



Euler es un [lenguaje de programación](#) creado por [Niklaus Wirth](#) y [Helmut Weber](#), concebido como una extensión y generalización de [ALGOL 60](#). La meta de sus diseñadores era crear un lenguaje:

- más simple y sencillo que ALGOL 60
- que fuese útil y al mismo tiempo eficiente
- que pudiese ser definido rigurosamente con herramientas formales

## El lenguaje

Euler usa un concepto de tipo general. En Euler, los arrays, procedimientos y condiciones múltiples no son entidades con nombre: al contrario que en ALGOL, no están al mismo nivel que las variables. En cambio, se comportan al nivel de constantes booleanas y numéricas. Por tanto, aparte del tradicional tratamiento de dichas constantes, Euler presenta los siguientes nuevos tipos de datos:

- referencia
- etiqueta
- símbolo
- lista (array)
- procedimiento
- indefinido

Todas las constantes pueden ser asignadas a variables, que tienen la misma forma que en ALGOL, pero que no tienen tipo fijo: Euler es un [lenguaje de programación con tipos dinámicos](#). Es más, un procedimiento puede devolver un valor de cualquier tipo, y este tipo puede ser distinto en distintas llamadas al mismo procedimiento.

Similarmente, los elementos de una lista pueden tener valores de cualquier tipo, incluso dentro de la misma lista. Así que si los elementos de la lista son etiquetas o procedimientos, tenemos una condición múltiple (*switch*) o una lista de procedimientos, respectivamente. La creación de listas de listas es en esencia un tipo de datos árbol. Como era de esperar, Euler



proporciona operadores generales de comprobación y conversión de tipos.