

## C.F.G.S.: DESARROLLO DE APLICACIONES WEB

### Módulo: Programación

---

# TEMA 1

## Introducción a la Programación

"Primero resuelve el problema.  
Entonces, escribe el código"

### ¿QUÉ SIGNIFICA PROGRAMAR?

Programar significa planificar y concretar la secuencia de órdenes concisas para la realización de una tarea. Cuando informáticamente se habla de programar se está pensando en las órdenes que se deben dar al ordenador para que lleve a cabo una tarea concreta. Esto se hace a través de lo que se denomina programa.

Aunque los ordenadores son máquinas complejas, no son capaces por sí solas de llevar a cabo aquellas tareas que el usuario desea, necesitan en todo momento que se le den las órdenes adecuadas. Esas órdenes escritas de la manera correcta son los **programas**. Todos estamos ya familiarizados con la ejecución de programas (editores de texto, navegadores, juegos).

Cuando se plantea un problema que debe solucionarse mediante un programa, una buena técnica de trabajo es analizar y representar de forma detallada las operaciones que se llevarán a cabo para solucionar el problema. **Algoritmo** es por tanto la representación detallada de las operaciones que se llevarán a cabo para solucionar un programa.

Un mismo problema puede resolverse con más de un algoritmo y por tanto más de un programa. Todo algoritmo debe ser:

- Preciso: orden de operaciones
- Finito: en el número de operaciones
- Correcto: debe solucionar el problema

Estas tres condiciones son imprescindibles en cualquier programa, pero además hay otras condiciones que se deben tener en cuenta y que hacen que de dos programas diferentes que resuelven correctamente un problema, uno sea mejor que otro.

- Eficiencia: debe ocupar lo menos posible para aprovechar al máximo la memoria y reducir el tiempo de ejecución.
- Fácil mantenimiento.

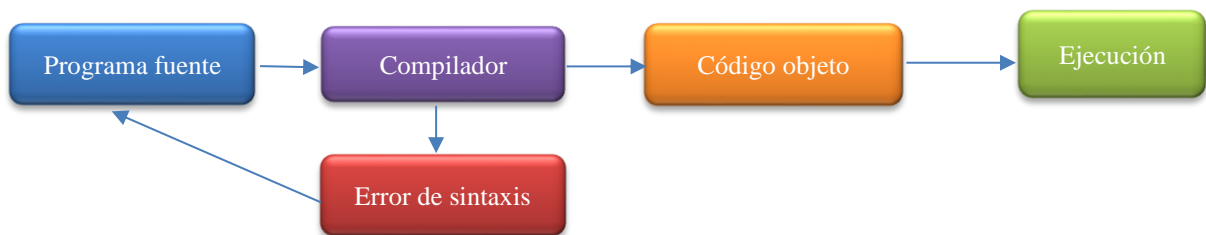
## LENGUAJE DE PROGRAMACIÓN

Un lenguaje de programación es una notación para escribir programas, a través de los cuales podemos comunicarnos con el hardware y dar así las órdenes adecuadas para la realización de un determinado proceso

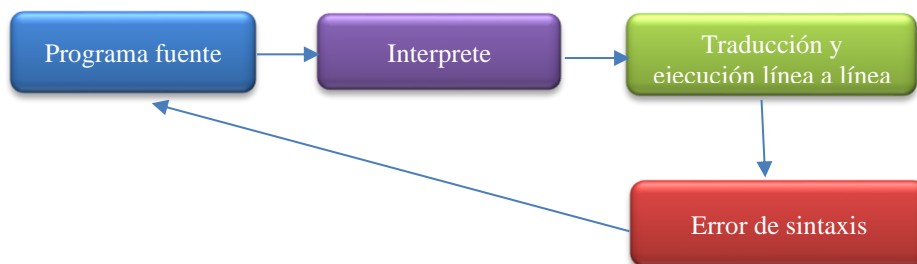
Si abrimos un programa con un editor de texto aparecen símbolos ininteligibles (para nosotros). Los programas están en lenguaje binario (lenguaje máquina), que es el que entienden las máquinas. Nosotros aprenderemos a escribir programas en un lenguaje entendible para los humanos (lenguajes de alto nivel)

Nosotros escribimos el código de un programa en un lenguaje determinado (código fuente), esas instrucciones son traducidas al lenguaje máquina (código máquina) por otros programas llamados compiladores o intérpretes

- **Compilador:** traduce un programa en lenguaje de alto nivel a lenguaje máquina, después ejecuta. El compilador detecta los errores sintácticos mediante mensajes. El código objeto sólo se genera si no hay errores sintácticos.



- **Intérprete:** traduce cada instrucción y la ejecuta línea a línea. Si se encuentra algún error sintáctico para.



## ERRORES

Cuando codificamos un programa pueden darse distintos tipos de errores

- **Sintácticos:** Errores de sintaxis, los detecta el compilador o el intérprete, son fáciles de corregir, se refieren a algo mal escrito.
- **De ejecución:** el programa es correcto sintácticamente pero no acaba su ejecución de forma correcta.
- **De lógica:** el programa si acaba de ejecutarse pero no procesa como debe.

