

LENGUAJES DE PROGRAMACION

Un **lenguaje de programación** es un **lenguaje formal** diseñado para expresar **procesos** que pueden ser llevados a cabo por máquinas como las **computadoras**.

Pueden usarse para crear **programas** que controlen el comportamiento físico y lógico de una máquina, para expresar **algoritmos** con precisión, o como modo de comunicación humana.¹

Está formado por un conjunto de símbolos y reglas **sintácticas** y **semánticas** que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se **compila**(de ser necesario) y se mantiene el **código fuente** de un **programa informático** se le llama **programación**.

También la palabra programación se define como el proceso de creación de un **programa de computadora**, mediante la aplicación de procedimientos lógicos, a través de los siguientes pasos:

- El desarrollo lógico del programa para resolver un problema en particular.
- Escritura de la lógica del programa empleando un lenguaje de programación específico (codificación del programa).
- Ensamblaje o compilación del programa hasta convertirlo en lenguaje de máquina.
- Prueba y **depuración del programa**.
- Desarrollo de la documentación.

```
def add5(x):
    return x+5

def dotwrite(ast):
    nodename = getNodeName()
    label=symbol.sym_name.get(int(ast[0]),ast[0])
    print '    %s [label="%s' % (nodename, label),
    if isinstance(ast[1], str):
        if ast[1].strip():
            print '= %s' % ast[1]
        else:
            print ']'
    else:
        print '["';
        children = []
        for n, child in enumerate(ast[1:]):
            children.append(dotwrite(child))
        print ', %s -> {' % nodename
        for n, child in enumerate(children):
            print '%s' % child,
```

Existe un error común que trata por sinónimos los términos 'lenguaje de programación' y 'lenguaje informático'. Los lenguajes informáticos engloban a los lenguajes de programación y a otros más, como por ejemplo **HTML(lenguaje para el marcado de páginas web** que no es propiamente un lenguaje de programación, sino un **conjunto de instrucciones** que permiten estructurar el contenido de los documentos).

Permite especificar de *manera precisa* sobre qué datos debe operar una computadora, cómo deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada

gama de circunstancias. Todo esto, a través de un [lenguaje](#) que intenta estar *relativamente* próximo al lenguaje humano o natural. Una característica relevante de los lenguajes de programación es precisamente que más de un programador pueda usar un conjunto común de instrucciones que sean comprendidas entre ellos para realizar la construcción de un programa de forma colaborativa.

Para que la computadora entienda nuestras instrucciones debe usarse un lenguaje específico conocido como [código máquina](#), el cual la máquina comprende fácilmente, pero que lo hace excesivamente complicado para las personas. De hecho sólo consiste en cadenas extensas de [números 0 y 1](#).

Para facilitar el trabajo, los primeros operadores de computadoras decidieron hacer un traductor para reemplazar los 0 y 1 por palabras o abstracción de palabras y letras provenientes del [inglés](#); éste se conoce como [lenguaje ensamblador](#). Por ejemplo, para sumar se usa la letra A de la palabra inglesa *add* (sumar). El lenguaje ensamblador sigue la misma estructura del lenguaje máquina, pero las letras y palabras son más fáciles de recordar y entender que los números.

La necesidad de recordar secuencias de programación para las acciones usuales llevó a denominarlas con nombres fáciles de memorizar y asociar: ADD (sumar), SUB (restar), MUL (multiplicar), CALL (ejecutar subrutina), etc. A esta secuencia de posiciones se le denominó "instrucciones", y a este conjunto de instrucciones se le llamó [lenguaje ensamblador](#). Posteriormente aparecieron diferentes lenguajes de programación, los cuales reciben su denominación porque tienen una estructura [sintáctica](#) similar a los lenguajes escritos por los humanos, denominados también [lenguajes de alto nivel](#).

La primera programadora de computadora conocida fue [Ada Lovelace](#), hija de [Anabella Milbanke Byron](#) y [Lord Byron](#). Anabella introdujo en las matemáticas a Ada quien, después de conocer a [Charles Babbage](#), tradujo y amplió una descripción de su máquina analítica. Incluso aunque Babbage nunca completó la construcción de cualquiera de sus máquinas, el trabajo que Ada realizó con éstas le hizo ganarse el título de primera programadora de computadoras del mundo. El nombre del [lenguaje de programación Ada](#) fue escogido como homenaje a esta programadora.

El primer manual para el lenguaje [Fortran](#) apareció en octubre de [1956](#), con el primer [compilador Fortran](#) entregado en [abril](#) de [1957](#). Esto era un compilador optimizado, porque los clientes eran reacios a usar un [lenguaje de alto nivel](#) a menos que su compilador pudiera generar código cuyo desempeño fuera comparable al de un código hecho a mano en lenguaje ensamblador.

En [1960](#), se creó [COBOL](#), uno de los lenguajes usados aún en la actualidad, en [informática de gestión](#).

A medida que la complejidad de las tareas que realizaban las computadoras aumentaba, se hizo necesario disponer de un método más eficiente para programarlas. Entonces, se crearon los [lenguajes de alto nivel](#), como lo fue [BASIC](#) (JAVA, C, PASCAL, .NET, PITON, PHP etc) en las versiones introducidas en los microordenadores de la década de [1980](#). Mientras que una tarea tan sencilla como sumar dos números puede necesitar varias instrucciones en lenguaje ensamblador, en un lenguaje de alto nivel bastará una sola sentencia