

Considérese el siguiente algoritmo, que determina si una matriz cuadrada A de tamaño nxn es simétrica:

```
es_traspuesta = TRUE;
c = 1;
mientras (c ≤ n) AND es_traspuesta hacer
    f = n;
    mientras (f > c) AND es_traspuesta hacer
        si A(f,c) ≠ A(c,f) entonces
            es_traspuesta = FALSE;
        finsi
        f = f - 1;
    finmientras
    c = c + 1;
finmientras
return es_traspuesta
```

Se pide hallar O,  $\Omega$  y  $\Theta$ .

-----

Cálculo de la traspuesta de una matriz

```
filas = numfilas (A);
para i = 1 hasta filas - 1 hacer
    para j= i + 1 hasta filas hacer
        b = A[j,i];
        A[j,i] = A[i,j];
        A[i,j] = b;
    finpara
finpara
```

Se pide hallar O,  $\Omega$  y  $\Theta$ .

-----

Búsqueda de un elemento en un vector ordenado

```
repetir
    m= (pri + ult) / 2;
    si v[m] > x entonces
        ult= m - 1
    si_no
        pri= m + 1
    finsi
hasta (pri > ult) OR v[m] = x;
si v[m] = x entonces
    devuelve m
si_no
    devuelve 0
fsi
```

Se pide hallar O,  $\Omega$  y  $\Theta$ .

-----

```

r = 0
para i = 0 hasta (x + y) hacer
    j = i;
    mientras (j ≠ 0) hacer
        j = j div 2;
        r = r + j;
    finmientras
finpara
devuelve r;

```

Se pide hallar  $O$ ,  $\Omega$  y  $\Theta$ .

---

```

permuta = CIERTO;
i = 1;
mientras (permuta) hacer
    i = i + 1;
    permuta = FALSO;
    para j = n hasta i hacer
        si (a[j] < a[j - 1]) entonces
            x = a[j];
            permuta = CIERTO;
            a[j] = a[j - 1];
            a[j - 1] = x;
        finsi
    finpara
finmientras

```

Se pide hallar  $O$ ,  $\Omega$  y  $\Theta$ .

---

Dado el siguiente algoritmo en Pascal:

```

función invierte (A: array [0..n, 0..n] of integer);
var
    i, j: integer;
begin
    for i:= 0 to n-1 do
        for j:= i+1 to n do
            A[i, j]:= 2*A[j, i]+A[i, j];
end;

```

Calcular el tiempo de ejecución  $t(n)$ . Expresar el orden de complejidad de  $t(n)$  con las notaciones  $\Theta$  y  $o$ .

---

Para cada una de las siguientes afirmaciones, decir justificadamente si son ciertas o falsas:

- $\Omega(f(n)+g(n)) = \Omega(\min(f(n), g(n)))$
  - $t(n+1) \in \Theta(t(n))$  para toda función  $t: \mathbb{N} \rightarrow \mathbb{R}^+$
  - $\Theta(2n^2 + 5n - 1) = \Theta(n^2)$
- 

Decir si las siguientes afirmaciones son ciertas o falsas, justificando la respuesta muy brevemente:

- i.  $4n^2 + 2n + \log_2 n \in O(n^2 + \log n)$
  - ii.  $(2n)! \in \Omega(n!)$
  - iii.  $\Omega(f(n) \mid n=2^k) \subset \Omega(f(n) \mid n=2^k)$
- 

Decir si las siguientes afirmaciones son ciertas o falsas, justificando la respuesta muy brevemente:

- i.  $\Omega(n) \subseteq O(n^2)$
  - ii.  $f(n+1) \in O(f(n)), \forall f: \mathbb{N} \rightarrow \mathbb{R}^+$
- 

Encontrar  $O$  y  $\Omega$  del algoritmo:

```
max = 0
para i = 1 hasta n hacer
    cont = 1
    j = i + 1
    mientras a[i] ≤ a[j] y j ≤ n hacer
        j = j + 1
        cont = cont + 1
    finmientras
    si cont > max entonces
        max = cont
    finsi
finpara
```

---