## LENGUAJE JAVA

### HISTORIA

**Java** es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principio de los años 90's.

*Patrick Naughton*, ingeniero de **Sun Microsystems**, reclutó a varios colegas entre ellos *James Gosling* y *Mike Sheridan* para trabajar sobre un nuevo proyecto conocido como "*El proyecto verde*".

Intentaban desarrollar una nueva tecnología para programar la siguiente generación de dispositivos inteligentes (microelectrónica), en los que *Sun* veía un campo nuevo a explorar. Querían crear un lenguaje de programación fácil de aprender y de usar. En un principio se consideraba *C++* como lenguaje a utilizar, pero tanto *Gosling* como *Bill Joy* lo encontraron inadecuado. *Gosling* intentó primero extender y modificar *C++*, pero lo abandonó para crear un nuevo lenguaje desde cero al que llamo **Oak** (roble en inglés, según la versión más aceptada, por el roble que veía a través de la ventana de su despacho).

El resultado fue un lenguaje que tenía similitudes con *C*, *C++* y que no estaba ligado a un tipo de CPU concreta.

Mas tarde, se cambiaría el nombre de **Oak a Java**, por cuestiones de propiedad intelectual. Se supone que le pusieron ese nombre mientras tomaban café (Java es nombre de un tipo de café, originario de Asia), aunque otros afirman que el nombre deriva de las siglas de *James Gosling*, *Arthur Van Hoff*, y *Andy Bechtolsheim*. Otros abogan por el siguiente acrónimo, *Just Another Vague Acronym* ("sólo otro acrónimo ambiguo más").

El 3 de septiembre de 1992 se termina el desarrollo y con ello el *Proyecto Verde*.

Afortunadamente, el cese del *Proyecto Verde* coincidió con el nacimiento del fenómeno mundial **WEB**. Al examinar las dinámicas de Internet, lo realizado por el ex equipo verde se adecuaba a este nuevo ambiente.

El 29 de septiembre de 1994 se termina el desarrollo del prototipo de *HotJava*. Cuando se hace la demostración a los ejecutivos de Sun, esta vez, se reconoce el potencial de Java y se acepta el proyecto.

Una de las características de *HotJava* fue su soporte para los "applets", que son las partes de Java que pueden ser cargadas mediante una red de trabajo para después ejecutarlo localmente y así lograr soluciones dinámicas en computación acordes al rápido crecimiento del ambiente WEB.

En enero de 1995 Sun forma la empresa *Java Soft* para dedicarse al desarrollo de productos basados en las tecnologías Java, y así trabajar con terceras partes para crear

aplicaciones, herramientas, sistemas de plataforma y servicios para aumentar las capacidades del lenguaje. Ese mismo mes aparece la versión **1.0 del JDK**.

*Netscape Communications* decide apoyar a *Java applets* en Netscape Navigator 2.0.

Ese fue el factor clave que lanzó a Java a ser conocido y famoso.

Y como parte de su estrategia de crecimiento mundial y para favorecer la promoción de la nueva tecnología, Java Soft otorgó permisos para otras compañías para que pudieran tener acceso al código fuente y al mismo tiempo mejorar sus navegadores. También les permitía crear herramientas de desarrollo para programación Java y los facultaba para acondicionar máquinas virtuales *Java (JVM)*, a varios sistemas operativos.

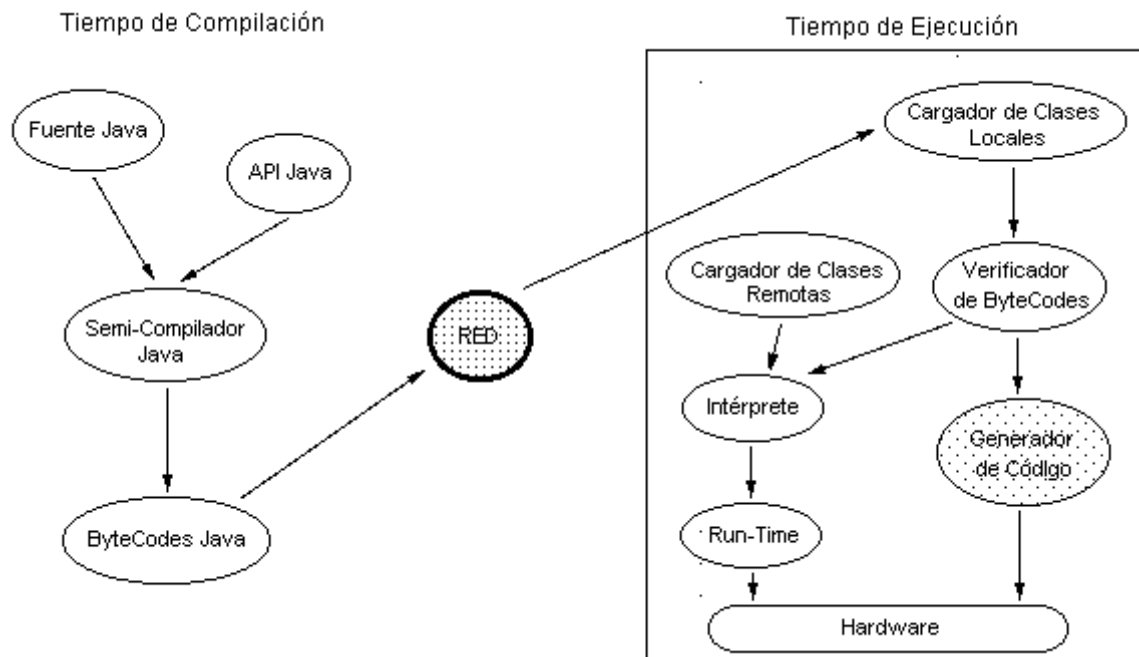
Los nuevos proyectos de *Java* son co-patrocinados por cientos de millones de dólares en capital disponible de recursos tales como la *Fundación Java*, un fondo común de capital formado por 11 compañías, incluyendo Cisco Systems, IBM, Netscape y Oracle.

