

**Prueba de análisis de algoritmos – Junio – Curso 2016-2017**

Diseño y Análisis de Algoritmos

Valor: 30 % de la nota final. Duración: **1 hora**

---

**Ejercicio 1 [2 puntos]**

Determina **mediante la definición** de  $\mathcal{O}$  si:

$$2^{\ln(n)} \in \mathcal{O}(n)$$

---

**Ejercicio 2 [4 puntos]**

La fórmula para considerar todas las operaciones que se llevan a cabo en un bucle de tipo FOR o WHILE es:

$$T_{\text{bucle}} = 1_{\text{inicialización}} + \sum^n (1_{\text{comparación}} + T_{\text{cuerpo}} + 1_{\text{incremento}}) + 1_{\text{última comparación}}$$

Utilízala para hallar el número de operaciones ( $T(n)$ , simplificada) del siguiente código en el **PEOR** caso:

```
1  for (int i=0; i<n; i++)
2    if (condicion(i)) // una operación
3      for (int j=i; j<=n; j++)
4        for (int k=0; k<j; k++)
5          procesa(i,j,k); // dos operaciones
```

---

**Ejercicio 3 [4 puntos]**

Resuelve la siguiente relación de recurrencia (se puede resolver mediante cualquiera de los dos métodos vistos en clase):

$$T(n) = \begin{cases} 0 & \text{si } n = 0 \\ T(n-1) + 2^n + 1 & \text{si } n > 0 \end{cases}$$