



Universidad Nacional Autónoma de México  
Facultad de Ingeniería



## Fundamentos de programación

Actividad asíncrona 1 || Historia de la programación

Laura Mildred Moreno Razo

Fecha: 02/10/2020

## Historia de la programación

La historia de la Programación está relacionada directamente con la aparición de los computadores, que ya desde el siglo XV tuvo sus inicios con la construcción de una máquina que realizaba operaciones básicas (Gottfried Wilheml von Leibniz); aunque en realidad la primera gran fue la máquina diferencial para el cálculo de polinomios, proyecto no concluido de Charles Babbage (1793-1871) con el apoyo de Lady Ada Countess of Lovelace (1815-1852), primera persona que incursionó en la programación y de quien proviene el nombre del lenguaje de programación ADA creado por el DoD (Departamento de defensa de Estados Unidos) en la década de 1970.

Babagge intento crear la máquina que se pudiera programar con tarjetas perforadas para efectuar cualquier cálculo con una precisión de 20 dígitos. Pero la tecnología de la época no bastaba para hacer realidad sus ideas. Si bien las ideas de Babagge no llegaron a materializarse de forma definitiva, su contribución es decisiva, ya que los ordenadores actuales responden a un esquema análogo al de la máquina analítica. El camino señalado de Babbage, no fue nunca abandonado y siguiéndolo, se construyeron los primeros computadores. Cuando surgió el primer ordenador, el famoso ENIAC (Electronic Numerical Integrator And Calculator), su programación se basaba en componentes físicos, o sea, que se programaba, cambiando directamente el Hardware de la máquina, lo que se hacía era cambiar cables de sitio para conseguir así la programación. La entrada y salida de datos se realizaba mediante tarjetas perforadas.

## Lenguajes de programación

1. Java Es el más actualizado actualmente y esto se debe a su simplicidad y legibilidad. No en vano, más de 9 millones de usuarios lo usan
2. Lenguaje de programación C. Segundo lenguaje más usado a nivel mundial.
3. Python Un lenguaje de programación multiplataforma y multiparadigma, que también tiene un propósito general
4. C++ Es una evolución de C, tiene un enfoque en desarrollo de aplicaciones y softwares más complejos, que exigen, por ejemplo, una intervención visual.
5. C# Evolución de C y C++ ,lenguaje de programación orientado a objetos.
6. Visual Basic. NET
7. SQL

8. PHP
9. Ruby
10. Lenguaje de programación R
11. Rust
12. TypeScript
13. Swift
14. Perl
15. Lenguaje de programación Go
16. Kotlin
17. Scheme
18. Erlang
19. Elixir
20. Pascal

#### Clasificación lenguajes de programación

**Lenguaje máquina.** Es el sistema de códigos interpretable directamente por un circuito micro programable, como el microprocesador de una computadora. Este lenguaje se compone de un conjunto de instrucciones que determinan acciones que serán realizadas por la máquina. Y un programa de computadora consiste en una cadena de estas instrucciones de lenguaje de máquina (más los datos). Normalmente estas instrucciones son ejecutadas en secuencia, con eventuales cambios de flujo causados por el propio programa o eventos externos. El lenguaje máquina es específico de cada máquina o arquitectura de la máquina, aunque el conjunto de instrucciones disponibles pueda ser similar entre ellas.

**Lenguajes de bajo nivel.** Es el que proporciona poca o ninguna abstracción del microprocesador de una computadora. Consecuentemente, su traslado al lenguaje máquina es fácil. El término ensamblador, se refiere a un tipo de programa informático encargado de traducir un archivo fuente, escrito en un lenguaje ensamblador, a un archivo objeto que contiene código máquina ejecutable directamente por la máquina para la que se ha generado.

**Lenguaje de alto nivel.** Los lenguajes de programación de alto nivel se caracterizan porque su estructura semántica es muy similar a la forma como escriben los humanos, lo que permite codificar los algoritmos de manera más natural, en lugar de codificarlos en el lenguaje binario de las máquinas, o a nivel de lenguaje ensamblador.

### Tipos de paradigma

**Lenguajes imperativos** Emplean instrucciones, órdenes concretas como unidad de trabajo de los programas (Cobol, Pascal, C, Ada).

**Lenguajes declarativos** Los programas se construyen mediante descripciones de funciones o expresiones lógicas para responder a las cuestiones (Lisp, Prolog).

**Lenguajes orientados a objetos** El diseño de los programas se basa más en los datos y su estructura. La unidad de proceso es el objeto y en él se incluyen los datos (variables) y operaciones que actúan sobre ellos (Smalltalk, C++).

**Lenguajes orientados al problema** Están diseñados para problemas específicos, principalmente de gestión; suelen ser generadores de aplicaciones.

**Lenguajes naturales** Están desarrollándose nuevos lenguajes con el objetivo de aproximar el diseño y construcción de programas al lenguaje de las personas.

### Bibliografía

Ceballos, F. J. (2004). Enciclopedia del lenguaje C. México: Alfaomega/RaMa.

Pineda, N., 2016. Un Poco De Historia | Fundamentos De Programación. [online] Utn.edu.ec. Disponible en:  
<[http://www.utn.edu.ec/reduca/programacion/fundamentos/un\\_poco\\_de\\_historia.html](http://www.utn.edu.ec/reduca/programacion/fundamentos/un_poco_de_historia.html)> [Accessed 2 October 2020].