

Nombre: _____

D.N.I: _____

| | | | | | | | | | | | | |
|--------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | Suma |
| | (4) | (4) | (4) | (4) | (3) | (5) | (6) | (4) | (6) | (4) | (6) | (50) |
| Puntos | | | | | | | | | | | | |

Pregunta 1: [4 Puntos]

Describe brevemente el término “programación concurrente” destacando las diferencias con la “programación secuencial”. Enumera unos indicadores que sugieran la aplicación de un método concurrente con casos prácticos.

Pregunta 2: [4 Puntos]

¿Cuáles son los principales problemas con los cuáles un programador se tiene que enfrentar cuando usa la sincronización entre procesos?

Pregunta 3: [4 Puntos]

Define los términos “seguridad” y “vivacidad” en el contexto de programación concurrente. Añade dos ejemplos de tal propiedades para cada término.

Pregunta 4: [4 Puntos]

¿Qué se entiende por “instrucción atómica” y “región crítica en lenguajes de programación de diferente nivel (incluyendo Java)?

Pregunta 5: [3 Puntos]

¿Cómo se define formalmente que un programa concurrente es correcto?

Pregunta 6: [5 Puntos]

¿Qué se entiende entre justicia entre procesos? ¿Cuáles son los algoritmos que se usa en la práctica para obtener algún tipo de justicia en el planificador?

Pregunta 7: [4 Puntos]

Describe el concepto de monitor y su uso en la programación concurrente.

Pregunta 8: [6 Puntos]

Explica brevemente los métodos de la clase `Thread` disponibles en Java enfocando solamente en su uso para la programación concurrente.

Pregunta 9: [6 Puntos]

En un sistema distribuido donde los nodos de procesamiento están conectados por canales de comunicación se pueden provocar diferentes tipos de fallos enviando mensajes de un nodo al otro. Enumera dichos tipos de fallos y razona sobre técnicas disponibles para superar la deficiencia del canal.

Pregunta 10: [6 Puntos]

Describe brevemente las partes principales por desarrollar para implementar un editor concurrente (sin servidor).

Pregunta 11: [4 Puntos]

Discute una posible solución para implementar una estructura de datos concurrente `array` donde la probabilidad de una escritura de un elemento sea mucho más pequeña que la probabilidad de su lectura, sobre todo considerando la eficiencia si muchos hilos acceden al mismo `array`.