# Recursividad

En programación se utilizan definiciones recursivas. Decimos que una función o acción es recursiva cuando contiene una llamada a si misma. El ejemplo prototípico que se utiliza con frecuencia para explicar la recursividad: el cálculo del factorial de *n*.
El factorial de 0 es, por definición, 1. Los factoriales de números mayores se calculan mediante la multiplicación de 1 * 2 * ..., incrementando el número de 1 en 1 hasta llegar al número para el que se está calculando el factorial.

Generalmente, si la primera llamada al subprograma se plantea sobre un problema de tamaño u orden N, cada nueva ejecución recurrente del mismo se planteará sobre problemas, de igual naturaleza que el original, pero de un tamaño menor que N. De esta forma, al ir reduciendo progresivamente la complejidad del problema a resolver, llegará un momento en que su resolución sea más o menos trivial (o, al menos, suficientemente manejable como para resolverlo de forma no recursiva). En esa situación diremos que estamos ante un caso base de la recursividad.

Es frecuente que los algoritmos recurrentes sean más ineficientes en tiempo que los iterativos aunque suelen ser mucho más breves en espacio.

## Recursividad directa vs indirecta.

Cuando en una subrutina hay llamadas a ella misma se habla de recursividad directa, en contraposición, cuando se tienen varias subrutinas y éstas se llaman unas a otras formando ciclos se dice que la recursión es indirecta.

Subrutina_A → Subrutina_A → Subrutina_A
Subrutina_A → Subrutina_B → Subrutina_C → Subrutina_D → Subrutina_A

Programación Recursiva:
Es mucho más difícil desarrollar una solución recursiva en un lenguaje determinado para resolver un problema especifico cuando no se tiene un algoritmo. No es solo el programa sino las definiciones originales y los algoritmos los que deben desarrollarse. En general, cuando encaramos la tarea de escribir un programa para resolver un problema no hay razón para buscar una solución recursiva. La mayoría de los problemas pueden resolverse de una manera directa usando métodos no recursivos. Sin embargo, otros pueden resolverse de una manera más lógica y elegante mediante la recursión.

Bibliografía: