

Práctica 8: Gramáticas de atributos

Versión del 18 de marzo de 2024

Ejercicio 1. Determinar el conjunto de cadenas generadas por la siguiente gramática de atributos:

$$G = \langle \{S, X, Y, Z\}, \{x, y, z\}, P, S \rangle.$$

size es un atributo sintetizado de *X* y un atributo heredado de *Y* y *Z*, y *P* es:

$$\begin{array}{ll} S \rightarrow XYZ & \{ Y.size = X.size ; Z.size = X.size \} \\ X \rightarrow x & \{ X.size = 1 \} \\ & | X_1x \quad \{ X.size = X_2.size + 1 \} \\ Y \rightarrow y & \{ \text{CONDICIÓN: } Y.size = 1 \} \\ & | Y_1y \quad \{ Y_2.size = Y.size - 1 \} \\ Z \rightarrow z & \{ \text{CONDICIÓN: } Z.size = 1 \} \\ & | Z_1z \quad \{ Z_2.size = Z.size - 1 \} \end{array}$$

Ejercicio 2. Sin cambiar el lenguaje generado, modificar la gramática del ejercicio 1 para que sea una gramática S-atribuida.

Ejercicio 3.

a. Dar una gramática de atributos que genere el lenguaje

$$\mathcal{L} = \{a^n(bc^n)^m \mid n \geq 2, m \geq 0\}.$$

b. Construir un árbol atribuido para la cadena *aabccbcc*.

Ejercicio 4.

a. Dar una gramática de atributos que genere el lenguaje

$$\mathcal{L} = \{\alpha \mid \alpha \in (a|b|c)^*d^* \wedge |\alpha|_a = |\alpha|_b = |\alpha|_c = |\alpha|_d\}.$$

b. Construir árboles atribuidos para las cadenas *aabcbcd* y *bcbcd*.

Ejercicio 5. Dada la siguiente gramática que genera expresiones aritméticas de suma y producto, definir un atributo *exp* de tipo string que sintetice la expresión generada pero sin paréntesis redundantes.

$$G = \langle \{E, T, F\}, \{+, -, *, \text{id}, (,)\}, P, S \rangle$$

$$\begin{array}{ll} E \rightarrow E + T & | T \\ \text{con } P: & T \rightarrow T * F & | F \\ & F \rightarrow \text{id} & | (E) \end{array}$$

Ejercicio 6. Dada la gramática $G = \langle \{E\}, \{+, *, \mathbf{var}, \mathbf{const}, (,)\}, P, E \rangle$, con P :

$$E \rightarrow E + E \mid E * E \mid \mathbf{var} \mid \mathbf{const} \mid (E)$$

- a. Definir una gramática de atributos que sintetice la expresión original, pero reemplazando las subexpresiones en las que solo aparezcan constantes por su resultado. Por ejemplo:

Expresión original	Expresión sintetizada
$a + 3 * 4$	$a + 12$
$(3 + 2) * 4 + a + 2$	$20 + a + 2$
$(3 + a) * 2 + 2 * a$	$(3 + a) * 2 + 2 * a$

El terminal **var** tiene un atributo *text* de tipo string. El terminal **const** tiene un atributo *val* de tipo entero. Se cuenta con la función *aString*: entero \rightarrow string, que convierte un entero en su representación decimal.

- b. Definir una gramática de atributos que sintetice la expresión original, pero aplicando las siguientes reglas de simplificación:

Expresión original	Expresión simplificada
$0 + E$	E
$E + 0$	E
$1 * E$	E
$E * 1$	E
$0 * E$	0
$E * 0$	0

Los terminales **var** y **const** tienen un atributo *text* de tipo string.

Nota: ignorar el hecho de que la gramática dada es ambigua. Asumir que el árbol de derivación se armará siguiendo las asociatividades y precedencias usuales.

Ejercicio 7. Sea *val* un atributo sintetizado que da el valor binario generado por S en la siguiente gramática.

$$G = \langle \{S, L, B\}, \{0, 1, .\}, P, S \rangle$$

$$\begin{aligned} S &\rightarrow L.L \\ \text{con } P: \quad L &\rightarrow LB \mid B \\ B &\rightarrow 0 \mid 1 \end{aligned}$$

Por ejemplo, si la entrada es 101.101, $S.val = 5,625$.

- a. Utilizar atributos sintetizados para calcular $S.val$.
- b. Calcular $S.val$ con una gramática de atributos en la cual el único atributo sintetizado de B sea *cont*, que da la contribución del bit generado por B al valor final. Por ejemplo, la contribución del primer bit en 101.101 es 4, la del último es 0,125.

Ejercicio 8. Escribir una gramática de atributos que acepte cadenas sobre el alfabeto:

$$\{ (,), +, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F \}$$

cuyo formato sea como el del siguiente ejemplo:

$$(20, 5) + (F, 16) - (110, 2)$$

Las cadenas se interpretan de la siguiente manera: cada par ordenado representa un número natural. El segundo elemento (escrito en base 10) indica la base y el primer elemento es la representación del número en esa base. En el ejemplo, la cadena debe ser interpretada como $10 + 15 - 6 = 19$.

La gramática debe sintetizar el valor de la expresión en un atributo del símbolo inicial. Además, se deben rechazar las cadenas en las que algún par no represente un número válido. Por ejemplo, la cadena $(124, 3)$ debe ser rechazada porque “4” no puede aparecer en un número escrito en base 3.