**TC2006 – Lenguajes de Programación***Construcción de un Analizador Sintáctico*

Matrícula1\_\_\_\_\_\_\_\_\_\_\_ Matrícula2\_\_\_\_\_\_\_\_\_\_\_ Matrícula3\_\_\_\_\_\_\_\_\_\_\_

Utilicen el léxico del ejercicio anterior (*Construcción de un Escáner*) y modifiquen el programa del Parser en Python visto en clase para que acepte instrucciones más complejas que las expresiones que ya acepta.

Toda instrucción válida debe ser una asignación seguida de un signo de pesos. Las asignaciones se forman con un identificador seguido de un operador de asignación (=) seguido por una expresión. Las expresiones pueden ser operaciones aritméticas binarias entre expresiones, expresiones entre paréntesis, expresiones condicionales, constantes, o identificadores.

Una expresión condicional se escribe entre llaves {} e incluye una expresión-relacional, que se forma con un operador relacional entre dos expresiones matemáticas, seguida del símbolo ?, seguido de dos expresiones matemáticas separadas con el símbolo :.

**Ejemplos de expresiones válidas:**

c = a + e $

v = 5.3 \* (2 + x) $

y = {x + 1 > z ? 2.34 : z – 4} $

tasa = 4 \* {tasa <= 0.2 ? 5 : 8} $

x = {{x != y/3 ? (x+1) \* 1.2 : y-2} == z ? 2.2 : {x >= y ? 2 : 4}} $

**¿Cómo?**

1. Modifiquen el archivo **obten\_token.py**, para que contenga el código del scanner para aceptar los elementos léxicos del ejercicio anterior.
2. Modifiquen la gramática original para aceptar asignaciones y expresiones condicionales.
3. Eliminen cualquier ambigüedad en la gramática modificando, si existen, reglas que incluyan recursión izquierda o factores comunes por la izquierda.
4. Modifiquen el código de las funciones en el archivo parser.py, para que implemente la gramática no ambigua.
5. Prueben el programa y verifiquen que trabaje correctamente.
6. Suban el programa fuente del escáner y parser modificados a Canvas (uno solo de los integrantes).