

## Laboratorio 1

### Enlaces:

- Git: [Gegdgt/Lab1\\_Compis](https://github.com/Gegdgt/Lab1_Compis)
- Video: <https://youtu.be/wLLd03uVCns>

### Analisis de MiniLang.g4:

El archivo .g4 es la gramática que define cómo se deben interpretar los textos (como expresiones matemáticas o lenguajes completos).

- Estructura g4

```
grammar MiniLang;
```

Aquí se declara el nombre de la gramática, el cual debe coincidir con el nombre del archivo.

- Reglas del parser y lexer

En ANTLR las letras minúsculas son reglas del parser (prog, expr, stat) y las letras mayúsculas son reglas del lexer (ID, INT, WS)

- Componentes importantes

- Prog

```
prog: stat+;
```

Aquí se define un programa como una o más sentencias

- Stat

```
stat: expr NEWLINE          # printExpr  
    | ID '=' expr NEWLINE   # assign  
    | NEWLINE               # blank  
    ;
```

Define los tipos de sentencias válidas:

- Una expresión aritmética (solo evaluarla)
- Una asignación (x = 5)
- Una línea vacía

El #nombre (como #assign) es un **label** que se usa para identificar variantes de una regla: esto facilita el uso de visitAssign, visitPrintExpr, etc., en el Visitor

- Expr

```
expr: expr op=('*' | '/') expr      # MulDiv
    | expr op=('+' | '-') expr      # AddSub
    | INT                          # int
    | ID                           # id
    | '(' expr ')'                  # parens
    ;
```

Define expresiones con:

- Operadores +, -, \*, /
- Agrupaciones con paréntesis
- Números enteros (INT) y variables (ID)

- Reglas del leer

```
ID      : [a-zA-Z]+ ;
INT      : [0-9]+ ;
NEWLINE  : '\r'? '\n' ;
WS       : [ \t]+ -> skip ;
```

Define cómo se reconocen:

- Identificadores (variables)
- Enteros
- Saltos de línea
- Espacios en blanco (que se ignoran con -> skip)