; \varepsilon \; N$, $\alpha \; \varepsilon \; (N \; U \; T)^*$

Ejemplos de reglas:

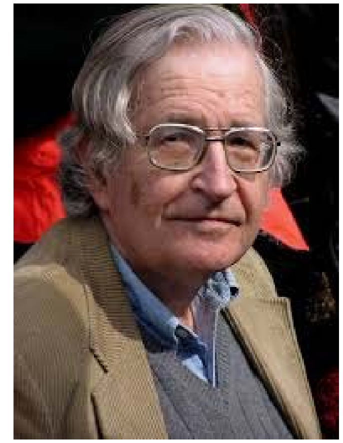
$S \rightarrow \lambda$ $S \rightarrow bA$ $A \rightarrow a$

$A \rightarrow aS$ $A \rightarrow bAA$ $B \rightarrow b$

$B \rightarrow bS$ $B \rightarrow aBB$



JERARQUIA DE LAS GRAMATICAS



D. Gramática de tipo 3 o regulares:

- Gramática lineal por la derecha
  - Las reglas P tienen la forma:

$$A \rightarrow aB \quad \text{ó} \quad A \rightarrow a$$

siendo A, B ε N, a ε T

- Gramática lineal por la izquierda
  - Las reglas P tienen la forma:

$$A \rightarrow Ba \quad \text{ó} \quad A \rightarrow a$$

siendo A, B ε N, a ε T

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

A**.** Derivación directa o inmediata

- La tira α produce o deriva directamente la tira β

$$\alpha \Rightarrow \beta$$

$$\text{si } \alpha = \delta A \mu \quad \text{y} \quad \beta = \delta \gamma \mu$$

y existe una regla de P, que sea: $A \rightarrow \gamma$

- Se puede aplicar repetidamente la noción de derivación directa: $\alpha \Rightarrow^{+} \beta$

•Si se incluye el caso de identidad, se escribe:

$$\alpha \Rightarrow^{*} \beta$$

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

B. Definición formal del lenguaje

Sentencias L(G):

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

- Ejemplo:

$$S \overset{(1)}{\Rightarrow} \mathbf{ab}$$

$$S \overset{(2)}{\Rightarrow} \mathbf{a}S\mathbf{b} \overset{(1)}{\Rightarrow} \mathbf{a}ab\mathbf{b}$$

$$S \overset{(2)}{\Rightarrow} \mathbf{a}S\mathbf{b} \overset{(2)}{\Rightarrow} \mathbf{a}aSb\mathbf{b} \overset{(1)}{\Rightarrow} \mathbf{aa}ab\mathbf{bb}$$

L(G) = { ab, aabb, aaabbb, ….}

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

Ejemplo:

- Sea la gramática: G1=(N, T, P, S)

donde: N={S} T={a, b}

P: S→ab (1) S→aSb (2)

$$S \overset{(1)}{\Rightarrow} ab$$

$$S \overset{(2)}{\Rightarrow} aSb \overset{(1)}{\Rightarrow} aabb$$

$$S \overset{(2)}{\Rightarrow} aSb \overset{(2)}{\Rightarrow} aaSbb \overset{(1)}{\Rightarrow} aaabbb$$

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

Formas Sentenciales D(G):

- Conjunto de tiras de símbolos terminales o no terminales.

$$D(G) = \{\alpha \mid (S \Rightarrow^{*} \alpha) \text{ and } \alpha \; \varepsilon \; (N \cup T)^{*}\}$$

- En el ejemplo anterior:

$$\overset{(2)}{S \Rightarrow} \; \overset{(2)}{aSb \Rightarrow} \; \overset{(1)}{aaSbb \Rightarrow} \; aaabbb$$

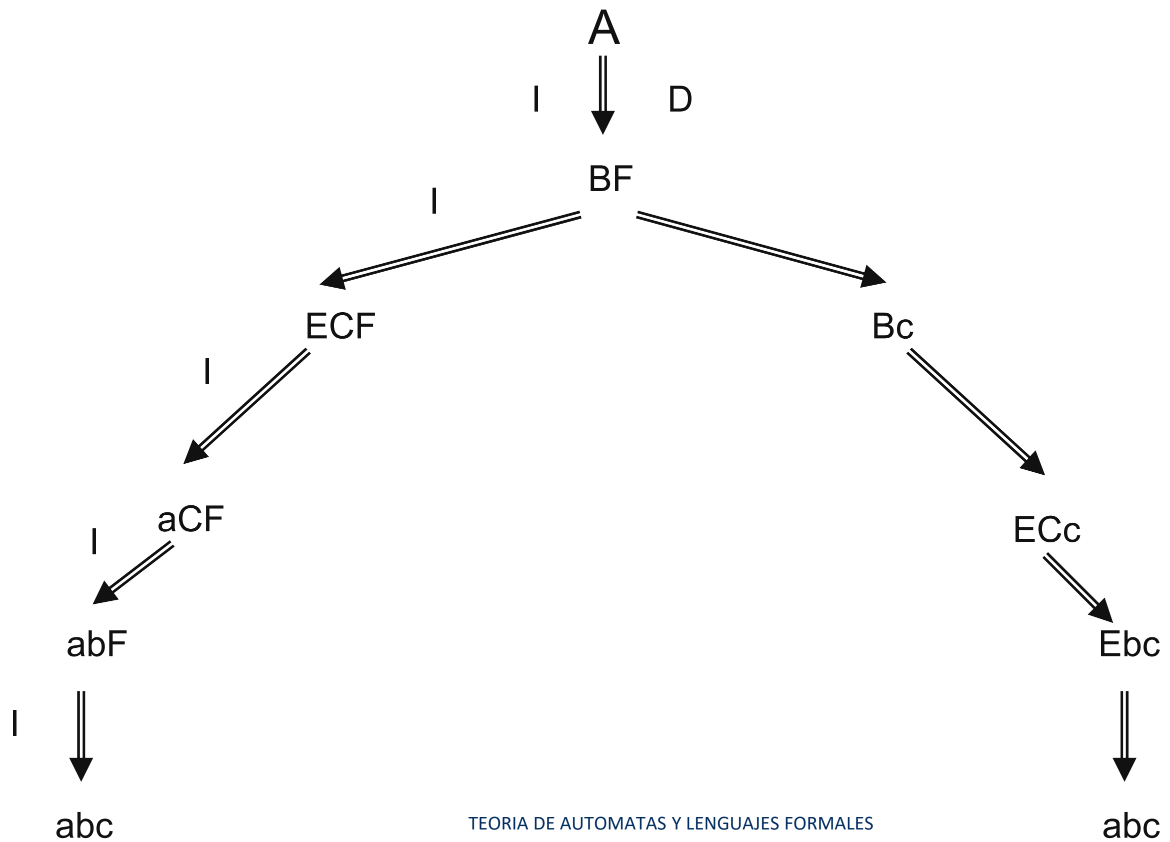
Las 4 tiras S, aSb, aaSbb y aaabbb, cumplen con las dos condiciones, por lo tanto son formas sentenciales de G1.

