

Universidad Autónoma del Estado de Hidalgo  
Escuela Superior de Huejutla  
Licenciatura en Ciencias Computacionales

# Gramáticas Ambiguas



Asignatura: Autómatas y Compiladores  
Catedrático: Gonzalo Hernández Hernández  
Periodo: Julio-Diciembre 2017

# Tema: Gramáticas Ambiguas

**Resumen:** Las gramáticas libres de contexto representan el lenguaje formal a partir del cual se construyen los analizadores sintácticos. Muchas de las veces, estas gramáticas deben ser reescritas hasta cumplir con ciertas características, siendo la ambigüedad uno de los problemas mas comunes.

**Palabras clave:** analizador sintáctico, componente léxico, gramática libre de contexto.

**Abstract:** Context-free grammars represent the formal language from which syntactic parsers are built. Many times, these grammars must be rewritten to meet certain characteristics, ambiguity being one of the most common problems.

**Keywords:** parser, token, context-free grammar.

# Gramáticas Ambiguas

---

Cuando una gramática libre del contexto genera mas de una estructura para alguna o algunas cadenas del lenguajes, se dice que se trata de un gramática ambigua.

Por ejemplo: Consideremos la sentencia **id=id+id\*id;**

**G=(VT, VN, S, P)**

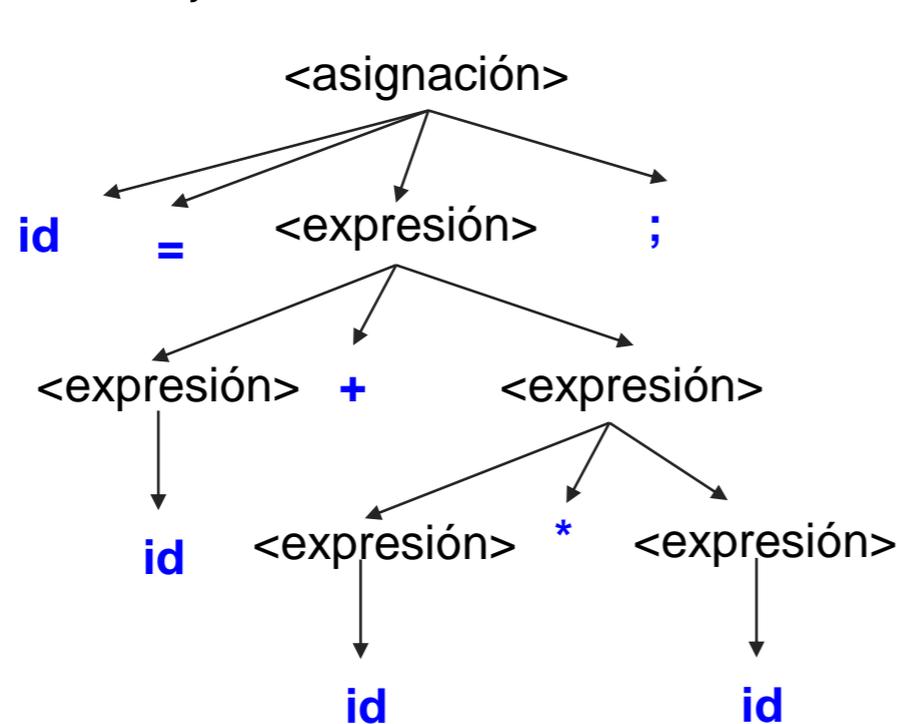
VT={**id**, (, ), +, \*, =, ; }

VN={<asignación>, <expresión>}

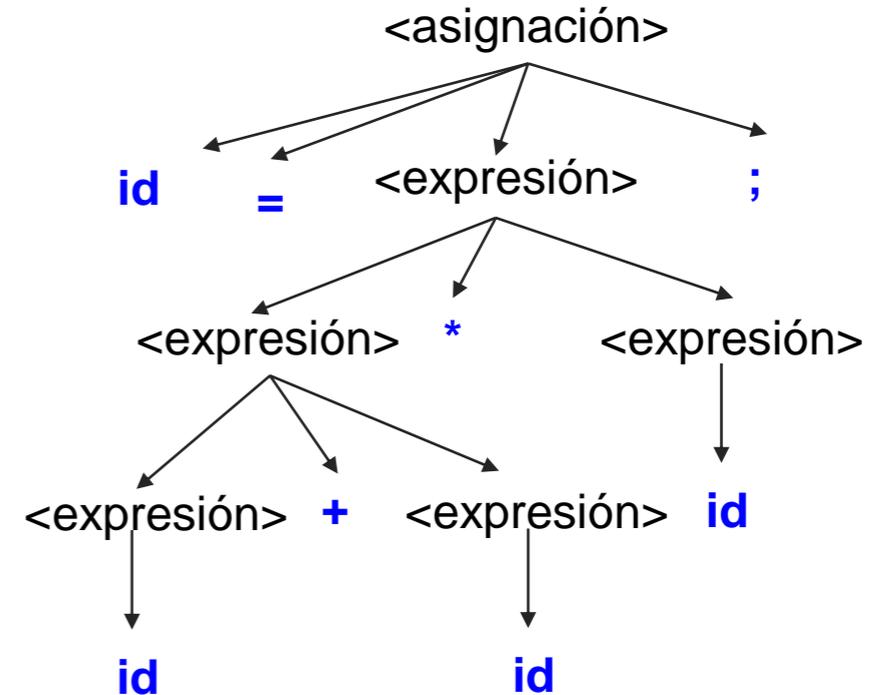
S={<asignación>}

P={

- ① <asignación> → **id** = <expresión> ;
  - ② <expresión> → <expresión> + <expresión>
  - ③ <expresión> → <expresión> \* <expresión>
  - ④ <expresión> → ( <expresión> )
  - ⑤ <expresión> → **id**
- }



a)



b)

Dos árboles de derivación con el mismo resultado

En el ejemplo anterior se puede observar que en la derivación a), la primera <expresión> es sustituida por <expresión>+<expresión>, mientras que en la derivación b) la primera <expresión> es sustituida por <expresión>\*<expresión> .

La diferencia entre ellas dos afecta a la estructura de las expresiones, ya que en a) primero se multiplica y después se suma, mientras que en b) primero se suma y luego se multiplica.

Por jerarquía de operaciones sabemos que a) es correcta y b) incorrecta, con lo cual vemos que la gramática no es adecuada debido a que no proporciona una estructura única.

Es necesario reescribir la gramática  $G$  de manera que solo proporcione estructuras correctas.

¿Existe un algoritmo para detectar la ambigüedad de una GLC?

¿Existe un algoritmo para eliminar la ambigüedad de las GLC?

# Bibliografía

---

Alfonseca, Manuel (2006). *Compiladores e intérpretes: teoría y práctica*. Pearson Educación, S.A., Madrid.

Hopcroft, Jhon, Motwani, Rajeev y Ullman, Jeffrey (2007). *Introducción a la teoría de autómatas, lenguajes y computación*. Addison-Wesley, Pearson Educación.