

Colección de normas estructuradas que especifican cómo se organiza la sintaxis de un idioma o lenguaje.

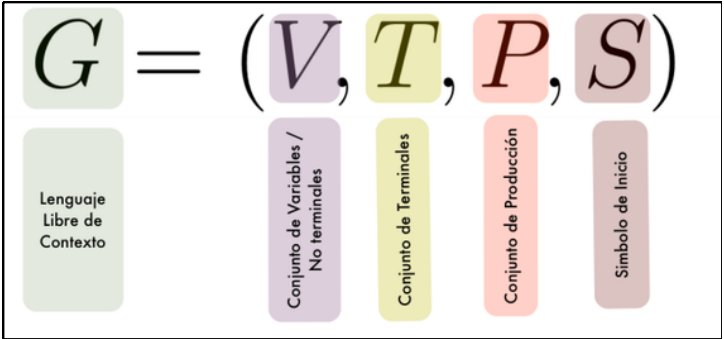
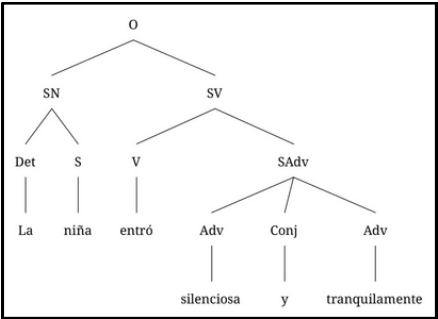
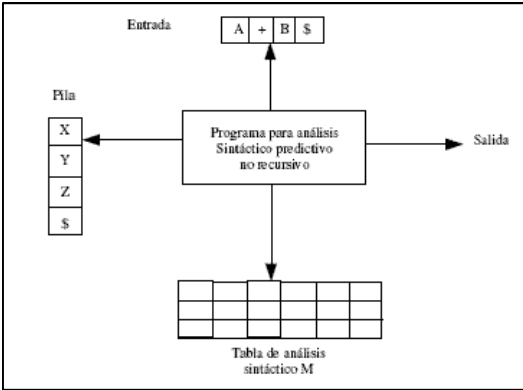


Diagrama arbóreo que muestra visualmente la organización interna de una oración.



Comprueba que las unidades léxicas se organicen en una construcción válida según la gramática.



Emplea una estructura de pila y un cuadro de referencia para dirigir el proceso analítico paso a paso.

Algoritmo 3.1: Construcción de una tabla de análisis sintáctico predictivo (Aho, Lam, Sethi, & Ullman, 2008, p. 224).

Entrada: La gramática G .

Salida: La tabla de análisis sintáctico M .

Método: Para cada producción $A \rightarrow \alpha$ de la gramática, hacer lo siguiente:

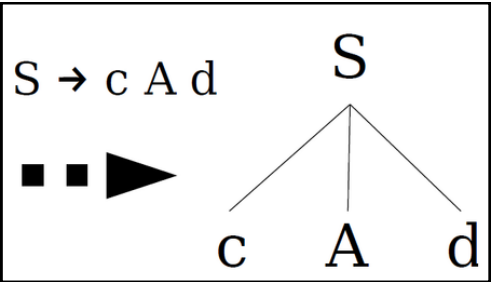
- Para cada terminal a en $\text{Primer}(A)$, agregar $A \rightarrow \alpha$ a $M[A, a]$.
- Si ϵ está en $\text{Primer}(a)$, entonces para cada terminal b en $\text{Sigui}(A)$, se agrega $A \rightarrow \alpha$ a $M[A, b]$. Si ϵ está en $\text{Primer}(a)$ y $\$$ se encuentra en $\text{Sigui}(A)$, se agrega $A \rightarrow \alpha$ a $M[A, \$]$ también.

Clase de gramática que posibilita un análisis anticipado claro y libre de ambigüedades.

GRAMATICAS LIBRES DE CONTEXTO

ARBOLES DE ANALISIS SINTACTICO

Genera el diagrama sintáctico iniciando en el nodo principal y avanzando hasta los nodos terminales.



ANALISIS SINTACTICO PREDICTIVO NO RECURSIVO

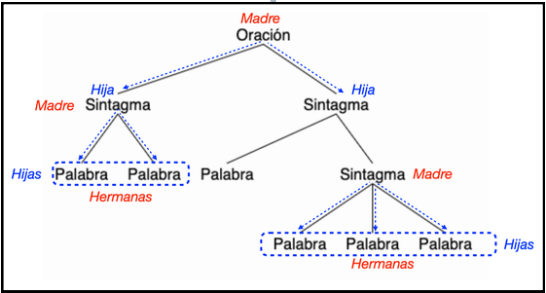
ANALISIS SINTACTICO

ANALISIS SINTACTICO DESCENDENTE

ANALISIS SINTACTICO DESCENDENTE RECURSIVO

GRAMATICAS LL

CONJUNTO PRIMERO Y SIGUIENTE



Mecanismos que permiten anticipar qué regla de producción debe utilizarse durante la interpretación.

Realiza el proceso de análisis sintáctico de arriba hacia abajo mediante llamadas recursivas en el programa.

Bosquejo de un procedimiento para un no terminal en un analizador sintáctico descendente

```
void A() {
    Elegir una producción A, A -> X1X2...Xk ;
    for( i = 1 a k ){
        if( X1 es un no terminal )
            llamar al procedimiento X1();
        else if( X1 es igual al símbolo de entrada actual a )
            avanzar la entrada hasta el siguiente símbolo;
        else /* rutina de error y retroceso()*/;
    }
}
```