

# Problemas

## Recurrencias

Resolver la recurrencia:

$$T(1) = 5$$

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

a

Usando expansión de la recurrencia (de  $n$  a 1)

b

Generando valores de la recurrencia (de 1 a  $n$ )

Resolver la recurrencia:

$$T(0) = 3$$

$$T(1) = 3\sqrt{2}$$

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

a

Generando valores de la recurrencia (de 1 a n)

b

Usando la ecuación característica

Resolver la recurrencia:

$$T(0) = 1$$

$$T(1) = 1$$

$$T(n) = -2 T(n - 1) + 15T(n - 2)$$

a

Usando la ecuación característica

Resolver la recurrencia:

$$T(0) = 1$$

$$T(1) = 7$$

$$T(n) = 8T(n-1) - 16T(n-2)$$

a

Usando la ecuación característica

Dado el siguiente programa:

```
int g(int n) {  
    if (n < 2)  
        return n;  
    else  
        return 5 * g(n-1) - 6 * g(n-2);  
}
```

a

Escribir la ecuación de recurrencia

b

Determinar qué valor dará como respuesta (para  $n > 2$ )

Resolver la recurrencia:

$$T(1) = 1$$

$$T(n) = 3 T(n - 1) + 2$$

Resolver la recurrencia:

$$T(0) = 4$$

$$T(n) = 2 T(n - 1) + 3^n$$



## Teorema Maestro (simple)

$$T(n) = aT\left(\frac{n}{b}\right) + \Theta(n^k)$$

$$T(n) = \begin{cases} \Theta(n^k \log n) & \text{if } a = b^k \\ \Theta(n^k) & \text{if } a < b^k \\ \Theta(n^{\log_b a}) & \text{if } a > b^k \end{cases}$$

## Otras formulas maestras

**Ejemplo 8.12** Con  $a, b, b' \in R^+$  y  $c \in N$  con  $c > 0$  y

$$f(n) = \begin{cases} b' & \text{si } n \leq n_0 \\ af(n-c) + b & \text{si } n > n_0 \end{cases}$$

$$\begin{aligned} a < 1 &\Rightarrow f \in \Theta(1), \\ a = 1 &\Rightarrow f \in \Theta(n), \\ a > 1 &\Rightarrow f \in \Theta\left(a^{\frac{n}{c}}\right). \end{aligned}$$

**Ejemplo 8.13** Si  $a, b, b' \in R^+$ ,  $c \in N$  con  $c > 0$  y  $d \in R$  y:

$$f(n) = \begin{cases} b' & \text{si } 0 \leq n \leq n_0 \\ af(n-c) + bn + d & \text{si } n > n_0 \end{cases}$$

$$\begin{aligned} a < 1 &\Rightarrow f \in \Theta(n), \\ a = 1 &\Rightarrow f \in \Theta(n^2), \\ a > 1 &\Rightarrow f \in \Theta\left(a^{\frac{n}{c}}\right). \end{aligned}$$

a

$$T(n) = 4 T\left(\frac{n}{2}\right) + n$$

b

$$T(n) = T\left(\frac{n}{3}\right) + 1$$

c

$$T(n) = 3T\left(\frac{n}{2}\right) + n$$

d

$$T(n) = 2T\left(\frac{n}{2}\right) + n$$

e

$$T(n) = 2T\left(\frac{n}{2}\right) + n^2$$

```
def algoritmo(x: array, n: int)
  var i, j
  var a, b, c: array
  if n > 1:
    for i =1 to n/2
      for j =1 to n/2
        a[i,j] = x[i, j]
        b[i,j] = x[i, j+n/2]
        c[i,j] = x[i+n/2, j]

    algoritmo(a, n/2)
    algoritmo(b, n/2)
    algoritmo(c, n/2)
```

a

Escribir la ecuación de recurrencia

b

Calcular el orden del algoritmo

