

EJERCICIOS sobre SINTAXIS (BNF)

1. Una gramática que incluya las reglas de producción:

<sentencia> ::= <sent-condicional> | begin <sentencia> end |

**<sent-condicional> ::= if <comparación> then <sentencia> |
if <comparación> then <sentencia> else <sentencia>**

produce la ambigüedad denominada “else peligroso”.

Con la ayuda de un fragmento de árbol sintáctico:

- a) mostrar cómo surge la ambigüedad
 - b) explicar cómo se puede eliminar dicha ambigüedad, cambiando la gramática de modo que detrás de “then” sólo se permita una sentencia no condicional.
 - c) explicar cómo se puede eliminar, alternativamente, cambiando la gramática de forma que cada sentencia condicional termine con el símbolo “fi”
2. Usando la tabla 2.1 (BNF de Pam), dibujar el árbol sintáctico completo de un programa Pam pequeño.
 3. Rescribir la gramática de la tabla 2.1 disminuyendo el número de símbolos no terminales tanto como sea posible.
 4. Dar un conjunto de reglas de producción BNF para describir “variables suscritas” de la forma: **<nombre> (<subíndice>,...,<subíndice>)**
 5. Lo siguiente es una descripción BNF de una expresión **<exp>** en algún lenguaje donde f1, f2 y f3 son operadores (f2 denota un operador binario y otro unario):

<lit> ::= A | B | C | D | E | F
<basico> ::= <lit> | <basico> f1 <lit>
<inter> ::= <basico> | <inter> f3 <basico>
<exp> ::= <inter> | f2 <inter> | <exp> f2 <inter>

Las reglas de producción definen implícitamente la prioridad y asociatividad de los operadores f1, f2 y f3 en la evaluación de expresiones. Dedúcelas razonadamente y parentiza la siguiente expresión para mostrar el orden de evaluación:

f2 F f1 D f1 B f2 C f3 A

6. Usando la tabla 2.2 (BNF de Eva), dibujar el árbol sintáctico completo de un programa Eva pequeño. Asegurar también que se cumplen las condiciones de contexto que se exigen a los programas Eva.
7. Modificar la gramática de la tabla 2.2 para que no sea necesario que las declaraciones de un “bloque” precedan a las sentencias (es decir, una sentencia ejecutable podría tener a continuación una declaración)
8. Traducir a reglas BNF puro cada una de las siguientes reglas descritas en distintos **BNF extendidos**:
 - a)
 $\langle \text{sentencia DO FORTRAN} \rangle ::=$
 $[\langle \text{etiqueta} \rangle] \text{ DO } \langle \text{etiqueta} \rangle \langle \text{variable} \rangle = \langle \text{inicial} \rangle , \langle \text{limite} \rangle [, \langle \text{incremento} \rangle]$
 - b)
 $\langle \text{decl variables Pascal} \rangle ::= \text{var } \langle \text{decl var} \rangle \{ ; \langle \text{decl var} \rangle \} ;$
 $\langle \text{decl var} \rangle ::= \langle \text{identif} \rangle \{ , \langle \text{identif} \rangle \} : \langle \text{tipo} \rangle$
9. Buscar, en los manuales de distintos lenguajes de programación, la descripción sintáctica formal de dichos lenguajes y comprobar qué símbolos se usan para el metalenguaje BNF en cada caso.