David Martínez Díaz 44669141S

1. Probar para dos funciones arbitrarias f y $g: N \rightarrow R^*$ que,

a) $\Theta(f(n)) = O(g(n))$ si y solo si $f(n)$ es $O(g(n))$ y $g(n)$ es $O(f(n))$

b) $O f((n)) \subset O(g(n))$ si y solo si $f(n)$ es $O(g(n))$ y $g(n)$ no es $O f(n)$

---

a) Si para en este caso tenemos que:

Para que la función $f(n)$ es $O(g(n))$ tienen que existir unas constantes positivas c, d y n_0 tales que:

$c g(n) \leq f(n) < d g(n) \Rightarrow \forall n \geq n_0$ tenemos que se cumple la siguiente condición:

$\left.\begin{array}{l} f(n) \Rightarrow O(g(n)) \\ g(n) \Rightarrow O(f(n)) \end{array}\right\}$ Por tanto nos queda $\Theta(f(n)) = O(g(n))$

Teniendo como conclusión que $f(n)$ es $\Theta(g(n))$ y que $g(n)$ es $\Theta(f(n))$.

---

b) Para que se cumpla dicha condición tenemos que $f(n)$ es $O(g(n))$, que necesita constantes positivas c:

$\boxed{f(n) \leq c\, g(n) \quad \forall n \geq n_0}$

Y por el otro lado tenemos que $g(n) \notin O(f(n))$ o también se puede decir que $g(n) \in \Omega(f(n))$, si existen valores constantes positivos tal que:

$\boxed{g(n) \geq c \cdot f(n) \quad \forall n \geq n_0}$

* Quedando demostrado dicha condición:

$$\left\{\begin{array}{l} f(n) = O(g(n)) \\ g(n) = \Omega(f(n)) \Rightarrow g(n) \neq O(f(n)) \end{array}\right\}$$

Escaneado con CamScanner

2. Si el tiempo de ejecución de un algoritmo A está descrito por la recurrencia: $T(n) = aT(n/2) + n^2$

y el de otro algoritmo B por recurrencia:

$T'(n) = bT'(n/8) + n^2$

¿Qué valores de a y b hacen a cada algoritmo asintóticamente más rápido que el otro?

→ Primer algoritmo: A.

$T(n) = aT(n/2) + n^2$; ⇒ Cambio de variable $n = 2^k$

↳ $T(2^k) = a\,T(2^{k-1}) + (2^k)^2$

↳ $T_k = a T_{k-1} + 4^k$

↳ Ecuación Característica: $(x-a)(x-4)$

$t_k = A a^k + B \cdot 4^k \Rightarrow T_n = A a^{\log_2 n} + B \cdot 4^{\log_2 n}$;

↳ $T_n = A \cdot n^{\log_2 a} + B n^2$;

→ Segundo Algoritmo: B

$T'(n) = bT'(n/8) + n^2$; ⇒ Cambio de variable $n = 8^k$

↳ $T(8^k) = b\,T'(8^{k-1}) + (8^k)^2$

↳ $T_k = b T_{k-1} + 64^k$;

↳ Ecuación Característica: $(x-b)(x-64)$

$T_k = A b^k + B \cdot 64^k \Rightarrow T_n = A b^{\log_8 n} + B\, 64^{\log_8 n}$;

↳ $T_n = A n^{\log_8 b} + B n^2$;

* Para comparar ambas ecuaciones, se nos queda que solo varía en los logaritmos que están elevados y se nos queda de tal forma:

$\log_2 a = \log_8 b \Rightarrow \dfrac{\log a}{\log 2} = \dfrac{\log b}{\log 8}$; $\left\{ \begin{array}{l} \log a = \dfrac{\log b}{\log 8} \cdot \log 2 \\ \\ \log b = \dfrac{\log a}{\log 2} \cdot \log 8 \end{array} \right\}$

*Quedando dichas variables en función de la otra.