

# LENGUAJES DE PROGRAMACIÓN Y PROCESADORES DE LENGUAJES

Construcción de un compilador

MenosC

Parte-I: Análisis léxico sintáctico

El objetivo principal es la **construcción de un compilador** completo,  
para un Lenguaje de Programación de alto nivel, sencillo pero no trivial

## MenosC

Parte I Construcción del analizador léxico-sintáctico

⇒ límite de entrega **29 de octubre de 2017**

Parte II Construcción del analizador semántico

⇒ límite de entrega **26 de noviembre de 2017**

Parte III Construcción del generador de código intermedio

⇒ límite de entrega **16 de enero de 2018**

[ para la recuperación: **26 de enero de 2018** ]

- Los identificadores son cadenas de letras (incluyendo “\_”) y dígitos, que comienzan siempre por una letra. Debe distinguirse entre mayúsculas y minúsculas.
- Las palabras reservadas se deben escribir en minúscula.
- Todas las constantes numéricas deben considerarse enteras.
- El signo + (ó –) de las constantes numéricas se tratará como un símbolo léxico independiente.
- Los espacios en blanco, retornos de línea y tabuladores deben ignorarse.
- Los comentarios deben ir precedidos por la doble barra (//) y terminar con el fin de la línea. Los comentarios no se pueden anidar.

programa	→ { secuenciaSentencias }
secuenciaSentencias	→ sentencia   secuenciaSentencias sentencia
sentencia	→ declaracion   instruccion
declaracion	→ tipoSimple <b>id</b> ;   tipoSimple <b>id</b> [ <b>cte</b> ] ;
tipoSimple	→ <b>int</b>   <b>bool</b>
instruccion	→ { listaInstrucciones }   instruccionExpresión → instruccionEntradaSalida   instruccionSeleccion → instruccionIteracion
listaInstrucciones	→ $\epsilon$   listaInstrucciones instruccion
instruccionExpresión	→ expresión ;   ;
instruccionEntradaSalida	→ <b>read</b> ( <b>id</b> ) ;   <b>print</b> ( expresion ) ;
instruccionSeleccion	→ <b>if</b> ( expresion ) instruccion restolf
restolf	→ <b>elseif</b> ( expresion ) instruccion restolf   <b>else</b> instruccion
instruccionIteracion	→ <b>while</b> ( expresion ) instruccion → <b>do</b> instruccion <b>while</b> ( expresion )

# ESPECIFICACIÓN SINTÁCTICA

---

expresion	→ expresionLogica   <b>id</b> operadorAsignacion expresion
	→ <b>id</b> [ expresion ] operadorAsignacion expresion
expresionLogica	→ expresionIgualdad
	→ expresionLogica operadorLogico expresionIgualdad
expresionIgualdad	→ expresionRelacional
	→ expresionIgualdad operadorIgualdad expresionRelacional
expresionRelacional	→ expresionAditiva
	→ expresionRelacional operadorRelacional expresionAditiva
expresionAditiva	→ expresionMultiplicativa
	→ expresionAditiva operadorAditivo expresionMultiplicativa
expresionMultiplicativa	→ expresionUnaria
	→ expresionMultiplicativa operadorMultiplicativo expresionUnaria
expresionUnaria	→ expresionSufija
	→ operadorUnario expresionUnaria   operadorIncremento <b>id</b>

expresionSufija	→ <b>id</b>   <b>id</b> [ expresion ]   <b>cte</b>   ( expresion )
	→ <b>id</b> operadorIncremento   <b>true</b>   <b>false</b>
operadorAsignacion	→ =   +=   -=   *=   /=
operadorLogico	→ &&
operadorIgualdad	→ ==   !=
operadorRelacional	→ >   <   >=   <=
operadorAditivo	→ +   -
operadorMultiplicativo	→ *   /   %
operadorUnario	→ +   -   !
operadorIncremento	→ ++   --