2.1 Todos los casos son iguales

$T(n) = 2T(n/2) + 1$

$\downarrow n = 2^m$

$T(2^m) - 2T(2^{m-1}) = 1$

$\downarrow x = T(2^m) \qquad \downarrow 1 = b^m \cdot p(m) \quad \begin{cases} b = 1 \\ p(m) = 1 \\ d = 0 \end{cases}$

Ec. característica: $(x-2)(x-1) = 0$

$T(2^m) = c_1 \cdot 1^m + c_2 \cdot 2^m$

$T(n) = c_1 \cdot 1^{\log_2 n} + c_2 \cdot 2^{\log_2 n}$

$T(n) = c_1 \cdot n^0 + c_2 \cdot n^1 = c_1 + c_2 \cdot n$

Como c_1 y c_2 son mayores que 0, $T(n) \in \Theta(n)$

2.3

Aunque se haya añadido la "ordenación" de los repetidos en la función Pivote, su orden de eficiencia sigue siendo de **n** para todos los casos. Por tanto:

Mejor caso: $T(n) = 2T(n/2) + 2n$ (se parten los vectores por la mitad)

$\downarrow n = 2^m$

$T(2^m) - 2T(2^{m-1}) = 2^{m+1}$

$\downarrow x = T(2^m) \qquad \downarrow 2 \cdot 2^m = b^m \cdot p(m) \quad \begin{cases} b = 2 \\ p(m) = 2 \\ d = 0 \end{cases}$

Ec. característica: $(x-2)(x-2) = 0$

$T(2^m) = c_1 \cdot 2^m + c_2 \cdot m \cdot 2^m$

$T(n) = c_1 \cdot n + c_2 \cdot \log_2(n) \cdot n$

Como c_1 y c_2 son mayores que 0, $T(n) \in \Omega(n \cdot \log_2(n))$

Juan Manuel Rubio Rodríguez
Ignacio Vellido Expósito

Peor caso: Todos los valores restantes del vector son mayores o menores que el pivote

$T(n) = T(n-1) + 2n$

$\downarrow$

$T(n) - T(n-1) = 2n$

$\downarrow x = T(n)$ $\quad \downarrow 2n = b^n \cdot p(n)$ $\quad \begin{cases} b = 1 \\ p(n) = 2n \\ d = 1 \end{cases}$

$(x-1)(x-1)^2 = 0$ (Ec. característica)

$T(n) = c_1 \cdot 1^n + c_2 \cdot n \cdot 1^n + c_3 \cdot n^2 \cdot 1^n$

$T(n) = c_1 + c_2 \cdot n + c_3 \cdot n^2$

Como c_1, c_2 y c_3 son mayores que 0, $T(n) \in O(n^2)$

Caso promedio: Podemos suponer que equivaldría al mejor caso (generando números aleatorios)

(4) La función Pivote equivale a la del ejercicio 2.3, con orden de eficiencia n. Entonces:

Caso peor: Todos tienen la misma cardinalidad

$T(n) = 1 + \sum_{1}^{n} (\underset{\text{Pivote}}{n} + 1) = 1 + n + n^2 \in O(n^2)$

Caso mejor: El primer pivote tiene la mayor cardinalidad (y mayor que el resto del conjunto)

$T(n) = 1 + \sum_{1}^{1} (n+1) = 1 + n + 1 \in \Omega(n)$

Juan Manuel Rubio Rodríguez
Ignacio Vellido Expósito

Ejercicio 2.2

$T(n) = 4T(n/2) + 1$

Se simplifica con un +1 porque el bucle for siempre realiza 4 iteraciones (O(1)) y los bloques if asumimos que son igualmente O(1).

Se llama recursivamente a la función 4 veces con ¼ del tamaño del problema.

$n=2^m \rightarrow m=\log(n)$

$T(2^m) = 4T(2^{m-1}) + 1$

$T(2^m) - 4T(2^{m-1}) = 1$ b=1; p(m)=1; d=0;

$X^1 = T(2^m)$

$(x-4)(x-1)$

$T(2^m) = c_1 * 4^m + c_2 * 1^m$

$T(n) = c_1 * n^2 + c_2 * 1^{\log(n)}$

$T(n) = c_1 * n^2 + c_2 * n^{\log(1)} = c_1 * n + c_2$

A expensas de calcular las dos constantes, vemos que la recurrencia es de orden $\mathbf{n^2}$.

Juan Manuel Rubio Rodríguez
Ignacio Vellido Expósito

Ejercicio 5

$T(n) = 4T(n/2) + 2*(n*b^3) + 1$

Siendo **b** el tamaño de la forma, comprobarForma es de eficiencia b^2 y se repite 2*n*b veces (dos veces el tamaño del barrido).

$n=2^m \rightarrow m=\log(n)$

$T(2^m) = 4T(2^{m-1}) + 2*(2^m)*b^3 + 1$

$T(2^m) - 4T(2^{m-1}) = 2*(2^m)*b^3 + 1$ b=1; p(m)=1; d=0;

b=2; $p(m)=2*b^3$; d=0;

$X^1 = T(2^m)$

(x-4)(x-1)(x-2)

$T(2^m) = c_1 * 4^m + c_2 * 1^m + c_3 * 2^m$

$T(n) = c_1 * n^2 + c_2 * 1^{\log(n)} + c_3 * n$

$T(n) = c_1 * n^2 + c_2 * n^{\log(1)} + c_3 * n = c_1 * n^2 + c_2 + n * c_3$

Como las **c** son positivas, la recurrencia es de orden $\mathbf{n^2}$.

Juan Manuel Rubio Rodríguez
Ignacio Vellido Expósito