

# Recursividad

Para que un problema pueda ser resuelto de modo recursivo ha de poseer las siguientes dos características:

- El problema original se ha de poder reducir al mismo problema y con una talla menor. Debe existir una caso base o condición de parada.

Por ejemplo, la función factorial se define en los siguientes términos: •  $F(n) = n * F(n-1)$ ; •  $F(0) = 1$ ; La primera parte de la definición de la función factorial nos dice que se puede calcular el factorial de un número multiplicando ese número por el factorial del número menos uno, se ha reducido el problema al 21 mismo problema pero con una talla menor.

La segunda parte de la definición corresponde al caso base, que fuerza la salida de la recursividad. Java soporta la programación recursiva.

A continuación se muestra una posible solución recursiva para el cálculo de la función factorial:

```
public class Factorial { public static void main(String args[]) { long n = Long.parseLong(args[0]); System.out.println("El factorial de " + n + " es: " + Factorial(n)); } private static long Factorial(long n) { long resultado; if(n > 0) resultado = n * Factorial(n-1); else resultado = 1; return resultado; } }
```

## Bibliografías

Oscar Belmonte Fernández, Introducción al lenguaje de programación Java.2005