

# Trabajo práctico N° 1

Abril 2025

Estudiante Emanuel Nicolás Herrador

## Ejercicio 1

Consideramos la gramática dada por:

$$\langle \text{intexp} \rangle ::= 0 \mid 1 \mid 2 \mid \dots \mid - \langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle + \langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle * \langle \text{intexp} \rangle$$

Con esta en consideración, es trivial nota que las frases ambiguas son  $-7 * 2$  y  $27 + 3 + -7$ . La primera por el orden en que se debe aplicar el menos y el por, y la segunda por cómo debemos asociar la suma.

Para resolver esta ambigüedad, debe usarse precedencia de la multiplicación sobre la negación en el primer caso, y en el segundo, asociatividad.

## Ejercicio 2

Para cada caso, queremos ver qué símbolos pertenecen al lenguaje objeto y cuáles al metalenguaje.

**Item A:**  $\llbracket 0 \rrbracket = 0$

El 0 de la izquierda pertenece al lenguaje objeto y el 0 de la derecha al lenguaje de los enteros.

**Item B:**  $\llbracket -e \rrbracket = -\llbracket e \rrbracket$

El  $-$  de la izquierda pertenece al lenguaje objeto y el de la derecha al lenguaje de los enteros. El  $e$  es una metavariante del lenguaje.

**Item C:**  $\llbracket e + f \rrbracket = \llbracket e \rrbracket + \llbracket f \rrbracket$

El  $+$  de la izquierda pertenece al lenguaje objeto y el de la derecha al lenguaje de los enteros. Tanto  $e$  como  $f$  son metavariante del lenguaje.

## Ejercicio 3

Dada la gramática de expresiones enteras considerada en el teórico, es decir la del primer ejercicio; y dadas las ecuaciones semánticas dirigidas por sintaxis definidas, entonces podemos resolver la semántica de ambas expresiones.

**Item A:**  $\llbracket 0 * (5 * (7 + 2)) \rrbracket$

$$\begin{aligned} \llbracket 0 * (5 * (7 + 2)) \rrbracket &= \llbracket 0 \rrbracket * \llbracket 5 * (7 + 2) \rrbracket \\ &= 0 * (\llbracket 5 \rrbracket * \llbracket 7 + 2 \rrbracket) \\ &= 0 * (5 * (\llbracket 7 \rrbracket + \llbracket 2 \rrbracket)) \\ &= 0 * (5 * (7 + 2)) \\ &= 0 \end{aligned}$$

**Item B:**  $\llbracket a * a + a * b + b * a + b * b \rrbracket$ 

Considero asociatividad a izquierda:

$$\begin{aligned}
\llbracket a * a + a * b + b * a + b * b \rrbracket &= \llbracket a * a \rrbracket + \llbracket a * b + b * a + b * b \rrbracket \\
&= \llbracket a \rrbracket * \llbracket a \rrbracket + \llbracket a * b \rrbracket + \llbracket b * a + b * b \rrbracket \\
&= a * a + \llbracket a \rrbracket * \llbracket b \rrbracket + \llbracket b * a \rrbracket + \llbracket b * b \rrbracket \\
&= a * a + a * b + \llbracket b \rrbracket * \llbracket a \rrbracket + \llbracket b \rrbracket * \llbracket b \rrbracket \\
&= a * a + a * b + b * a + b * b \\
&= a^2 + 2ab + b^2 \\
&= (a + b)^2
\end{aligned}$$

**Ejercicio 4**

Se considera la siguiente gramática y ecuaciones semánticas con precedencia habitual entre  $*$  y  $+$ :

$$\begin{aligned}
\langle \text{intexp} \rangle &::= 1 \mid \langle \text{intexp} \rangle + \langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle * \langle \text{intexp} \rangle \\
\llbracket 1 \rrbracket &= 1 \\
\llbracket e + e' \rrbracket &= \llbracket e \rrbracket + \llbracket e' \rrbracket \\
\llbracket e * 1 \rrbracket &= \llbracket e \rrbracket \\
\llbracket e * (e' + 1) \rrbracket &= \llbracket e * e' \rrbracket + \llbracket e \rrbracket
\end{aligned}$$

**Item A**

El conjunto de ecuaciones no define una función semántica porque no es total ya que no tiene definición para  $\llbracket 1 * (1 * 1) \rrbracket$ .

