

Nombre y apellidos del alumno: .....

## Tema 01. Introducción a la Programación Concurrente y Paralela

**1** (Transparencia: 17) (Tiempo estimado: 5') ¿Cuál es el cuello de botella (CPU, memoria o E/S) de los siguientes fragmentos de código?

1.1)

```
1 xk = a / 2.0;  
2 for( i = 0; i < n; i++ ) {  
3     xk = ( xk * xk + a )/( 2.0 * xk );  
4 }
```

1.2)

```
1 for( i = 0; i < n; i++ ) {  
2     fprintf( f, "%f \n", 2.0 * v[ i ] );  
3 }
```

1.3)

```
1 for( i = 0; i < n; i++ ) {  
2     w[ i ] = 2.0 * v[ i ];  
3 }
```

1.4)

```
1 esPrimo = 1;  
2 i = 2;  
3 while( ( i <= n/2 ) && esPrimo ) {  
4     if( n % i == 0 ) {  
5         esPrimo = 0;  
6     } else {  
7         i++;  
8     }  
9 }
```

**2** (Transparencia: 26) (Tiempo estimado: 5') Para cada problema indica el tipo de programación más adecuada de entre los siguientes paradigmas: Programación Secuencial, Programación Concurrente o Programación Paralela.

2.1) Navegador con varias pestañas abiertas.



2.2) Simulaciones del choque de dos galaxias.



2.3) Procesador de texto con la corrección habilitada.



2.4) Cálculo del factorial de un número pequeño.



**3** (*Transparencia: 31*) (*Tiempo estimado: 10'*) Indica si cada uno de los siguientes problemas es paralelizable o no. Si es paralelizable, indica cómo se podría llevar a cabo dicha tarea.

3.1) Suma de los elementos de un vector.



3.2) Búsqueda binaria de un determinado elemento en un vector ordenado.



3.3) Búsqueda de un determinado elemento en un vector no ordenado.



3.4) Aproximación de la raíz de un número  $a$  mediante la sucesión siguiente:

$$x_{k+1} = \frac{x_k^2 + a}{2x_k}, \quad x_0 = a/2$$



**Nota:** Esta entrega forma parte de la evaluación de la asignatura. Debe ser guardado por el estudiante junto con el resto de entregas en una carpeta. El profesor podrá pedir al estudiante que le entregue dicha carpeta en cualquier momento.