**Qué es la recursividad**

La recursividad es una técnica de programación que se utiliza para realizar una llamada a una función desde ella misma, de allí su nombre. El ejemplo más utilizado por su fácil comprensión es el cálculo de números factoriales.

¿Dónde se aplica la recursividad?

Recursión (ciencias de computación) La mayoría de los lenguajes de programación dan soporte a la recursión permitiendo a una función llamarse a sí misma desde el texto del programa. Los lenguajes imperativos definen las estructuras de loops como while y for que son usadas para realizar tareas repetitivas.

**Cuáles son los tipos de recursividad que existen**

Recursividad simple: Aquella en cuya definición sólo aparece una llamada recursiva. Se puede cambiar a una función con ciclo iterativo.

Recursividad múltiple: Se da cuando hay más de una llamada a sí misma dentro del cuerpo de la función.

Recursividad anidada: En algunos de los argumentos de la llamada recursiva hay una nueva llamada a sí misma.

Recursividad cruzada o indirecta: Son algoritmos donde una función provoca una llamada a sí misma de forma indirecta, a través de otras funciones.

Ventajas:

– Soluciona problemas recurrentes

– Son programas cortos.

Desventajas:

– Creación de muchas variables

– Puede necesitar mucha memoria

<https://sites.google.com/site/portafoliosenati/fundamentos-de-programacion/13-aplicar-recursividad>

<http://www.utm.mx/~mgarcia/PE4(Funciones).pdf>

<https://www.uv.mx/personal/ocastillo/files/2011/04/Recursividad.pdf>