



# LIBRERÍAS DE C



**Departamento de Ciencias  
de la Computación**

**Asignatura:**

*"Lenguajes de  
Computación"*

**Maestro:**

Rosalinda Avendaño  
López

**Alumno:**

Juan Francisco Gallo  
Ramírez

**ID: 232872**

**Ingeniería en  
Computación Inteligente**



## LIBRERÍAS DE C

La biblioteca estándar de ANSI C consta de 24 ficheros cabecera que pueden ser incluidos en un proyecto de programación con una simple directiva. Cada cabecera contiene la declaración de una o más funciones, tipos de datos y macros.

Librería	Descripción
<assert.h>	Contiene la macro <code>assert</code> (aserción), utilizada para detectar errores lógicos y otros tipos de fallos en la depuración de un programa.
<complex.h>	Conjunto de funciones para manipular números complejos (nuevo en C99).
<ctype.h>	Contiene funciones para clasificar caracteres según sus tipos o para convertir entre mayúsculas y minúsculas independientemente del conjunto de caracteres (típicamente ASCII o alguna de sus extensiones).
<errno.h>	Para analizar los códigos de error devueltos por las funciones de biblioteca.
<fenv.h>	Para controlar entornos en coma flotante (nuevo en C99).
<float.h>	Contiene la definición de constantes que especifican ciertas propiedades de la biblioteca de coma flotante, como la diferencia mínima entre dos números en coma flotante ( <code>_EPSILON</code> ), el número máximo de dígitos de precisión ( <code>_DIG</code> ), o el rango de valores que se pueden representar ( <code>_MIN</code> , <code>_MAX</code> ).
<inttypes.h>	Para operaciones de conversión con precisión entre tipos enteros (nuevo en C99).
<iso646.h>	Para utilizar los conjuntos de caracteres ISO 646 (nuevo en NA1).
<limits.h>	Contiene la definición de constantes que especifican ciertas propiedades de los tipos enteros, como rango de valores que se pueden representar ( <code>_MIN</code> , <code>_MAX</code> ).
<locale.h>	Para la función <code>setlocale()</code> y las constantes relacionadas. Se utiliza para seleccionar el entorno local apropiado (configuración regional).
<math.h>	Contiene las funciones matemáticas comunes.
<setjmp.h>	Declara las macros <code>setjmp</code> y <code>longjmp</code> para proporcionar saltos de flujo de control de programa no locales.

<code>&lt;signal.h&gt;</code>	Para controlar algunas situaciones excepcionales como la división por cero.
<code>&lt;stdarg.h&gt;</code>	Posibilita el acceso a una cantidad variable de argumentos pasados a una función.
<code>&lt;stdbool.h&gt;</code>	Para el tipo booleano (nuevo en C99).
<code>&lt;stdint.h&gt;</code>	Para definir varios tipos enteros (nuevo en C99).
<code>&lt;stddef.h&gt;</code>	Para definir varios tipos de macros de utilidad.
<code>&lt;stdio.h&gt;</code>	Proporciona el núcleo de las capacidades de entrada/salida del lenguaje C (incluye la venerable función printf).
<code>&lt;stdlib.h&gt;</code>	Para realizar ciertas operaciones como conversión de tipos, generación de números pseudo-aleatorios, gestión de memoria dinámica, control de procesos, funciones de entorno, de señalización (??), de ordenación y búsqueda.
<code>&lt;string.h&gt;</code>	Para manipulación de cadenas de caracteres.
<code>&lt;tgmath.h&gt;</code>	Contiene funcionalidades matemáticas de tipo genérico (type-generic) (nuevo en C99).
<code>&lt;time.h&gt;</code>	Para tratamiento y conversión entre formatos de fecha y hora.
<code>&lt;wchar.h&gt;</code>	Para manipular flujos de datos anchos y varias clases de cadenas de caracteres anchos (2 o más bytes por carácter), necesario para soportar caracteres de diferentes idiomas (nuevo en NA1).
<code>&lt;wctype.h&gt;</code>	Para clasificar caracteres anchos (nuevo en NA1).