Algoritmia - Eficiencia de Algoritmos
Octubre 2015

¿Si $f(n) \in \Omega(n)$ entonces $f(n) \in O(n^2)$?

No porque para $f(n) = n^3$, sí se cumple $f(n) \in \Omega(n^2)$ ya que $n^2$ es cota inferior de $n^3$ pero no se cumple que $f(n) \in O(n^2)$ ya que $n^2$ no es cota superior de $n^3$.

Cotas de eficiencia.

Sup $O(1) \subset O(\log n) \subset O(n^{1/2}) \subset O(n) \subset O(n \log n) \subset O(n^2) \subset O(n^3) \subset O(2^n) \subset O(n^n)$

Inf $\Omega(n^n) \subset \Omega(2^n) \subset \Omega(n^3) \subset \Omega(n^2) \subset \Omega(n \log n) \subset \Omega(n) \subset \Omega(n^{1/2}) \subset \Omega(\log n) \subset \Omega(1)$

¿Qué podemos decir si se cumple que $f(n) \in \Omega(n^2)$ y $f(n) \in O(n^2)$?

Que $f(n)$ es exactamente de orden $n^2$ ya que $\Omega(n^2)$ y $O(n^2)$ son exact. igual.

Demostrar sin utilizar límites que $f(n) = n^2+1$ es del mismo orden que $f(n) = 3n^2+3$

$\exists c \in \mathbb{R}^+ \wedge \exists n_0 \in \mathbb{N} \mid n^2+1 \geq c(3n^2+3) \ \forall n \geq n_0$, se cumple con $c = 1/3$ y $n_0 = 1$

$\exists c \in \mathbb{R}^+ \wedge \exists n_0 \in \mathbb{N} \mid 3n^2+3 \geq c(n^2+1) \ \forall n \geq n_0$, se cumple para $c = 1$ y $n_0 = 1$

Se demuestra que $3n^2+1$ es cota superior de $n^2+1$ y que $n^2+1$ es cota superior de $3n^2+3$.

¿$\log n$ y $(\log n)^2$ representan órdenes de complejidad equivalentes?

$$\lim_{n \to \infty} \frac{(\log n)^2}{\log n} = \lim_{n \to \infty} \log n = \infty$$ , luego $(\log n)^2$ domina asintóticamente a $\log n$

```
Funcion Ejercicio1 (num: entero, v[1..num]: vector de enteros) retorna entero
    var aux: entero fvar
    para i desde 2 hasta num hacer
        aux = v[i]
        j = i
        mientras (j > 1 && v[j-1] > aux) hacer
            v[j] = v[j-1]
            j--;
        fmientras
        v[j] = aux;
    fpara
    retorna 0;
ffuncion
```

Talla: num , Existe mejor y peor caso: sí.

Mejor caso: todos los elementos del vector están ordenados en orden creciente, con lo que no se entra en el bucle mientras.

$$T_{MC}(num) = \sum_{i=2}^{num} 3 + \underbrace{1}_{ret.} = \underbrace{3(num-1)}_{(num-2(1)+1)} + 1 \in \Theta(num)$$

Peor caso: $v[1] > v[2] > v[3] \ldots$ esto es, todos los elementos del vector están en orden decreciente, estrictamente. El bucle mientras se ejecuta desde 2 hasta i en todas las iteraciones del bucle para, donde i es la variable de dicho bucle.

$$T_{PC}(num) = \sum_{i=2}^{num}\left(3 + \sum_{j=2}^{i} 2\right) + 1 = \sum_{i=2}^{num}(3 + (2i-1)) + 1 =$$

$$= \sum_{i=2}^{num} 3 + \sum_{i=2}^{num}(2i-1) + 1 = 3(num-1) + 2\sum_{i=1}^{num-1} i + 1 = 3num - 3 + 1 +$$

$$+ 2\frac{(num-1)(1+num-1)}{2} \in \Theta(num^2)$$

Funcion Ejercicio2 (n : entero) retorna (b : entero)
 si (n=1) o (n=2) entonces retorna n
 si no retorna 2 * Ejercicio2 (n-1) + 3 * Ejercicio2 (n-2);
 fsi
ffuncion

Talla : n

∃ mejor y peor caso : no

Ecuación de recurrencia recursiva:

$$\begin{cases} C1 & \text{si } n \leq 2 \\ \underbrace{T(n-1)+T(n-2)}+C2 & \text{si } n>2 \end{cases}$$

$$T(n) = T(n-1) + T(n-2) + C2 =$$
$$= T(n-2) + T(n-3) + C2 + T(n-3) + T(n-4) + C2 = \ldots$$

El número de términos aumenta conforme nos acercamos a la base, por lo que es difícil establecer una fórmula general. Para solventar este problema podemos acotar $T(n)$ con las ecuaciones $T1(n)$ y $T2(n)$:

$$T_1(n) = \begin{cases} C_1 & \text{si } n \leq 2 \\ 2T(n-1) + C_2 \end{cases} \qquad T_2(n) = \begin{cases} C_2 & n \geq 2 \\ 2T(n-2) + C_2 & \text{si } n > 2 \end{cases}$$

$$T_1(n) = 2T(n-1) + C_2 = 2[2T(n-2) + C_2] + C_2 = 2^2 T(n-2) + 3C_2 =$$
$$= 2^2[2T(n-3) + C_2] + 3C_2 = 2^3 T(n-3) + 7C_2 \ldots$$
$$= 2^i T(n-i) + (2^i - 1)C_2$$

Se alcanza la base cuando $n - i \leq 2$ y esto ocurre cuando $i \approx n$. Reemplazamos

$n - i = 2, \; i = n-2$

$2^n C_1 + (2^n - 1) C_2 \in \Theta(2^n)$

$2^n T(n-(n)) =$
$2^n T(0) =$
$\underline{2^n C_1}$

$$T_2(n) = 2T(n-2) + C_2 = 2[2T(n-4) + C_2] + C_2 = 2^2 T(n-4) + 3C_2 =$$
$$= 2^2[2T(n-6) + C_2] + 3C_2 = 2^3 T(n-6) + 7C_2 = \ldots$$
$$= 2^i T(n-2i) + (2^i - 1)C_2$$

$n - 2i = 2$
$n - 2 = 2i$
$-1 + n/2 = i$

Se alcanza la base para $n - 2i \leq 2$, luego esto ocurre cuando $i \approx n/2$, sustituimos

$2^{n/2} C_1 + (2^{n/2} - 1) C_2 \in \Theta(2^{n/2})$

$2^n T(n - 2n/2)$
$2^n T(0)$
$2^n C_1$

$T_1 \in \Theta(2^n)$ y $T_2 \in \Theta(2^{n/2})$

$\lim\limits_{n \to \infty} \dfrac{2^n}{2^{n/2}} = \lim\limits_{n \to \infty} 2^{n \operatorname{div} 2} = \infty$, $T(n) \in \Omega(2^{n/2})$ y $T(n) \in O(2^n)$

$n - n/2$
$n/2$