

Análise Sintática

O Papel do Analisador Sintático.

- Receber uma cadeia de *tokens* do analisador léxico e verificar se essa sequência de *tokens* pertence a linguagem gerada pela gramática.
- Deve emitir mensagens para todos erros de sintaxe encontrados; e
- Recuperar-se desses erros a fim de continuar analisando o programa fonte.
- Para o programa livre de erros sintáticos, o *parser* deve construir uma árvore de derivação

Métodos de Análise Sintática

Existem 3 estratégias de análise sintática:

- Universal
- Descendente
- Ascendente

Algoritmos da estratégia universal, como o algoritmo de Cocke-Younger-Kasami e o algoritmo de Earley, apesar de poderem analisar qualquer gramática são muito ineficientes para serem usados em compiladores de produção.

Os métodos descendentes e ascendentes mais eficientes funcionam apenas para subclasses de gramáticas, particularmente as gramáticas LL e LR que são capazes de representar a maioria das construções sintáticas das linguagens de programação moderna.

Gramáticas Livre de Contexto

Uma gramática livre de contexto consiste em terminais, não-terminais, um símbolo inicial e produções, onde:

- **Terminais:** são os símbolos básicos apartir dos quais as cadeias são formadas. O termo “nome de *token*” é um sinônimo para terminal.(usaremos *token*).
- **Não-Terminais:** são variáveis sintáticas que representam conjuntos de cadeias. Impõem uma estrutura hierárquica sobre a linguagem.
- **Símbolo inicial:** é um símbolo não-terminal e o conjunto de cadeias formado apartir dele é a linguagem.

Gramática Livre de Contexto

• **Produções:** especificam como terminais e não-terminais podem ser combinados para formar as cadeias. Cada produção consiste em:

- Um não-terminal chamado de **cabeça** ou **lado esquerdo da produção**.
- O símbolo \rightarrow (seta) .
- Um **corpo** ou **lado direito da produção** que consiste em zero ou mais terminais e não-terminais. Os componentes do corpo descrevem como as cadeias do não-terminal do lado direito da produção podem ser combinados.

Convenção de notação

Como serão apresentados os componentes das gramáticas livre de contexto:

• Terminais

- letras minúsculas do início do alfabeto como a, b, c
- Símbolos operadores como +, *, assim por diante
- Símbolos de pontuação como parênteses, vírgula e assim por diante
- Os dígitos 0,1,...,9
- Cadeias em **negrito** como **if** ou **id** cada um representando um único símbolo terminal

Convenção de Notação

• Não-Terminais

- Letras maiúsculas do início do alfabeto como A, B, C
- A letra S, quando aparecer, é o símbolo inicial da gramática
- Nomes em minúsculas e itálico como *expr*, *stmt*
- Letras gregas minúsculas representam **cadeias de símbolos da gramática**
- O lado esquerdo da primeira produção é o **símbolo inicial da gramática**, exceto quando definido explicitamente
- **As várias produções de um não-terminal como:**
 - $E \rightarrow E + T$
 - $E \rightarrow E - T$
 - $E \rightarrow T$podem ser denotadas por :
 - $E \rightarrow E + T \mid E - T \mid T$

Exemplo de gramática

Identifique os componentes da gramática abaixo.

$$E \rightarrow E + T$$

$$E \rightarrow E - T$$

$$E \rightarrow T$$

$$T \rightarrow T * F$$

$$T \rightarrow T / F$$

$$T \rightarrow F$$

$$F \rightarrow (E)$$

$$F \rightarrow \text{id}$$

$$F \rightarrow \text{num}$$

O que essa gramática gera ?

Derivação

As produções determinam regras de substituição do não-terminal cabeça, ou seja, o lado esquerdo da produção é substituído pelo lado direito. Cada substituição dessa é chamada de **Passo de Derivação**.

Uma **derivação** consiste em partindo do símbolo inicial da gramática e fazendo consecutivos passos de derivação resultar em uma cadeia formada apenas por símbolos terminais.

Exemplo de Derivação

$E \rightarrow E + T$

$E \rightarrow E - T$

$E \rightarrow T$

$T \rightarrow T * F$

$T \rightarrow T / F$

$T \rightarrow F$

$F \rightarrow (E)$

$F \rightarrow id$

$F \rightarrow num$

E (símbolo inicial da gramática)

$\underline{E} \rightarrow E + T$

(E deriva em um passo $E+T$)

$E \rightarrow \underline{E} + T \rightarrow T + T \rightarrow F + T \rightarrow num + T \rightarrow$
 $\rightarrow num + F \rightarrow num + id$

(E deriva em 1 ou mais passos $num + id$)

$E \rightarrow^* num + id$

(E deriva em 1 ou mais passos $num + id$)

Derivações

As cadeias formadas durante a derivação são chamadas de **formas sentenciais**.

- **Derivação mais à esquerda**

- É aquela que em todos os passos de derivação sempre substitui(deriva) o não-terminal mais à esquerda das formas sentenciais.

- **Derivação mais à direita**

- É aquela que em todos os passos de derivação sempre substitui(deriva) o não-terminal mais à direita das formas sentenciais.

Árvores de Derivação

Uma árvore de derivação é uma representação gráfica de uma derivação que filtra a ordem na qual as produções são aplicadas para substituir não-terminais.

Exemplo de árvore de derivação no quadro negro.