

Nombre: _____

D.N.I: _____

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

Pregunta 1: [4 Puntos]

Describe brevemente cuándo se usa la “programación concurrente” en vez de la “programación secuencial”. Enumera unos indicadores que sugieran la aplicación de un método concurrente con casos prácticos.

Pregunta 2: [4 Puntos]

Enumera y critica unas desventajas de la programación concurrente.

Pregunta 3: [4 Puntos]

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

Pregunta 4: [4 Puntos]

¿Cuál es la diferencia entre las propiedades de “seguridad” y de “vivacidad” en el contexto de programación concurrente. Añade dos ejemplos de tal propiedades para cada término.

Pregunta 5: [4 Puntos]

¿Qué es un semáforo y cuándo se usa? Distingue entre semáforos binarios y semáforos generales.

Pregunta 6: [4 Puntos]

¿Cuáles son posibles métodos de representación de programas concurrentes que se usan en la práctica (claro, menos el propio código)?

Pregunta 7: [4 Puntos]

Explica dos algoritmos que se usa en la práctica para obtener algún tipo de justicia en el planificador de tareas concurrentes?

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 `matriz` donde la probabilidad de acceso a elementos en la misma fila es mucho mas pequeña que la probabilidad del acceso a la misma columna, sobre todo considerando la eficiencia si muchos hilos acceden a la misma `matriz`.