

# Escuela Politécnica Nacional

Nombre: Fernando Elieco Huilca Villagomez

El análisis sintáctico es la segunda fase del compilador después del análisis léxico. Su objetivo es verificar si la secuencia de tokens generada por el analizador léxico forma una estructura gramatical válida según las reglas del lenguaje.

Las gramáticas libres de contexto que son las más utilizadas para describir la sintaxis de los lenguajes de programación. Una CFG consta de:

- Un conjunto de símbolos terminales (tokens).
- Un conjunto de símbolos no terminales (como expresiones o instrucciones).
- Un símbolo inicial.
- Un conjunto de producciones o reglas.

Estas gramáticas permiten construir árboles de derivación, que muestran como una cadena de tokens pueden generarse paso a paso desde un símbolo inicial.

Una gramática antigua puede ser problemática, ya que impide que el compilador interprete el programa de forma unívoca. Por ejemplo, esto ocurre en expresiones aritméticas sin una jerarquía clara de operadores.

Un árbol de sintaxis representa visualmente la estructura jerárquica del código.



Las hojas son los tokens del programa, los nodos internos son los tokens del programa.

Los nodos internos son los no terminales que representan construcciones sintácticas.

Los árboles ayudan a la verificación de estructuras, en la semántica posterior y la generación de código intermedio.

El análisis sintáctico descendente puede fallar si una gramática contiene recursión a la izquierda como en  $A \rightarrow A\alpha$ . Esto produce llamados recursos infinitos. Por eso se enseña a eliminar la recursión a la izquierda mediante transformaciones equivalentes que reescriben la gramática de forma no recursiva. Los analizadores sintácticos descendentes contruyen derivaciones de la forma de arriba hacia abajo, comenzando con el símbolo inicial e intentando generar la cadena de entrada.

El más común es el analizador predictivo LL(1) que utiliza un conjunto de predicción para decidir cuál producción usar sin necesidad de retroceso.

Fernando J. López



# Escuela Politécnica Nacional

Resumen Video: Registro en RAM: Crash Course Computer Science #6.

Tras construir una ALU, se plantea un problema: ¿de qué sirve calcular si no guardamos los resultados? Aquí entra la memoria. La RAM es la memoria principal, volátil y temporal usada mientras el equipo está encendido.

## El latch.

Se crea un circuito de retroalimentación usando compuertas OR y AND para almacenar un bit persistente combinando un OR y AND se forma un latch S-R (Set Reset), capaz de 'atraparse' en 0 o 1 guardando 1 bit hasta que se lo cambia.

Un Gated latch añade una señal write enable y una línea de datos usando un solo cable para escribir, agrupando 8 de estas celdas se forma un registro que guarda un número.

## Matrices y Multiplexores.

Para reducir cables, las celdas se organizan en una matriz de filas y columnas, con compuertas AND se selecciona una celda específica y se escribe, lee usando una única línea de datos.

Se emplean multiplexores para traducir direcciones binarias y seleccionar la celda correcta.

Jenandylf