



UNIVERSIDAD SIMÓN BOLÍVAR  
Departamento de Computación y Estudios Tecnológicos  
Traductores e Interpretadores

---

## Entrega 2: Analizador Sintáctico para BRAINIAC

---

*Autores:*  
Patricia Wilthew 09-10910  
Leopoldo Pimentel 06-40095

13 de diciembre de 2013

# Índice

1. Revisión Práctico-Teórica	2
------------------------------	---

# 1. Revisión Práctico-Teórica

1. Sea G1 la gramática  $\{ \{ Expr \} ; \{ + , NUM \} ; P1 ; Expr \}$ , con P1 compuesto por las siguientes producciones:

$Expr \rightarrow Expr + Expr$

$Expr \rightarrow NUM$

(a) Muestre que la frase  $NUM + NUM + NUM$  de G1 es ambigua.

(b) Dé una gramática no ambigua Izq(G1) que genere el mismo lenguaje que G1 y que asocie las expresiones aritméticas generadas hacia la izquierda. Dé también una gramática Der(G1) con las mismas características pero que asocie hacia la derecha.

(c) ¿Importa si se asocian las expresiones hacia la izquierda o hacia la derecha? Considere el caso del operador "-" o el caso del operador "/".

**Respuestas:**

(a) ***Leftmost:***

$\underline{Expr} \Rightarrow \underline{Expr} + Expr \Rightarrow \underline{Expr} + Expr + Expr \Rightarrow NUM + \underline{Expr} + Expr \Rightarrow$   
 $NUM + NUM + \underline{Expr} \Rightarrow NUM + NUM + NUM$

***Rightmost:***

$\underline{Expr} \Rightarrow Expr + \underline{Expr} \Rightarrow Expr + Expr + \underline{Expr} \Rightarrow Expr + \underline{Expr} + NUM \Rightarrow$   
 $\underline{Expr} + NUM + NUM \Rightarrow NUM + NUM + NUM$

Con estas dos maneras de derivar una frase, se muestra que ésta es ambigua.

(b) Sin ambigüedad:

Izq(G1):

$Expr \rightarrow Expr + NUM$

$Expr \rightarrow NUM$

Der(G1):

$Expr \rightarrow NUM + Expr$

$Expr \rightarrow NUM$

(c) Sí importaría. En el caso del operador "/", no es lo mismo  $(1 + 2)/3$  que resulta en 1, que  $1 + (2/3)$  que resulta en  $1,6$ . En el caso del operador "-", no es lo mismo  $3 - (5 - 2)$  que resulta en 0, que  $(3 - 5) - 2$  que resulta en  $-4$ .

2. En Brainiac la secuenciación, o composición secuencial, es un operador binario infijo sobre instrucciones denotado por el símbolo ";" . Suponga que para el manejo de este operador se decide utilizar la gramática G2 definida como  $\{ \{ Instr \} , \{ ; , IS \} , P2 , Instr \}$  , con P2 compuesto por las siguientes producciones:

$Instr \rightarrow Instr ; Instr$

$Instr \rightarrow IS$

Por conveniencia, momentáneamente se ignora al resto de los constructores de instrucciones compuestas de Brainiac, y se simplifica a las instrucciones simples con el símbolo terminal IS .

(a) ¿Presenta G2 los mismos problemas de ambigüedad que G1 ? ¿Cuáles son las únicas frases no ambiguas de G2?

(b) ¿Importa si la ambigüedad se resuelve con asociación hacia la izquierda o hacia la derecha?

(c) Dé una derivación leftmost y una derivación rightmost de G2 para la frase IS ; IS ; IS.

**Respuestas:**

(a) Sí presenta los mismos problemas de ambigüedad. La única frase no ambigua de G2 es IS.

(b) En este caso no, porque el operador de secuenciación no se utiliza para asociar sino para separar.

