

# Procesadores de Lenguaje

GRADO EN INGENIERÍA INFORMÁTICA

## Práctica Final: Grupo 3

**Curso** 2020/2021

Jorge Rodríguez Fraile, 100405951, Grupo 83, [100405951@alumnos.uc3m.es](mailto:100405951@alumnos.uc3m.es)  
Carlos Rubio Olivares, 100405834, Grupo 83, [100405834@alumnos.uc3m.es](mailto:100405834@alumnos.uc3m.es)

# Índice

Explicación de lo implementado:	3
Punto 1	3
Punto 2	3
Punto 3	3
Punto 4	3
Punto 5	3
Punto 6	3
Punto 7	3

# Explicación de lo implementado:

## Punto 1

Para el primer paso hemos cambiado en la impresión de expresiones el caracter '#' por '\$', para esto, hemos sustituido dicho caracter en las reglas de producción de impresión.

## Punto 2

Para poder introducir más de un parámetro en la función de impresión que se llama con '\$(parámetros);', hemos creado un nuevo no terminal llamado impr que represente las expresiones impresas. Este nuevo no terminal puede ser una expresión o una expresión, coma e impr.

De esta manera se permite imprimir una sola expresión o generar una secuencia de impresiones manteniendo el orden en el que se introdujeron.

## Punto 3

En este caso nos hemos fijado en la regla de producción *INTEGER IDENTIF*; esta regla generará una producción parecida al del comando setq, pero en este caso concatenamos un '0' para inicializar la variable. La introducción de este paso ha sido bastante sencilla en este caso.

## Punto 4

En este punto hemos decidido extraer las asignaciones en un nuevo no terminal llamado setq, en el que hemos metido las anteriores asignaciones creadas y hemos utilizado una estructura muy similar a la del punto anterior, pero en este caso en vez de terminar recibiendo 'IDENTIF;' recibe 'IDENTIF=NUMERO;'.

## Punto 5

Para la creación de main hemos utilizado la palabra reservada *MAIN*, y creado un no terminal llamado mainfun, este NT se llama en def y puede ser vacío. Dentro de mainfun tenemos la cabecera, un cuerpo que pueden ser asignaciones de variables o cualquier expresión (se explicará en el punto 7) y el cierre de la llave de la función.

## Punto 6

Para poder establecer que primero se encuentren solo asignaciones y posteriormente se definan todas las funciones, hemos cambiado el axioma para que produzca la regla 'decl def', decl solo podrá generar asignaciones o nada.

Por otro lado, def como se ha descrito anteriormente produce las funciones, incluido main. Aunque puede no haber funciones declaradas.

## Punto 7

Para esta funcionalidad nos fijamos en el no terminal cuerpo que es el que tiene todas las expresiones permitidas en la función. Este elemento puede derivar en una impresión seguida de cuerpo, una asignación de variable, o una expresión, también tenemos en cuenta que pueda ser vacío.