



Universidad Carlos III  
Grado en ingeniería informática  
Curso Procesadores del Lenguaje 2024-25  
Laboratorio 1  
Curso 2024-25

Fecha: **29/01/2025** - ENTREGA: 1

GRUPO: 3

Alumnos: **Mario Ramos Salsón (100495849)**

**Miguel Yubero Espinosa (100495984)**

# Cuestiones abiertas:

## 1.1 ¿Por qué aparece este error?

Este error de sintaxis aparece porque la gramática proporcionada sólo admite una única expresión seguida de un <intro> o salto de línea. Tras procesar esta primera línea de la expresión la función del parser “yyparse()” finaliza, y cualquier expresión que queramos introducir a posteriori no es reconocida por la gramática, lo que deriva en el error.

## 1.2 ¿Eres capaz de evitar que aparezca? Describe qué hay que añadir a la gramática (también las acciones semánticas) y qué efectos produce en el resultado.

Para solucionar el problema, redefinimos la gramática para que acepte múltiples expresiones.

axioma:

```
/* Vacío para permitir múltiples expresiones */  
| axiom expression '\n' { printf("Expresion = %.2lf\n", $2); }  
;
```

La clave está en hacer que la gramática sea recursiva, para que el axioma pueda procesar expresión tras expresión sin que el programa se cierre. Al poner axioma: axiom expresión '\n', el parser entiende que puede seguir recibiendo líneas (cada una con una expresión seguida de intro). Cuando ejecuta yyparse() una sola vez en el main(), el parser entra en un bucle interno y sigue procesando hasta que detecta un EOF (cuando presiono Ctrl+D o Ctrl+Z). Cada vez que se escribe una expresión y presionamos Enter, la acción semántica con printf muestra el resultado al instante. Si meto un error de sintaxis, como un "5+\*3", el programa avisa con yyerror(), pero no se cierra.