

|  |
| --- |
| Bibliotecas de C++  Angel Eduardo Muñoz Perez |
|  |
| 7/Feb/2023  Ingenieria en computación  Programación estructurada  Prof. Sergio Franco Casillas |



Imagen: https://www.wallpaperflare.com/c-logo-code-c-plus-plus-communication-glowing-data-technology-wallpaper-mfgax

Tabla de contenido

[¿Qué es una biblioteca? 3](#_Toc126662205)

[Tipos de bibliotecas 3](#_Toc126662206)

[Librería estática 3](#_Toc126662207)

[Librería dinámica 3](#_Toc126662208)

[Librerías remotas 3](#_Toc126662209)

[Bibliotecas en c++ 3](#_Toc126662210)

[De entrada, o salida 4](#_Toc126662211)

[Cadena 4](#_Toc126662212)

[Contenedores 4](#_Toc126662213)

[Iteradores o algoritmos 4](#_Toc126662214)

[Números 4](#_Toc126662215)

[Soporte del lenguaje 4](#_Toc126662216)

[Bibliografía 5](#_Toc126662217)

# ¿Qué es una biblioteca?

“Una librería es un conjunto de archivos que se utiliza para desarrollar software. Suele estar compuesta de código y datos, y su fin es ser utilizada por otros programas de forma totalmente autónoma.” [1] “Puede decirse que se trata de un conjunto de instrucciones que pueden ser utilizadas por diferentes programas. La misma librería, en teoría, es usada en varios programas, dándoles el mismo uso o uno diferente.” [2] Así que prácticamente las bibliotecas son importantes para la compilación de un código, sin ellos no habría lenguajes de alto nivel, ya que estos son básicamente como diccionarios que utiliza el lenguaje par que podamos comunicar con la computadora, logrando así que los comandos puedan ser exitosamente leídos por esta.

## Tipos de bibliotecas

A su vez, existen varios tipos de bibliotecas, y se pueden clasificar en 3 tipos ejecutables, copiándolos y relocalizándolos

### Librería estática

Aquí los archivos se mantienen estáticos a través de un fichero copiándolos y relocalizándolos, después podemos borrarlas sin problemas, ya que el programa seguirá funcionando con la función necesaria.

### Librería dinámica

Son archivos con código objeto que trabajan de manera independiente a su ubicación. Esas librerías se utilizan cargando los archivos en el tiempo de ejecución y no se enlazan en el tiempo de compilación.

### Librerías remotas

Estas se utilizan con ejecutables separados por medio de un procedimiento remoto, llevado a cabo por otra computadora. Trabajan enfocándose en la máxima reutilización del sistema operativo; el código que soporta la biblioteca es el que le da a la aplicación del soporte y la seguridad para el otro programa.

## Bibliotecas en c++

La biblioteca estándar de C++ está definida en el espacio de nombres std y las declaraciones necesarias para su utilización son proporcionadas por un conjunto de archivos de cabecera que se exponen a continuación. Con el fin de dar una idea general de la funcionalidad aportada por esta biblioteca.

* Entrada/salida
* Cadenas
* Contenedores
* Iteradores
* Algoritmos
* Números
* Diagnósticos
* Utilidades generales
* Localización
* Soporte del lenguaje

La aportación que realizan los contenedores, iteradores y algoritmos a la biblioteca estándar a menudo se denomina STL (Estandard Template Library)

### De entrada, o salida

*<cstdio> E/S de la biblioteca de C.*

*<cstdlib> Funciones de clasificación de caracteres*

*<cwchar> E/S de caracteres extendidos*

*<fstream> Flujos para trabajar con archivos en disco*

*<iomanip> Manipuladores*

*<ios> Tipos y métodos básicos de E/S*

*<iosfwd> Declaraciones adelantadas de utilidades de E/S*

*<iostream> Objetos y operaciones sobre flujos estándar de E/S*

*<istream> Objetos y operaciones sobre flujos de entrada*

*<ostream> Objetos y operaciones sobre flujos de salida*

### Cadena

*<ccytpe> Examinar y convertir caracteres*

*<cstdlib> Funciones de cadena estilo C*

*<cwchar> Funciones de cadena de caracteres extendidos estilo C*

*<string> Clases para manipular cadenas de caracteres*

### Contenedores

*<bitset> Matriz de bits*

*<deque> Cola de dos extremos de elementos tipo T*

*<list> Lista doblemente enlazada de elementos de tipo T*

*<map> Matriz asociativa de elementos de tipo T*

*<set> Conjuntos de elementos de tipo T*

*<stack> Pila de elementos de tipo T*

*<vector> Matriz de elementos de tipo T*

### Iteradores o algoritmos

*<iterator> soporte para iteradores*

*<algorithm> algoritmos generales (buscar, ordenar, contar, etc)*

*<cstdlib> bsearch y qsort*

### Números

<cmath> funciones matemáticas

<complex> operaciones con números complejos

<cstdlib> Números aleatorios tipo C

<numeric> Algoritmos numéricos generalizados

<valarray> Operaciones con matrices numéricas

### Soporte del lenguaje

*<cfloat> Limites numéricos en coma flotante estilo C*

*<ctime> echa y hora estilo C*

*<functional> Objetos función*

*<memory> Métodos para manipular bloques de memoria*

*<utility> Manipular pares de objetos*

*<climits> Limites numéricos estilo C*

*<limits> limites numéricos*

*<typeinfo> Identificacion de tipos durante la ejecución*

[3]

# Bibliografía

|  |  |
| --- | --- |
| [1] | «Qué es una librería en programación,» devcamp.es, - - 2020. [En línea]. Available: https://devcamp.es/que-es-libreria-programacion/. [Último acceso: 7 febrero 2023]. |
| [2] | «Qué es una librería de programación,» auladig.co, 23 marzo 2022. [En línea]. Available: https://auladig.co/que-es-una-libreria-de-programacion/. [Último acceso: 7 febrero 2023]. |
| [3] | F. J. Ceballos, Programación orientada a objetos con C++, Madrid: Ra-Ma, 2018. |