

Recursividad

Recursividad se puede definir como una técnica que permite que una función se llame así misma.

Son Recursivos aquellos algoritmos que, estando encapsulados dentro de una función, son llamados desde ella misma una y otra vez, en contraposición a los algoritmos que hacen uso de bucles while, do-while, for.

Podemos distinguir dos tipos de recursividad:

Recursividad Múltiple: Se da cuando hay más de una llamada a sí misma dentro del cuerpo de la función, resultando más difícil de hacer de forma iterativa.

Recursividad Simple: Aquella en cuya definición sólo aparece una llamada recursiva.

Se puede transformar con facilidad en algoritmos iterativos.

Consta de una parte recursiva, otra iterativa o no recursiva y una condición de terminación. La parte recursiva y la condición de terminación siempre existen. En cambio la parte no recursiva puede coincidir con la condición de terminación.

Algo muy importante a tener en cuenta cuando usemos la recursividad es que es necesario asegurarnos que llega un momento en que no hacemos más llamadas recursivas. Si no se cumple esta condición el programa no parará nunca.

Referencias:

<http://jitorres.blogspot.com/2011/08/recursividad.html>