



# **GUÍA DE APRENDIZAJE PARA CONCURSANTES ICPC Y IOI: INTRODUCCIÓN A LA PROGRAMACIÓN EN C/C++**

---

## 1. Introducción

Durante el desarrollo de una posible solución a un ejercicio de concurso por parte del concursante se puede dividir en tres posibles etapas:

- **Lectura y comprensión del problema:** El concursante lee y comprende de que se trata el problema. Es capaz de delimitar cual es la información descartable y cual es la importante del la cual extrae y sintetiza los datos con que cuenta.
- **Análisis y diseño de una solución lógica-matemática:** Elaboración de un algoritmo que permita resolver cualquier instancia de dicho problema de forma correcta sin importar los valores que pueden tomar datos para dicha instancia del problema. Para esto el concursante se apoya en algoritmos ya conocidos los cuales puede usar propiamente como fueron definidos por sus autores o con modificaciones realizadas por el concursante o un algoritmo elaborado por el propio concursante.
- **Implementación del algoritmo solución:** El concursante con apoyo de un lenguaje comprensible para la computadora escribe un programa que realice el algoritmo que da solución a cualquier instancia del problema.

En el paso 3 se dice *un lenguaje comprensible para la computadora* que no son más que los llamados lenguajes de programación. En los concursos de programación se permiten varios lenguajes de programación y los problemas deben tener una solución factible en cada uno de los lenguajes permitidos. Lo que sucede que esa llamada *solución factible* en cada uno de los lenguajes de programación no es equitativa producto que muchas veces dependerá de las restricciones del propio lenguaje y el conocimiento del concursante en dicho lenguajes. Por lo anterior sucede que en los concursos de programación existan preferencias por determinados lenguajes de programación. Entre dichos lenguajes de programación de encuentra C++.

## 2. Conocimientos previos

**Sistema Operativo** Es el conjunto de programas de un sistema informático que gestiona los recursos de hardware y provee servicios a los programas de aplicación de software. Estos programas se ejecutan en modo privilegiado respecto de los restantes.

**Lenguaje de Programación:** Es una herramienta que nos permite comunicarnos e instruir a la computadora para que realice una tarea específica. Cada lenguaje de programación posee una sintaxis y un léxico particular, es decir, forma de escribirse que es diferente en cada uno por la forma que fue creado y por la forma que trabaja su compilador para revisar, acomodar y reservar el mismo programa en memoria.

**Compilador:** Es un programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación. Usualmente el segundo lenguaje es lenguaje de máquina, pero también puede ser un código intermedio (bytecode), o simplemente texto.

**Lenguaje de máquina:** Es el sistema de códigos directamente interpretable por un circuito microprogramable como el microprocesador de una computadora o microcontrolador de un automata.

**Bytecode:** Es un tipo de código de programa que se compila y ejecuta en sistemas informáticos llamados VM (máquinas virtuales, por sus siglas en inglés). Los programadores pueden usar el bytecode en su forma original en cualquier plataforma en la que opere la VM, por lo que el código es independiente de la plataforma. También es intermedio, ya que sus funciones están a medio camino entre el código fuente y el código máquina. El bytecode también se conoce como código portable o P-Code.

**Entorno Integrado de Desarrollo (IDE):** Un entorno de desarrollo integrado, conocido también como IDE (*Integrated Development Environment*) por sus siglas en inglés, es un programa informático compuesto por un conjunto de herramientas de programación que facilitan el desarrollo de aplicaciones. Un IDE puede denominarse como un entorno de programación, esto significa que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación

### 3. Desarrollo

C++ es un lenguaje de programación orientado a objetos que toma la base del lenguaje C y le agrega la capacidad de abstraer tipos como en Smalltalk.

La intención de su creación en los principios de la década de los ochenta por Bjarne Stroustrup fue el extender al exitoso lenguaje de programación C con mecanismos que permitieran la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

Posteriormente se añadieron facilidades de programación genérica, que se sumó a los otros dos paradigmas que ya estaban admitidos (programación estructurada y la programación orientada a objetos). Por esto se suele decir que el C++ es un lenguaje de programación multiparadigma.

Las principales herramientas necesarias para escribir un programa en C++ son las siguientes:

- Un equipo ejecutando un sistema operativo (Una computadora con Windows, Mac o alguna distribución de Linux).
- Un compilador de C++ el mismo puede variar dependiendo del sistema operativo.
- Un entorno de desarrollo (IDE) de igual manera puede variar dependiendo del sistema operativo. A continuación de listan algunos de los mas usados por los concursantes.
  - Code::Blocks
  - Eclipse
  - CLion
- Tiempo para practicar

- Paciencia, pero mucha paciencia y perseverancia.

Adicionalmente deberías conocer

- Inglés (Recomendado). Casi toda la documentación y los problemas de las competencias internacionales se escriben en este idioma. Por tanto al menos tener la habilidad de lectura y comprensión en este idioma es fundamental.
- Estar familiarizado con C u otro lenguaje derivado (PHP, Python, etc).

Es recomendable tener conocimientos de C, debido a que C++ es una mejora de C, tener los conocimientos sobre este te permitirá avanzar más rápido y comprender aún más. También, hay que recordar que C++, admite C, por lo que se puede programar (reutilizar), funciones de C que se puedan usar en C++.

Aunque no es obligación aprender C, es recomendable tener nociones sobre la programación orientada a objetos en el caso de no tener conocimientos previos de programación estructurada. Asimismo, muchos programadores recomiendan no saber C para saber C++, por ser el primero de ellos un lenguaje imperativo o procedimental y el segundo un lenguaje de programación orientado a objetos.

## 4. Consejos iniciales antes de programar

Con la práctica, se puede observar que se puede confundir a otros programadores con el código que se haga. Antes de siquiera hacer una línea de código, si se trabaja con otros programadores, ha de tenerse en cuenta que todos deben escribir de una forma similar el código, para que de forma global puedan corregir el código en el caso de que hubieran errores o rastrearlos en el caso de haberlos. También es muy recomendable hacer uso de comentarios (comenta todo lo que puedas, hay veces que lo que parece obvio para ti, no lo es para los demás) y tratar de hacer un código limpio y comprensible, especificando detalles y haciendo tabulaciones, aunque te tome un poco más de tiempo, es posible que más adelante lo agradezcas tú mismo.