



ARQUITECTURA Y SISTEMAS OPERATIVOS

Lenguajes de programación

100

Un lenguaje de programación es un lenguaje especial, que especifica su propia sintaxis, reglas y un conjunto de palabras clave, utilizado para poder comunicar instrucciones a una máquina, más específicamente, a una computadora. Los lenguajes de programación, son los que nos permiten el desarrollo del software.

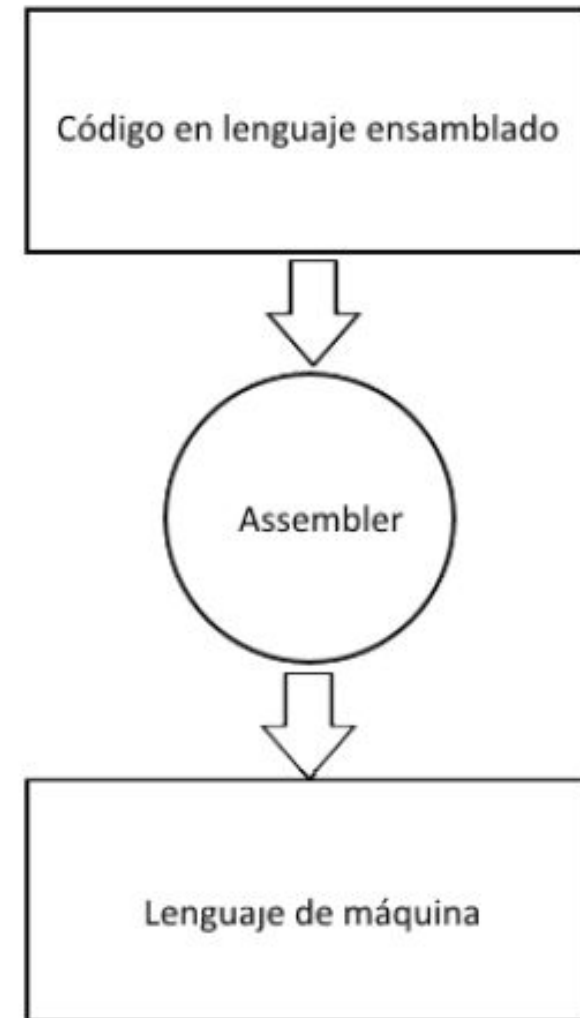


HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

- Durante la primera generación de computadoras (1938-1952) y hasta la creación de las primeras tarjetas perforadas, la lógica que ejecutaba cualquier sistema de computación se almacenaba de manera cableada en el equipo.
- La implementación de las tarjetas perforadas como medio de almacenamiento a principio de los años '50 (en el final de la 1ra generación de computadoras), da lugar a lo que se conoce como la primera generación de lenguajes de programación.
- Coincidiendo con el inicio de la segunda generación de computadoras (1953-1962), surgen los primeros lenguajes de ensamblado. Esto da lugar a una segunda generación de lenguajes de programación.
- Por último, a finales de los años '50, el desarrollo de nuevos lenguajes de programación como C, COBOL, BASIC, y muchos de los lenguajes de programación actuales, corresponden a la tercera generación de lenguajes de programación.

