

Metodología y tecnología de la programación

Ingeniería Informática (UCM)

Hoja de ejercicios 1

Curso 2007/2008

Ejercicio 1 Demostrar por inducción sobre $n \geq 0$ las siguientes igualdades:

1. $\sum_{i=1}^n i = n(n+1)/2$.
2. $\sum_{i=1}^n i^2 = n(n+1)(2n+1)/6$.

Ejercicio 2 Demostrar en detalle:

$$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0 \quad \text{implica} \quad f \in O(g) \text{ y } g \notin O(f).$$

Ejercicio 3 Encontrar dos funciones $f, g : \mathbb{N} \rightarrow \mathbb{R}^*$ tales que $f \notin O(g)$ y $g \notin O(f)$.

Ejercicio 4 Suponiendo que todas las funciones que aparecen en los enunciados están definidas de \mathbb{N} en \mathbb{R}^* , demostrar:

1. Si $f_1 \in O(g_1)$ y $f_2 \in O(g_2)$, entonces $f_1 f_2 \in O(g_1 g_2)$.
2. $f \in O(g)$ si y solo si $g \in \Omega(f)$.
3. $\Omega(f) \subset \Omega(g)$ (inclusión estricta) si y solo si $f \in \Omega(g)$ y $g \notin \Omega(f)$.

Ejercicio 5 Hallar la solución *exacta* de las siguientes recurrencias:

$$(a) T(n) = \begin{cases} 1 & n = 1 \\ 2T(n-1) + n - 1 & n \geq 2. \end{cases} \quad (b) T(n) = \begin{cases} 1 & n = 1 \\ 6 & n = 2 \\ T(n-2) + 3n + 4 & n \geq 3. \end{cases}$$

Indicación: $\sum_{i=1}^n i 2^i = (n-1)2^{n+1} + 2$.

Ejercicio 6 Hallar la solución *exacta* de las siguiente recurrencia:

$$T(n) = \begin{cases} 1 & n = 1 \\ 2 \sum_{i=1}^{n-1} T(i) + 1 & n \geq 2. \end{cases}$$

Ejercicio 7 Supongamos que el siguiente algoritmo se ejecuta sobre un número natural n que es potencia de 5. Dar una relación de recurrencia para el número exacto de divisiones que se realizan en función de n y resolverla de forma exacta.

```
proc mickey(n : nat)
  si n = 1 entonces nada
  si no mouse := n/5
    para i := 1 hasta 4 hacer
      mickey(mouse)
    fpara
  fsi
fproc
```

Ejercicio 8 ¿Qué valor devuelve la siguiente función? Dar la respuesta como función en términos de n y calcular su complejidad.

```

fun valor(n : nat) dev r : nat
var i, j, k : nat
  r := 0
  para i = 1 hasta n - 1 hacer
    para j = i + 1 hasta n hacer
      para k = 1 hasta j hacer
        r := r + 1
      fpara
    fpara
  fpara
ffun

```