Hoy en día, se puede encontrar la tecnología *Java* en redes y dispositivos que comprenden desde Internet y superordenadores científicos hasta portátiles y teléfonos móviles; desde simuladores de mercado en Wall Street hasta juegos de uso doméstico y tarjetas de crédito: *Java* está en todas partes.

## CARACTERÍSTICAS

### POSITIVAS

- Simple. Se basa en C++ pero elimina elementos complejos:
  - . No existen punteros.
  - . No es necesario liberar memoria (free) pues hay un recolector automático.
  - . No tiene estructuras ni uniones, ni definición de tipos.
  - . No hay funciones. Todo con métodos actuando sobre objetos.
  - . No existe preprocesador (#define)
  - . No existen las referencias.
- Orientado a objetos.
- Distribuido: Los programas pueden correr interactuando con varias máquinas.
- Seguro: comprobaciones para evitar virus y usos indebidos (límites de arrays, comprobación de punteros, excepciones, comprobación de byte-code).
- Arquitectura neutral: El compilador de Java compila su código a un fichero objeto de formato independiente de la arquitectura de la máquina (bytecode) Cualquier máquina que tenga un programa llamado Máquina Virtual de Java (MVJ) que entienda el código en bytes puede ejecutar ese código objeto sin importar la máquina en que ha sido generado.( cuadro de la Carlos III)



- Multithreaded: Permite hebras de ejecución paralelas. Por ejemplo en un mismo programa un thread trae una imagen mientras otro va mostrando la información.
- Dinámico: Enlace de clases según demanda. Si alguna clase se actualiza, no influye en nuestro código que cogerá la actualización automáticamente del servidor.
- Lenguaje de propósito general (aplicaciones) y además adaptado a Internet. Todos los navegadores están adaptados a Java, esto es, contienen la MVJ. Cuando un usuario con una pulsación de ratón en una página Web activa un programa java, lo que sucede es que este programa es enviado por el servidor a nuestra máquina en bytecode a través de la red, y nuestro navegador, mejor dicho la MVJ contenida en nuestro navegador lo ejecuta. Estos programas se llaman applets (abreviatura de pequeña aplicación).

#### NEGATIVAS

- Lento, debido a que es interpretado, pero se están desarrollando chips Java.
- Complejo: La mayoría de los elementos crípticos de C++ permanecen.

JAVA es la principal arma de la estrategia contra Microsoft de Netscape, Sun, IBM, Oracle...

## ENTORNO DE DESARROLLO

JDK ( Java Developmen Kit). Es gratuito. Contiene un compilador completo y un intérprete para la última versión, y un tutorial.

Incluye el JRE(Java Runtime Environment) que consta de los mínimos componentes necesarios para ejecutar una aplicación java: la máquina virtual y las librerías de clases. Contiene además otras herramientas de consola

- javac: compilador
- javadoc: generador de documentación

Sistemas de desarrollo integrados:

- Los que tienen su propio compilador de Java: Java Workshop, Visual J++ de MicroSoft, Symantec's Café, Jbuilder de Borland.
- Los que tienen por debajo el JDK: Visual Studio Code, NetBeans, Eclipse.

Visual Studio Code:

Descargarlo (en la máquina física ya está instalado):

[Download Visual Studio Code - Mac, Linux, Windows](#)

Instalar el paquete Java Extension pack

En preferences/settings poner en java.home la ruta del jdk

Creamos los proyectos desde explorer

NetBeans, lo descargamos accediendo al siguiente enlace (Lo usamos más adelante):

<https://www.oracle.com/technetwork/es/java/javase/downloads/jdk-netbeans-jsp-3413139-es.html>

Eclipse, no se instala, solo se descarga. Es el que vamos a usar, ya está instalado en la máquina física. Se puede descargar en linux.

[Eclipse Packages | The Eclipse Foundation - home to a global community, the Eclipse IDE, Jakarta EE and over 350 open source projects...](#)

Programa simple:

```
public class Primero {  
  
    // Mi primer programa en Java  
  
    public static void main( String [] args){  
        System.out.println("!Este es mi primer programa en Java!");  
    }  
}
```