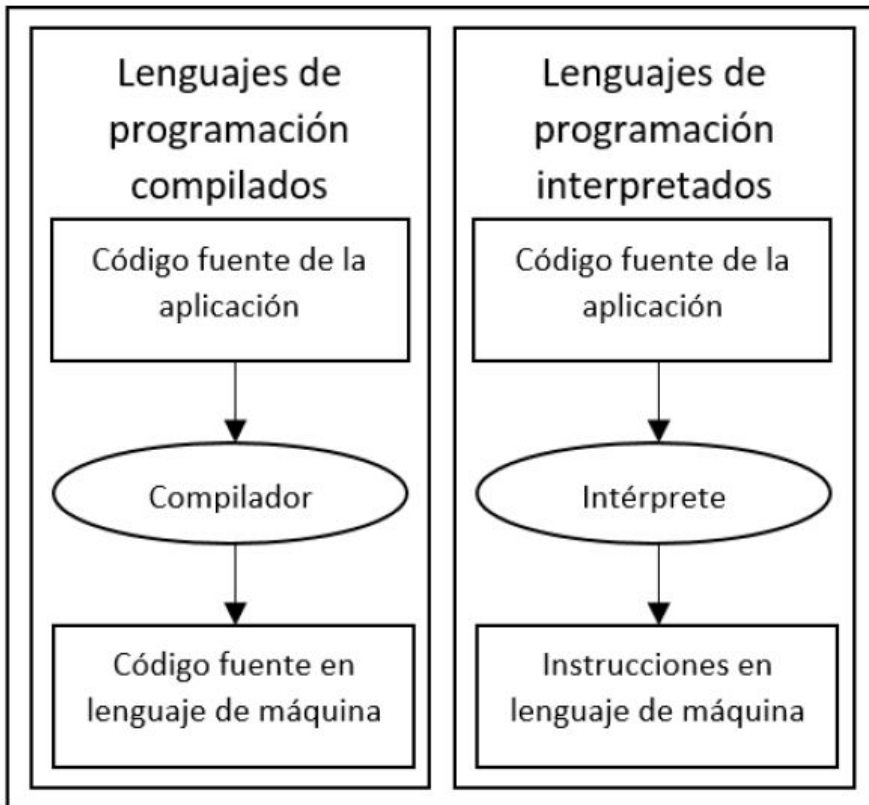
NIVELES DE ABSTRACCIÓN EN LOS LENGUAJES DE PROGRAMACIÓN

- Lenguajes de programación de bajo nivel: Lenguaje de máquina y los lenguajes de ensamblado. Para programar en cualquiera de estos lenguajes, es necesario conocer sí o sí la arquitectura del hardware del equipo.
- Los lenguajes de programación de alto nivel son lenguajes de programación que tienen como característica principal, la abstracción del hardware del equipo. Dentro de los lenguajes de programación de alto nivel también pueden encontrarse distintos niveles de abstracción ya que algunos permiten el manejo del hardware directamente (por ejemplo, c).



DISTINTOS TIPOS DE LENGUAJES DE PROGRAMACIÓN: COMPILADOS E INTERPRETADOS

Lenguajes de programación de alto nivel



- Los lenguajes de programación compilados son aquellos que se traducen enteramente al lenguaje de máquina para ser ejecutados directamente en el hardware.
- Los lenguajes de programación interpretados son aquellos en los que necesitan de un software intermedio llamado intérprete que será el encargado de ejecutar las instrucciones.

LENGUAJES DE PROGRAMACIÓN COMPILADOS

- Los lenguajes de programación compilados son aquellos que utilizan un compilador para transformar el código fuente de la aplicación en código de máquina que pueda interpretar el sistema.
- La principal ventaja de los lenguajes compilados sobre los interpretados, es la velocidad de ejecución.
- Dentro de los lenguajes de programación de alto nivel, aquellos que tienen a tener un menor nivel de abstracción suelen ser compilados ya que, al almacenarle finalmente en lenguaje de máquina, permiten ejecutar instrucciones directamente sobre el hardware. Algunos ejemplos de estos lenguajes son C, C++, etc.
- Entre las desventajas que podemos mencionar de los lenguajes compilados, se encuentra la necesidad de compilar el código fuente antes de poder ejecutar la aplicación. Además, cada arquitectura de hardware necesitará que el código fuente sea compilado especialmente para la misma.

LENGUAJES DE PROGRAMACIÓN INTERPRETADOS

- Los lenguajes de programación interpretados son aquellos que tienen un componente llamado “intérprete”, que será el encargado de traducir el código fuente a código de máquina, pero en lugar de almacenarlo en un archivo, ejecutará las instrucciones directamente.
- La principal desventaja de este tipo de lenguajes es la velocidad de ejecución.
- La principal ventaja radica en que no es necesario “compilar” la aplicación para las distintas arquitecturas.

CLASIFICACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN DEPENDIENDO DEL PARADIGMA

- Dentro del ámbito del desarrollo de software, podemos encontrar distintos paradigmas o “formas” de programación.
- Existen paradigmas orientados a la metodología de trabajo, que imponen reglas o formas de trabajar desde lo práctico. Por otro lado, podemos identificar paradigmas de programación que dependen del lenguaje.
- Algunos ejemplos de distintos paradigmas son:
 - Programación estructurada
 - Programación Orientada a Objetos
 - Programación funcional