**Item B**

Se garantiza que para las frases con significado este sea único, dado que para toda frase representada por las ecuaciones semánticas, solo una de ellas se aplica a la expresión (obviando recursividad, claro). Es decir, no existe ninguna producción de la gramática que pueda tomar dos “caminos” en las definiciones.

**Item C**

No es dirigido por sintaxis el conjunto de ecuaciones porque no hay una ecuación por cada producción de la gramática abstracta.

**Item D**

No es composicional y se puede ver por contraejemplo. Notemos que  $\llbracket 1 * 1 \rrbracket = 1$  y que  $\llbracket 1 \rrbracket = 1$  pero  $\llbracket 1 * (1 * 1) \rrbracket$  no está definida. Luego,  $\llbracket 1 * 1 \rrbracket$  y  $\llbracket 1 * (1 * 1) \rrbracket$  no tienen iguales significados.

**Ejercicio 5**

Consideremos la siguiente gramática abstracta para números binarios:

$$\langle \text{bin} \rangle ::= 0 \mid 1 \mid 0 \langle \text{bin} \rangle \mid 1 \langle \text{bin} \rangle$$

**Item A**

Dada la función:

$$\begin{aligned}
\llbracket \_ \rrbracket^s &: \langle \text{bin} \rangle \rightarrow \mathbb{N} \\
\llbracket \alpha_0 \dots \alpha_{n-1} \rrbracket^s &= \sum_{i=1}^n \alpha_{i-1} 2^{n-i}
\end{aligned}$$

No es dirigida por sintaxis porque no está definido puramente en base a sus subfrases inmediatas, sino que usa el largo de la expresión para ello. No es composicional por contraejemplo dado que  $\llbracket 01 \rrbracket = 0 * 2^1 + 1 * 2^0 = 1$  y  $\llbracket 11 \rrbracket = 1 * 2^1 + 1 * 2^0 = 3$  pero  $\llbracket 101 \rrbracket = 1 * 2^2 + 0 * 2^1 + 1 * 2^0 \neq \llbracket 11 \rrbracket$ .

## Item B

Ahora, si consideramos la definición dada por  $\llbracket \alpha \alpha_1 \dots \alpha_{n-1} \rrbracket^i = \alpha 2^{n-1} + \llbracket \alpha_1 \dots \alpha_{n-1} \rrbracket^i$ , tenemos que tampoco es dirigida por sintaxis porque no se define exclusivamente de los significados de sus subfrases inmediatas sino que usa también el largo. Otra forma de mostrarlo es viendo que no es composicional dado que  $\llbracket 11 \rrbracket \neq \llbracket 101 \rrbracket$  y, por ende, no puede ser dirigido por sintaxis.

## Item C

No se puede dar un conjunto de ecuaciones dirigidas por sintaxis para esta función dado que siempre va a depender del largo del número binario, lo cual no es una subfrase inmediata.

## Ejercicio 6

No, el argumento no es correcto porque lo importante no es que la frase completa cambie de denotación, sino que cambia su significado al hacer el reemplazo. Es decir, el hecho de que cambie de denotación no es lo que genera que no sea composicional, sino que esto se produce porque el significado es otro.

Esto se puede ver porque  $\llbracket 01 \rrbracket = \llbracket 1 \rrbracket$  aunque se denoten distinto.

## Ejercicio 7

Como bien se dijo en ejercicios anteriores, no se puede definir una función dirigida por sintaxis para la semántica de números binarios si el largo de este no podemos considerarlo de alguna forma. Por ello, se tendrá ahora en cuenta una función  $\llbracket \_ \rrbracket^p : \langle \text{bin} \rangle \rightarrow \mathbb{N} \times \mathbb{N}$  tal que  $\pi_1 \llbracket e \rrbracket^p = \llbracket e \rrbracket^s$ . Aquí, la función es:

$$\begin{aligned}\llbracket 0 \rrbracket &= (0, 1) \\ \llbracket 1 \rrbracket &= (1, 1) \\ \llbracket 0e \rrbracket &= (\pi_1 \llbracket e \rrbracket, 1 + \pi_2 \llbracket e \rrbracket) \\ \llbracket 1e \rrbracket &= (2^{\pi_2 \llbracket e \rrbracket} + \pi_1 \llbracket e \rrbracket, 1 + \pi_2 \llbracket e \rrbracket)\end{aligned}$$

Claramente es dirigido por sintaxis y es trivial notar que la proyección de la primer coordenada se corresponde al significado de las funciones mostradas en el ejercicio 5.

