



# MATERIA:

## Paradigmas de Programacion

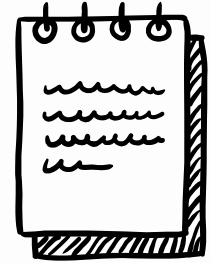
Modulo: Definición de paradigmas de programación

Docente: Ing. Luis A. Barrera

Grupo: 01 al 05

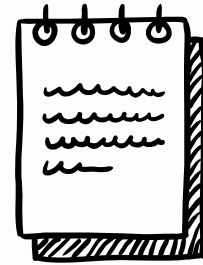
**UNIVERSIDAD DE EL SALVADOR**

**Facultad Multidisciplinaria de Occidente  
Departamento de Ingeniería y Arquitectura**



## Definición de Paradigmas de Programación

Un paradigma de programación **es una manera o estilo de programación** de software. Existen diferentes formas de diseñar un lenguaje de programación y varios modos de trabajar para obtener los resultados que necesitan los programadores.

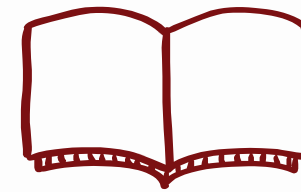


## Definición de Paradigmas de Programación

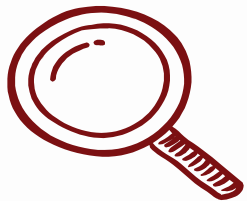
Se trata de un **conjunto de métodos sistemáticos** aplicables en todos los niveles del diseño de programas para resolver problemas computacionales.

**Los lenguajes de programación adoptan uno o varios paradigmas** en función del tipo de órdenes que permiten implementar como, por ejemplo, **Python o JavaScript**, que son multi-paradigmas.

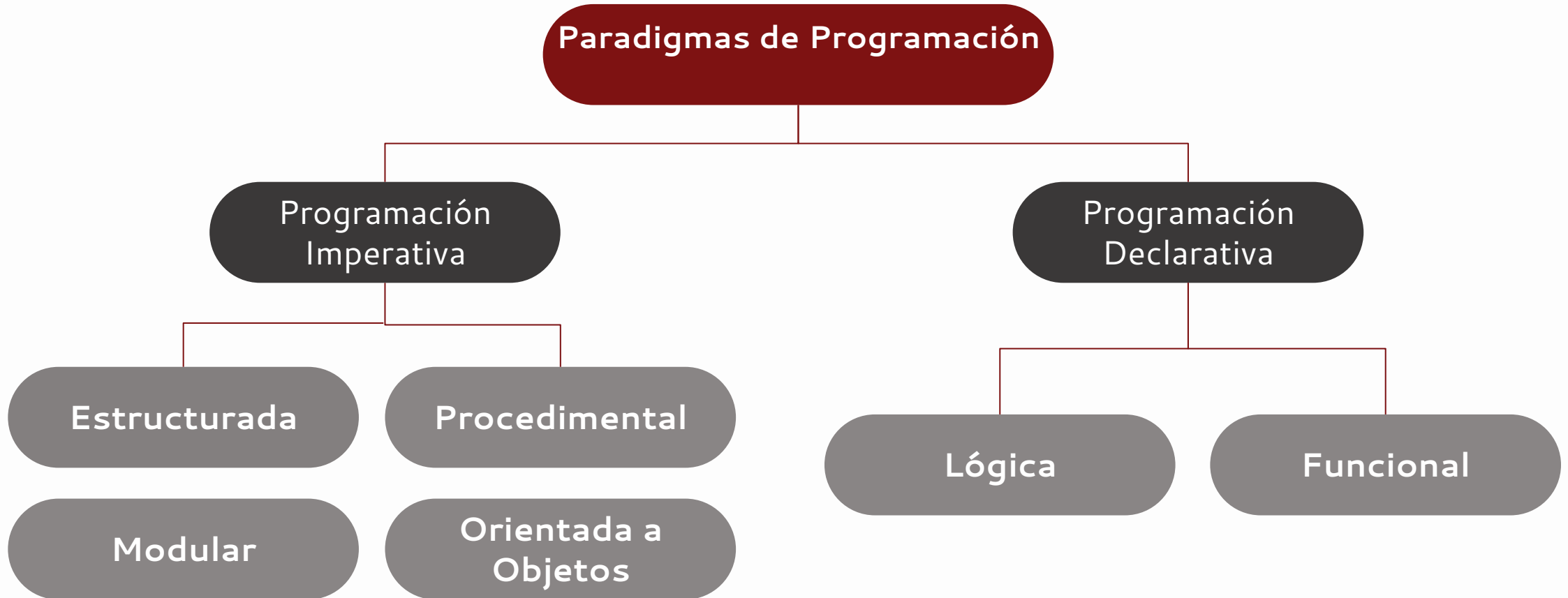
Los paradigmas de programación son los principios fundamentales de la programación de software.



