

<b>Programar: RAE</b>	<b>2</b>
<b>Conceptos de la programación:</b>	<b>2</b>
<b>Historia de la programación:</b>	<b>2</b>
<b>INTÉRPRETE Y COMPILADOR</b>	<b>3</b>
<b>METODOLOGÍA</b>	<b>4</b>

# INTRODUCCIÓN A LA PROGRAMACIÓN

## Programar: RAE

- Formar programas previa declaración de lo que se piensa hacer y anuncio de las partes de que se ha de componer un acto o espectáculo o una serie de ellos.
- Ideal y ordenar las acciones necesarias para realizar un proyecto.
- Preparar ciertas máquinas o aparatos para que empiecen a funcionar en el momento y en la forma deseados.
- Elaborar programas para su empleo en computadoras.

## Conceptos de la programación:

ORDEN -> INSTRUCCIÓN -> RESULTADO

## Historia de la programación:

El comienzo de la programación se dio con el **lenguaje máquina**, el único lenguaje que entiende el microprocesador siendo su código binario (0, 1)

El **código binario** es el resultado de compilar el código para que se pueda interpretar y obtener **bytecode**, que son los archivos que forman parte de ejecutables que utilizan el código programado.

El **lenguaje ensamblador** es un claro ejemplo de lo que es el lenguaje máquina también denominado de bajo nivel.

**Un programa de bajo nivel es aquel que ejerce control directo al hardware** la palabra bajo hace referencia a una reducida abstracción entre lo que es el lenguaje y el hardware. Los programas de bajo nivel están hechos para controladores (drivers ) y aplicaciones especializadas en tiempo real. Trabaja directamente con el microprocesador de una máquina.

**Con la práctica se cayó en la cuenta de que se podía utilizar a la propia máquina para ayudar en la traducción de estos programas.**

**También se introdujo la posibilidad de indicar a la computadora la dirección de un salto en la secuencia de ejecución de un programa mediante la utilización de etiquetas.**

Los programas que pasan el programa escrito de esta manera al lenguaje máquina se denomina ensambladores. El lenguaje ensamblador tiene un control preciso sobre las tareas que realiza un microprocesador se dispone de instrucciones del CPU que no están disponibles en el lenguaje de alto nivel.

**Este tipo de programación tiene la desventaja de que es lenta y laboriosa ya que los datos e instrucciones se deben introducir en código binario además de conocer las posiciones de memoria donde se almacenan los datos.**

En los años cincuenta se creó una notación simbólica que utiliza una serie de abreviaturas para representar las operaciones básicas, sumar, copiar, etcétera.

Se desarrolló un programa traductor, ASSEMBLER, que funcionaba como compilador e interpreta el código máquina.

Así aparecen los lenguajes de programación de alto nivel cuya lógica es parecida al pensamiento humano: BASIC, PASAL, C...

En los noventa comienza a utilizarse javascript y Java además de otros como perl y PHP

En el año 2000 comienza a utilizarse C, Python y el lenguaje de iOS (Objective C) además se consolidan Java, JavaScript y Python.

Cabe destacar que el lenguaje es como COBOL o Fortran nunca mueren.

En dos mil veinticinco con la evolución de la inteligencia artificial el lenguaje más utilizado es Python seguido de Java por ser el más robusto y JavaScript el más utilizado por los desarrolladores web y frontends.

## **INTÉRPRETE Y COMPILADOR**

El **intérprete** traduce las instrucciones una a una, y línea a línea.

El **compilador** traduce todo el programa para que pueda ser ejecutado.

La compilación es una de las características que tienen todos los lenguajes de programación. La ordenes se tienen que traducir al código binario del ordenador que es el único que comprende su unidad central.

## **METODOLOGÍA**

Conjunto coherente de métodos y técnicas que cubre más de una etapa del ciclo de la vida. Representa el camino para desarrollar software de forma sistémica.

**Estructuradas:** entrada - proceso - salida.

**Orientados a objetos:** un objeto es una entidad tangible que almacena datos y los métodos que controlan dichos datos.