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

C. Derivación izquierda

- Derivación del símbolo no terminal "más hacia la izquierda" por algunas de sus partes derechas que la definen (reglas P).
- Análogamente, se define la derivación derecha.
- Sean las reglas P de una gramática:

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE



A→ BF

B→ EC

E →a

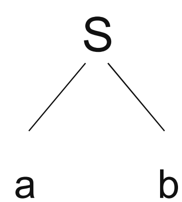
C→ b

F →c

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

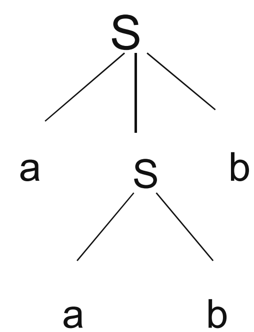
D. Árboles sintácticos

- Representación gráfica del proceso de reconocimiento o "parsing" de una sentencia o tira terminal.
- Ejemplo:



(1)

Tira: **ab**



(2)

(1)

Tira: **aabb**

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

F. Recursividad

- Mecanismo amplificador que nos permitirá definir formas complicadas de un lenguaje con reglas sucintas.
- Se requieren de dos tipos de reglas:
  - Una regla (**no recursiva**)
  - Una o mas reglas **recursivas**
- La recursividad define un lenguaje de programación de infinitas tiras o sentencias con un numero limitado de reglas.

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

- Ejemplos de Recursividad
  - Tren

    <tren> → <locomotora>

    <tren> → <tren> <vagón>
  - Entero sin signo:

    <entero> → <digito>

    <entero> → <entero> <digito>
  - Identificador:

    <identificador> → <letra>

    <identificador> → <identificador> <letra>

    <identificador> → <identificador> <digito>

GRAMATICA DE TIPO 2 O CONTEXTO LIBRE

H. Ambigüedad

El perro de mi tío.

- Una gramática es ambigua si el lenguaje definido tiene alguna sentencia que tenga más de un árbol sintáctico.
- No existe ninguna algoritmo que acepte una gramática y determine con certeza y en un tiempo finito si la gramática es ambigua o no.

ACTIVIDAD PRACTICA

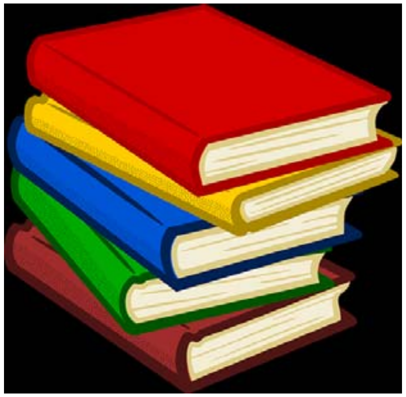
- Resolver los ejercicios de la Actividad Practica (ver SIA):
  - Operaciones con palabras y lenguajes
  - Reconocimiento de tipos de gramáticas
  - Gramáticas de contexto libre: reconocimiento de palabras usando arboles sintácticos, determinación del lenguaje definido por una gramática, diseño de una gramática, determinación de ambigüedad y recursividad.

CONCLUSIONES

- Las operaciones que se pueden efectuar con palabras son similares a las operaciones que se pueden realizar con los lenguajes.
- Noam Chomsky definió la *Teoría de lenguajes formales* y la *noción de Gramática formal: G* =(N, T, P, S), clasificándolas en cuatro tipos.
- Las reglas de sintaxis de los lenguajes de Programacion están basadas en la Gramática del tipo 2 o de contexto libre.

BIBLIOGRAFIA

- ALFONSECA Enrique, ALFONSECA Manuel y MORIYON Roberto**. *Teoría de Autómatas y Lenguajes Formales***. 2007. Madrid. Editorial Mc Graw Hill.
- http://dehesa.unex.es/bitstream/10662/2367/1/978-84-691-6345-0.pdf



RECURSOS GRAFICOS

- Pixabay
- Pexels
- Icon-Icons