

# LENGUAJE C

**C** es un lenguaje de programación de propósito general<sup>2</sup> :<sup>1</sup> originalmente desarrollado por Dennis Ritchie entre 1969 y 1972 en los Laboratorios Bell, <sup>1</sup> como evolución del anterior lenguaje B, a su vez basado en BCPL.<sup>2</sup> :<sup>13</sup> <sup>4</sup>

Al igual que B, es un lenguaje orientado a la implementación de sistemas operativos, concretamente Unix. C es apreciado por la eficiencia del código que produce y es el lenguaje de programación más popular para crear software de sistema, aunque también se utiliza para crear aplicaciones. Se trata de un lenguaje de tipos de datos estáticos, débilmente tipificado, de medio nivel, ya que dispone de las estructuras típicas de los lenguajes de alto nivel pero, a su vez, dispone de construcciones del lenguaje que permiten un control a muy bajo nivel. Los compiladores suelen ofrecer extensiones al lenguaje que posibilitan mezclar código en ensamblador con código C o acceder directamente a memoria o dispositivos periféricos. La primera estandarización del lenguaje C fue en ANSI, con el estándar X3.159-1989. El lenguaje que define este estándar fue conocido vulgarmente como ANSI C. Posteriormente, en 1990, fue ratificado como estándar ISO (ISO/IEC 9899:1990). La adopción de este estándar es muy amplia por lo que, si los programas creados lo siguen, el código es portable entre plataformas y/o arquitecturas

Uno de los objetivos de diseño del lenguaje C es que solo sean necesarias unas pocas instrucciones en lenguaje máquina para traducir cada elemento del lenguaje, sin que haga falta un soporte intenso en tiempo de ejecución. Es muy posible escribir C a bajo nivel de abstracción; de hecho, C se usó como intermediario entre diferentes lenguajes.

En parte, a causa de ser de relativamente bajo nivel y de tener un modesto conjunto de características, se pueden desarrollar compiladores de C fácilmente. En consecuencia, el lenguaje C está disponible en un amplio abanico de plataformas (más que cualquier otro lenguaje). Además, a pesar de su naturaleza de bajo nivel, el lenguaje se desarrolló para incentivar la programación independiente de la máquina. Un programa escrito cumpliendo los estándares e intentando que sea portátil puede compilarse en muchos computadores.

C se desarrolló originalmente (conjuntamente con el sistema operativo Unix, con el que ha estado asociado mucho tiempo) por programadores para programadores. Sin embargo, ha alcanzado una popularidad enorme,

y se ha usado en contextos muy alejados de la programación de software de sistema, para la que se diseñó originalmente.

# Propiedades

- Núcleo del lenguaje simple, con funcionalidades añadidas importantes, como funciones matemáticas y de gestión de archivos, proporcionadas por bibliotecas.
- Es un lenguaje estructurado, i.e. tiene estructuras de control y tipos de datos estructurados definidos por el programador a partir de los tipos atómicos típicos y mediante arreglos, estructuras, uniones y apuntadores, incluidos los apuntadores a función.
- En su primera edición no había advertencias sobre asignar a una variable un valor un tipo distinto. Por lo que había un programa llamado `lint` que detectaba este tipo de errores. Actualmente los compiladores pueden detectar inconsistencias de tipos y otros errores.
- Usa un lenguaje de preprocesado, el preprocesador de C, para tareas como definir macros e incluir múltiples archivos de código fuente.
- Acceso a memoria de bajo nivel mediante el uso de apuntadores o punteros.
- Manejo de Interrupciones mediante la biblioteca `signal`.
- Un conjunto reducido de palabras clave.
- El llamado a funciones es por valor. Aunque se pueden pasar apuntadores a variables para hacer llamados por referencia.
- Distintos tipos de almacenamiento que permiten un diseño modular.

La palabra clave `static` permite encapsular variables y funciones.

- Las estructuras se declaran mediante la palabra clave `struct`, implementan el producto cartesiano de tipos también llamados registros.

Y la unión disyunta, también llamada suma, co-producto o tipos variantes, se declara con la palabra clave `union`. Las estructuras y uniones permiten la implementación de árboles.

- Lenguaje muy eficiente puesto que es posible utilizar sus características de bajo nivel para realizar implementaciones óptimas.
- Se diseñó como un lenguaje de programación de sistemas portátil.

Es muy poco lo que requiere uso de ensamblador, lo que permitió portar Unix y otros sistemas operativos a distintos equipos.

- Distingue entre mayúsculas y minúsculas, pero conserva su portabilidad en equipos que manejan caracteres de 6 bits, como las computadoras *Cyber* de CDC que tenían una palabra de 60 bits, (10 caracteres), porque usan trigráficos para codificar símbolos especiales.
- Su compilador de C se ha portado a casi todos los sistemas conocidos. Porque además de que no se requiere codificar en ensamblador, interactúa con los lenguajes especializados YACC y LEX. también escritos en C como parte de Unix.
- Facilita la programación modular gracias a que cuenta con distintos tipos de almacenamiento de las variables y compilación separada. Además del macro procesador integrado cpp, que permite declarar los encabezados de las funciones y los tipos de datos en archivos con extensión h. En el caso de las bibliotecas del sistema Unix y de C se necesita incluir los prototipos con la macros `#include` por ejemplo `#include <system.h>`. Ello únicamente sirve para automatizar la declaración de objetos externos que se usarán en el programa, por lo que es necesario pasar los nombres de las bibliotecas como parámetros del compilador que este pasa al encadenador (linker loader).