



Práctica 4 - Resolución de Recurrencias

1. Muestre utilizando el método de sustitución que la solución de la recurrencia

$$T(n) = T(\lfloor n/2 \rfloor) + 1$$

es $O(\lg n)$

2. Sean $a, b \in \mathbb{R}^+$, utilizar el método de sustitución para encontrar cotas asintóticas Θ para las siguientes recurrencias:

$$T(n) = \begin{cases} a & n = 1 \\ 2T(\lfloor n/2 \rfloor) + n & n > 1 \end{cases}$$

3. Utilice un árbol de recurrencia para encontrar una cota asintótica para la recurrencia

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

donde c es una constante. Verifique que la cota encontrada es correcta.

4. Use un árbol de recurrencia para obtener una cota asintótica para

$$\begin{aligned} T(n) &= T(n-a) + T(a) + cn & \text{si } n > a \\ T(n) &= c' & \text{si } n \leq a \end{aligned}$$

donde $a \geq 1$, $c > 0$ son constantes.

5. Use el Teorema Maestro para encontrar cotas asintóticas para las siguientes recurrencias (asuma $T(1) > 0$):

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

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

c) $T(n) = 4T(n/2) + n^3$

6. Encuentre cotas asintóticas inferiores y superiores lo más justas posible para cada una de las siguientes recurrencias (asuma $T(1) > 0$):

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

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

7. Dadas las siguientes definiciones en Haskell que implementan distintos algoritmos para multiplicar, calcular el trabajo de cada una de ellas y determinar que función es más eficiente resolviendo sus recurrencias:

```
producto1 n 1 = n
producto1 n m = n + producto1 n (m - 1)

producto2 n 1 = n
producto2 n m = case (even m) of
  True  -> producto2 (2 * n) (div m 2)
  False -> n + producto2 (2 * n) (div m 2)
```

8. Escribir las recurrencias del trabajo y la profundidad de las siguientes funciones y resolverlas.

```
data Tree a = E | L a | N (Tree a) (Tree a)

split [] = ([], [])
split [x] = ([x], [])
split (x:y:xs) = let (ys, zs) = split xs
                  in (x:ys, y:zs)

toTree [] = E
toTree [x] = L x
toTree ls@(x:y:xs) = let (ys, zs) = split ls
                      (l, r) = toTree ys (|||) toTree zs
                      in Node l r
```