(c) **Leftmost:**

$\underline{Instr} \Rightarrow \underline{Instr}; Instr \Rightarrow \underline{Instr}; Instr; Instr \Rightarrow IS; \underline{Instr}; Instr \Rightarrow IS; IS; \underline{Instr} \Rightarrow IS; IS; IS$

**Rightmost:**

$\underline{Instr} \Rightarrow Instr; \underline{Instr} \Rightarrow Instr; Instr; \underline{Instr} \Rightarrow Instr; Instr; IS \Rightarrow Instr; IS; IS \Rightarrow IS; IS; IS$

3. Consideremos ahora los constructores de instrucciones condicionales de Brainiac, pero ignorando momentáneamente el uso que la sintaxis de éstos hace de los **done**. Sea G3 la gramática  $\{ \{ \text{Instr} \} , \{ ; , \text{IF} , \text{THEN} , \text{ELSE} , \text{Bool} , \text{IS} \} , \text{P3} , \text{Instr} \} ,$  con P3 compuesto por:

`Instr -> Instr ; Instr`

`Instr -> IF Bool THEN Instr`

`Instr -> IF Bool THEN Instr ELSE Instr`

`Instr -> IS`

(a) Note que G3 mantiene el mismo problema de G2 , i.e. frases con más de una ocurrencia de ";" son ambiguas. Más aún, G3 tiene otros problemas pues frases con ninguna o sólo una ocurrencia de ";" también pueden ser ambiguas. Sea f la frase "IF Bool THEN IS ; IS". Muestre que f es una frase ambigua de G3.

(b) Dé una frase g de G3 sin ocurrencias de ";" que sea ambigua, y muestre que lo es.

(c) En lenguajes como Java, C y C++ se hace uso de los símbolos { y } (begin y end en Pascal) para diferenciar las dos posibles interpretaciones de la frase f. Lo mismo ocurre con la frase g. ¿Cómo se escribirían las dos interpretaciones de f y las dos interpretaciones de g usando las llaves como separadores?

(d) En Brainiac, que utiliza los terminadores **done** en la sintaxis de sus condicionales, ¿cómo se escribirían las dos interpretaciones de f y las dos interpretaciones de g en Brainiac?

### Respuestas:

(a) Derivaciones:

#### ***Leftmost:***

$\underline{Instr} \Rightarrow \underline{Instr}; Instr \Rightarrow IF Bool THEN \underline{Instr}; Instr \Rightarrow IF Bool THEN IS ; \underline{Instr}$   
 $\Rightarrow IF Bool THEN IS ; IS$

#### ***Rightmost:***

$\underline{Instr} \Rightarrow Instr; \underline{Instr} \Rightarrow IF Bool THEN Instr ; \underline{Instr} \Rightarrow IF Bool THEN \underline{Instr} ; IS \Rightarrow$   
 $IF Bool THEN IS ; IS$

(b) "IF Bool THEN IF Bool THEN IS ELSE IS"

Derivaciones para mostrar ambigüedad:

**Leftmost:**

$\underline{Instr} \Rightarrow IF\ Bool\ THEN\ \underline{Instr}\ ELSE\ Instr \Rightarrow$

$IF\ Bool\ THEN\ IF\ Bool\ THEN\ \underline{Instr}\ ELSE\ Instr \Rightarrow$

$IF\ Bool\ THEN\ IF\ Bool\ THEN\ IS\ ELSE\ \underline{Instr} \Rightarrow$

$IF\ Bool\ THEN\ IF\ Bool\ THEN\ IS\ ELSE\ IS$

**Rightmost:**

$\underline{Instr} \Rightarrow IF\ Bool\ THEN\ Instr\ ELSE\ \underline{Instr} \Rightarrow$

$IF\ Bool\ THEN\ \underline{Instr}\ ELSE\ IS \Rightarrow$

$IF\ Bool\ THEN\ IF\ Bool\ THEN\ \underline{Instr}\ ELSE\ IS \Rightarrow$

$IF\ Bool\ THEN\ IF\ Bool\ THEN\ IS\ ELSE\ IS$

(c) Usando llaves para evitar ambigüedades:

f usando llaves sería:  $IF\ Bool\ THEN\ \{IS\ ;\ IS\}$  ó  $IF\ Bool\ THEN\ \{IS\}\ ;\ IS$

g usando llaves sería:  $IF\ Bool\ THEN\ \{IF\ Bool\ THEN\ IS\}\ ELSE\ IS$  ó  $IF\ Bool\ THEN\ \{IF\ Bool\ THEN\ IS\ ELSE\ IS\}$

(d) Usando done:

Para f:

$IF\ Bool\ THEN$

$IS$

$done;$

$IS$

ó

$IF\ Bool\ THEN$

$IS\ ;\ IS$

$done$

Para g:

*IF Bool THEN*

*IF Bool THEN*

*IS*

*done*

*ELSE*

*IS*

ó

*IF Bool THEN*

*IF Bool THEN IS ELSE IS*

*done*

*done*