

Escuela Politécnica Nacional

Nombre: Fernando Elceo Huila Villagomez Curso: GR2SW

El análisis sintáctico ascendente construye derivaciones de abajo hacia arriba, comenzando con los tokens y construyendo la estructura del programa hasta llegar al símbolo inicial.

Este enfoque es más poderoso que el descendente, ya que puede manejar una clase más amplia de gramáticas, incluyendo algunas ambiguas o que no son adecuadas para LL.

Método LR Left to right, Rightmost derivation in reverse.

El análisis LR es la técnica más potente y utilizada en compiladores reales. Funciona reconociendo derivaciones mas a la derecha pero de manera inversa y utiliza una pila para construir el árbol sintáctico. Hay varias variantes del análisis LR:

LR(0): el más básico, sin uso de lookahead

SLR (Simple LR): Usa los conjuntos FOLLOW para tomar decisiones precisas

LALR (LookAhead LR) Variante compacta de LR(1) que equilibra potencia y eficiencia.

Durante el análisis LR, el compilador realiza dos acciones principales

- Desplazar (shift) Avanzar en la entrada y apilar el token.
- Reducir (reduce) Reemplaza una secuencia de símbolos en la pila por un no terminal usando una regla gramatical.

Estas acciones están controladas por tablas LR, que determinan cual acción tomar según el estado actual y el símbolo de entrada.

Para construir las tablas del analizador, se utiliza un autómata LR, cuyo núcleo son los elementos LR(0). Estos elementos son reglas de la gramática con un punto que indica cuánto del lado derecho se ha reconocido.

Las transiciones entre elementos permiten construir un autómata determinista, que se usa para guiar el análisis sintáctico.

SLR (Simple LR): Utiliza conjuntos FOLLOW para resolver ambigüedades.

Es fácil de construir pero puede ser inadecuado para algunas gramáticas.

LALR (Lookahead LR): Mejora SLR al usar un análisis más preciso de lookahead. Es un método usado por herramientas como Yacc y Bison, ampliamente utilizadas en compiladores reales.

LALR combina la potencia de LR con el tamaño compacto de las tablas LR, lo que lo convierte en la opción más práctica para muchos lenguajes de programación.

Se destacan herramientas como Yacc y Bison, que permiten generar analizadores sintácticos automáticamente a partir de una gramática.

Estas herramientas se complementan con lex, el generador de analizadores léxicos, formando un pipeline completo de análisis para construir compiladores funcionales.

José María
García

Escuela Politécnica Nacional

Nombre: Fernando Elicer Huilca Villagómez

La CPU (unidad de procesamiento central) es el corazón que ejecuta programas mediante instrucciones como Suma, Resta, leer o escribir memoria.

Sus componentes principales son:

1. ALU: realiza operaciones aritméticas y lógicas.
2. Registros: Almacenamiento ultrarápido dentro de la CPU.
3. Unidad de control CU: detecta y decodifica instrucciones y controla señales para ALU, memoria y otros registros.
4. Memoria RAM: Donde se almacenan tanto programas como datos.

Reloj de la CPU (clock)

El reloj sincroniza cada fase de forma secuencial, la velocidad del reloj en hertz determina cuanto actos puede realizar por segundo.

Ejemplo: humano $\sim 0,03 \text{ Hz}$ un Intel 4004 (1971) alrededor de 740 kHz . los CPUs modernos alcanzan varios GHz con escalado dinámico de frecuencia.

La construcción de la CPU es un ejemplo clásico de la abstracción por capas; ya que orquesta el flujo, lee instrucciones, las traduce, ejecuta operaciones matemáticas o de memoria y repite todo sincronizado por el reloj.

Fernando Huilca