

Resumen de paradigmas de la programación 2019

¿Qué es un paradigma?

- Es una forma de resolver algo. La fundamentación de la forma de resolver un problema.
- Aquello que es la base aceptada sobre algo.
- Refleja algo específico que funciona como ejemplo a seguir. Se utiliza para señalar aquello que funciona y es tomado como ejemplo.

Paradigma de programación

- Consiste en un método para llevar a cabo cálculos y la forma en que la deben estructurarse y organizarse las tareas que debe realizar un programa.
- Es un estilo de desarrollo de programas, es decir, un modelo para resolver problemas computacionales.

Principales paradigmas de programación

- Orientado a objetos: Es un intento de abstracción del mundo real, el comportamiento del programa es llevado a cabo por objetos, entidades que representan elementos del problema a resolver y tienen atributos y comportamiento.
- Imperativo: los programas se componen de un conjunto de sentencias que cambian su estado. Son secuencias de comandos que ordenan acciones a la computadora
- Declarativo: opuesto al imperativo, los programas describen los resultados esperados (el que) sin listar explícitamente los pasos (el cómo) a llevar a cabo para alcanzarlos.
- Funcional: los programas se componen de funciones, es decir, implementaciones de comportamiento que reciben un conjunto de datos de entrada y devuelven un valor de salida.
- Lógico: El problema se modela con enunciados de lógica de primer orden.

Lenguaje Imperativo vs. Declarativo

En la programación imperativa, de la cual hacen parte muchos de los principales lenguajes de programación tales como C, Java y PHP, un programa se describe en términos de instrucciones, condiciones y pasos que modifican el estado de un programa al permitir la mutación de variables, todo esto con el objetivo de llegar a un resultado. En contraparte, en la programación declarativa un programa se describe en términos de proposiciones y afirmaciones que son declaradas para describir el problema, sin especificar los pasos para resolverlo; en este tipo de programas, el estado no puede ser modificado ya que todos los tipos de datos son inmutables. De esta familia hacen parte lenguajes como Scala, Haskell, Erlang y Elixir.

Lenguaje Compilado vs. Interpretado

Lenguaje compilado: el código fuente es analizado por un compilador que examina la lógica y sintaxis y genera un archivo ejecutable. Interpretado: se tiene un intérprete que lo que hace es leer el archivo y hacer la traducción (interpretar) lo que entiende, normalmente lo hace línea por línea.

Tipado Estático vs. Dinámico

Tipado estático: la comprobación de la tipificación se realiza durante la **compilación**. Tipado dinámico: la comprobación de la tipificación se realiza durante la **ejecución**.

Tipado débil vs. Tipado Fuerte

Tipado fuerte: no se permiten violaciones de los tipos de datos. Tipado débil: no controlan los tipos de las variables que declaran, de este modo, es posible usar variables de cualquier tipo en un mismo escenario.

Paso de parámetros

- **Por valor:** se crea una copia del argumento y se asigna a esa variable local.
- **Por referencia:** el parámetro formal es una variable local que almacena una referencia al argumento.

Características de los lenguajes de programación

Lenguaje	Paradigma	Imperativo /Declarativo	Estático /Dinámico	Débil /Fuerte	Compilado /Interpretado	Paso de parámetros	Sensible a Mayus/Minus
Java	OO	Imperativo	Estático	Fuerte	Compilado	Por Valor	Si
Assembler	Imperativo	Imperativo	Dinámico	Débil	Interpretado	Referencia	No
Pascal	Procedural	Imperativo	Estático	Fuerte	Compilado	Ambos	No
Racket	Funcional	Declarativo	Dinámico	Débil	Interpretado	Referencia	Si
Prolog	Lógico	Declarativo	Dinámico	Débil	Interpretado	Referencia	Si

Traductores

Son programas que traducen a su vez los programas fuente escritos en lenguajes de alto nivel a código máquina. Los traductores se dividen en:

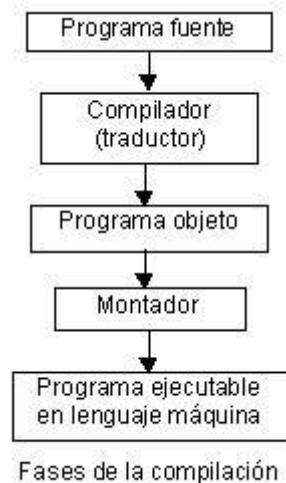
- Compiladores
- Interpretes

Compiladores

Un compilador es un programa que traduce los programas fuente escritos en lenguajes de alto nivel a lenguaje máquina.

Los programas escritos en lenguajes de alto nivel (en el editor del lenguaje) se llaman programas fuente y el programa traducido programa objeto o código objeto. El compilador traduce (sentencia a sentencia) el programa fuente.

Lenguajes compiladores típicos son: PASCAL, COBOL, C.



Fases de la compilación

Interpretes

Un intérprete es un traductor que toma un programa fuente, lo traduce y a continuación lo ejecuta (dicho programa por medio de la computadora desarrolla una tarea específica).

Un lenguaje que soporte un traductor de tipo intérprete se denomina lenguaje interpretado. BASIC es el modelo por excelencia interpretado.

Los programas fuente en BASIC se escriben con ayuda de un programa denominado editor que suele venir incorporado al programa intérprete.



Interprete