Ejemplos de recursividad

Ejemplo 1: Cálculo del factorial

Podemos definir el factorial de un entero positivo n ($n!$) como el producto de n con el factorial de $n - 1$: $n! = n \cdot (n - 1)!$. Esta definición deberá tener un límite: el momento en el que se llega al valor cero: el factorial de cero es, por definición, igual a uno.

Ejemplo 2: El algoritmo de Euclides

El máximo común divisor de dos números m y n ((m, n)) se puede definir como el máximo común divisor del segundo (de n) y del resto de dividir el primero con el segundo: $mcd\ m, n = mcd(n, m\ mod\ n)$. La secuencia se debe parar cuando se alcanza un valor de segundo entero igual a cero: de lo contrario en la siguiente vuelta se realizaría una división por cero.

Conclusiones del Algoritmo de Euclides: Los divisores comunes de dos números naturales a y b, con a $\geq$b y b$\neq$0 , son los divisores del último resto no nulo, de las sucesivas divisiones entre a y b y los restos.

Ejemplo 3: Potencia de exponente entero no negativo

La función `potencia (b,n)` es posible optimizarla teniendo en cuenta que:

- b^n = b^(n/2) × b^(n/2) si n es par.
- b^n = b^(n−1)/2 × b^(n−1)/2 × b si n es impar.

Para esta función, tomaremos n = 0 como el caso base, en el que devolveremos 1 (b elevado a cero es 1); y el caso recurrente tendrá dos partes, correspondientes a los dos posibles grupos de valores de n (n par / n impar).

Ejemplo 4: Números de fibonacci

Un ejemplo clásico en el cual la recursividad tiene un **resultado muy poco eficiente** es el de los números de fibonacci.

Esta sucesión fue descrita por Fibonacci como la solución a un problema de cría de conejos: "*Cierto hombre tiene una pareja de conejos juntos en un lugar cerrado y desea saber cuántos son creados a partir de este par en un año cuando, de acuerdo a su naturaleza, cada pareja necesita un mes para envejecer y cada mes posterior procrea otra pareja*" (Laurence Sigler, Fibonacci's Liber Abaci, página 404).

La sucesión de fibonacci está definida por la siguiente relación:

fib(0) = 0

fib(1) = 1

fib(n) = fib(n-1) + fib(n-2)

Podemos definir la serie de Fibonacci como aquella cuyo término n es igual a la suma de los términos $(n - 1)$ y $(n - 2)$: $Fibn = Fibn\text{-}1 + Fibn\text{-}2$. Y de nuevo se tiene el límite cuando $n \leq 2$ donde el valor del término es la unidad.

Los primeros números de esta sucesión son: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55.

Funciona pero es poco eficiente, la complejidad aumenta exponencialmente a medida que el número que buscamos, *n*, es mayor, pues se trata de recursividad en árbol.

Ejemplo 5: Convertir un número decimal en número binario (base 2).

Caso base: Número menor que 2, ya que 1 en binario es 1 y 0 en binario es 0.

Caso recurrente: Hay que tomar el módulo de la división del número decimal entre dos y llamar de nuevo.

Ejemplo 6: Cálculo del cociente en la división de dos números enteros.

Caso Base: Si el dividendo es menor que el divisor el cociente es cero.

Caso recurrente: Se restan el dividendo y el divisor. Al resultado se le vuelve a restar el divisor. Esta resta se repite mientras el resultado sea mayor o igual que el divisor. El número de veces que se realiza la resta es el cociente.