

## Árboles de análisis

Un **árbol de análisis (parse tree)** muestra gráficamente las derivaciones.

- Cada nodo interno: LHS de una producción.
  - Hijos: RHS de esa producción.
  - Hojas (de izquierda a derecha): **yield** (rendimiento) del árbol.
  - Parsear una cadena: verificar si pertenece al lenguaje de la gramática.
  - Derivación  $\leftrightarrow$  Reducción.
- 

## Ambigüedad

Una gramática es **ambigua** si existen **dos o más árboles de análisis** para la misma cadena.

- Ejemplo dado no ambiguo:  $1+2+3$ .
  - Ejemplo de demostración incorrecta:  $S \rightarrow A \ B, \ A \rightarrow x, \ B \rightarrow x$ .  
Tiene dos derivaciones, pero el mismo árbol.
- 

## Asociatividad

La gramática dada tiene **asociatividad por la izquierda**.

- Ejemplo:  $8-8-8 = -8$ .
  - Para asociatividad por la derecha (como exponentiación), se cambia a:
    - lista  $\rightarrow$  digito + lista
    - lista  $\rightarrow$  digito - lista
- 

## Precedencia

La precedencia se maneja con no terminales adicionales:

```
expr   → expr + term | expr - term | term
term   → term * factor | term / factor | factor
factor → digit | ( expr )
digit  → 0 | 1 | ... | 9
```

Permite operaciones con precedencia normal y paréntesis opcionales.

---

## Declaraciones (Statements)

Ejemplo de gramática con sentencias:

```
STMT → id := EXPR
      | if EXPR then STMT
      | if EXPR then STMT else STMT
      | while EXPR do STMT
      | begin OPT-STMITS end
OPT-STMITS → STMT-LIST | ε
STMT-LIST → STMT-LIST ; STMT | STMT
```

Observaciones:

1. Uso de mayúsculas solo por claridad.
2. OPT-STMITS: bloques vacíos opcionales.
3.  $\epsilon$ : cadena vacía.
4. Las producciones con  $\epsilon$  complican el análisis.
5. Algunos lenguajes no permiten bloques vacíos.
6. La gramática es ambigua.
7. Problema del **dangling else**.

El "problema del else colgante" (dangling else) ocurre cuando hay varias sentencias if anidadas y una única sentencia else que podría pertenecer lógicamente a cualquiera de ellas, generando una ambigüedad en la ejecución del código. El problema surge porque el else se asocia a uno de los if que la preceden, y el lenguaje de programación no siempre tiene una regla clara para determinar a cuál if debe asignarse, llevando a errores lógicos o a un comportamiento no deseado del programa

---