Lo más fácil es plantearlos como **estilos de programación fundamentalmente diferenciados** que, en consecuencia, generan códigos software que están estructurados de forma distinta.

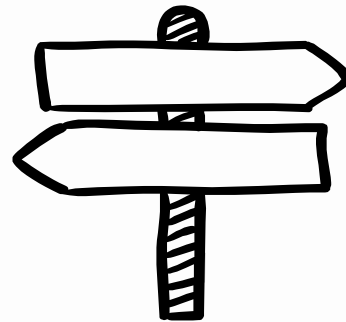


# Paradigmas de Programación



# Dos grandes tipos de Paradigmas de Programación

El concepto clásico es el de la **Programación Imperativa**, en la que se define claramente en el código fuente qué pasos debe ejecutar un programa y en qué secuencia. Sus subtipos son la programación procedimental y la orientada a objetos.



Según el principio de la **Programación Declarativa**, solo se describe lo que debe solucionar el software (es decir, solo el resultado y no los pasos individuales). Subtipos del paradigma declarativo son la programación funcional y la lógica. ¿Cómo se distinguen entre sí los citados paradigmas de software?

## Programación imperativa: El paradigma de programación clásico

Entre los paradigmas de programación de software, la programación imperativa (del latín *imperare*, ordenar) se considera el paradigma clásico.

Los primeros lenguajes de programación y, por extensión, también los primeros programas informáticos, se basan completamente en este enfoque, que prevé una secuencia regularizada de órdenes o instrucciones determinadas.

Este paradigma de programación es la base, por ejemplo, de los veteranos lenguajes **Pascal** y **C**, así como de todos los **lenguajes ensambladores**, entre otros.

## Programación Imperativa: El paradigma de programación clásico

En la programación imperativa, el centro de atención radica, entre otras cosas, en trabajar lo más cerca posible del sistema. Como consecuencia, el código de programación resultante es fácil de comprender y, a la vez, muy abarcable.

La **programación estructurada**, la **procedimental** y la **modular** son tres enfoques adicionales subordinados al paradigma de programación imperativa para escribir y estructurar código software.



## **Programación Declarativa: Paradigmas de software del pasado más reciente**

De forma paralela a la evolución continuada del hardware y el software, con el enfoque declarativo se desarrolló un paradigma alternativo para la programación de código.

El principio fundamental de la programación declarativa radica en la descripción del resultado final que se busca. Por lo tanto, en primera línea se encuentra el “qué” del resultado y no el “cómo” de los pasos que llevan a la solución, como es el caso en la programación imperativa.

## Programación Declarativa: Paradigmas de software del pasado más reciente

Esto provoca que el código de la programación declarativa sea más difícil de comprender debido al alto grado de abstracción, aunque resulta muy corto y preciso.

Dentro de los subtipos del paradigma de programación declarativa, existen más diferencias que dentro del estilo imperativo.

Asimismo, su definición y clasificación no es siempre exacta.  
Los dos enfoques más importantes del paradigma de programación declarativa son la programación **funcional y la lógica**.

Paradigma de programación imperativa	Paradigma de programación declarativa
“¿Cómo?”	“¿Qué?”
Clásico.	Nueva corriente.
Un programa según este paradigma consta de instrucciones sucesivas que indican al ordenador aquello que debe computar y hacer y en qué secuencia.	Un programa según este paradigma consta de instrucciones que determinan cómo debe proceder un programa con una petición. Los cálculos se llevan a cabo mediante la manipulación de valores y el proceso se controla mediante el procedimiento de recursión.
El nombre viene del latín “imperare”, que significa “ordenar”.	El nombre viene del latín “declarare”, que significa “denominar”.
Se predetermina la vía de solución deseada.	Se predetermina el resultado deseado.
Lenguajes de programación típicos serían C, Pascal, Fortran, ALGOL y los denominados lenguajes ensambladores.	Lenguajes de programación típicos serían Lisp, ML, Haskell, F#, Prolog y Oz.