## Ejercicio 8

### Item A

Para extender el lenguaje de las expresiones aritméticas agregando la división entera, si queremos que siga siendo dirigido por sintaxis y solo agregar la división a la gramática (i.e., ninguna otra constante), entonces podemos darle un significado arbitrario. Es decir, podemos decir que todo número dividido por 0 es igual a 0.

De este modo, consideramos:

$$\begin{aligned}\langle \text{intexp} \rangle &::= 0 \mid 1 \mid \dots \mid - \langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle + \langle \text{intexp} \rangle \\ &\quad \mid \langle \text{intexp} \rangle * \langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle \div \langle \text{intexp} \rangle \\ \llbracket [n] \rrbracket &= n \\ \llbracket -e \rrbracket &= -\llbracket e \rrbracket \\ \llbracket e + e' \rrbracket &= \llbracket e \rrbracket + \llbracket e' \rrbracket \\ \llbracket e * e' \rrbracket &= \llbracket e \rrbracket * \llbracket e' \rrbracket \\ \llbracket e \div e' \rrbracket &= \begin{cases} 0 & \text{si } \llbracket e' \rrbracket = 0 \\ \llbracket e \rrbracket \div \llbracket e' \rrbracket & \text{cc.} \end{cases}\end{aligned}$$

### Item B

$$\begin{aligned}\llbracket 2 \div 0 \rrbracket &= \begin{cases} 0 & \text{si } \llbracket 0 \rrbracket = 0 \\ 2 \div 0 & \text{cc.} \end{cases} \\ &= \begin{cases} 0 & \text{si } 0 = 0 \\ 2 \div 0 & \text{cc.} \end{cases} \\ &= 0\end{aligned}$$

## Ejercicio 9

Ahora, la idea va a ser extender el lenguaje de las expresiones aritméticas agregando la división entera, pero también la constante error para el caso de la división por 0.

### Item A

Vamos a considerar:

$$\langle \text{intexp} \rangle ::= \text{error} \mid 0 \mid 1 \mid \dots \mid -\langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle + \langle \text{intexp} \rangle \\ \mid \langle \text{intexp} \rangle * \langle \text{intexp} \rangle \mid \langle \text{intexp} \rangle \div \langle \text{intexp} \rangle$$

$$\llbracket [n] \rrbracket = n$$

$$\llbracket -e \rrbracket = \begin{cases} \text{error} & \text{si } \llbracket e \rrbracket = \text{error} \\ -\llbracket e \rrbracket & \text{cc.} \end{cases}$$

$$\llbracket e + e' \rrbracket = \begin{cases} \text{error} & \text{si } \llbracket e \rrbracket = \text{error} \vee \llbracket e' \rrbracket = \text{error} \\ \llbracket e \rrbracket + \llbracket e' \rrbracket & \text{cc.} \end{cases}$$

$$\llbracket e * e' \rrbracket = \begin{cases} \text{error} & \text{si } \llbracket e \rrbracket = \text{error} \vee \llbracket e' \rrbracket = \text{error} \\ \llbracket e \rrbracket * \llbracket e' \rrbracket & \text{cc.} \end{cases}$$

$$\llbracket e \div e' \rrbracket = \begin{cases} \text{error} & \text{si } \llbracket e \rrbracket = \text{error} \vee \llbracket e' \rrbracket \in \{\text{error}, 0\} \\ \llbracket e \rrbracket \div \llbracket e' \rrbracket & \text{cc.} \end{cases}$$

### Item B

Para calcular  $\llbracket (7 + (2 \div (5 * 0))) \rrbracket$ , sería bastante ilegible hacer el “paso por paso” por cómo se definieron las ecuaciones semánticas. Por ello mismo, se deja sin realizar porque es algo trivial de hacer también.