

Apellidos, Nombre: _____

D.N.I., Firma: _____

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

Pregunta 1: [4 Puntos]

Destaca brevemente las ventajas de la programación concurrente tanto en entornos con un procesador como en entornos distribuidos.

Pregunta 2: [4 Puntos]

Se distingue en la programación concurrente entre propiedades de “seguridad” y “vivacidad”. Explica brevemente ambos términos y añade ejemplos.

Pregunta 3: [6 Puntos]

Describe brevemente como el hardware del procesador con una operación atómica de “compare-and-swap” puede ayudar a implementar los protocolos de entrada y salida para el acceso con exclusión mutua a una sección crítica.

Pregunta 4: [4 Puntos]

¿Cuáles son las condiciones que se tienen que cumplir para que se produzca un bloqueo entre procesos?

Pregunta 5: [6 Puntos]

Describe brevemente los patrones de diseño *proactor* y *reactor*.

Pregunta 6: [8 Puntos]

Razona sobre las propiedades (política de justicia) del siguiente sistema que controla el acceso a un recurso compartido. Cada proceso que quiere acceso al recurso se añade a una cola; el primer proceso en la cola obtiene acceso al recurso (con exclusión mutua) cuando el recurso esté disponible; mientras los procesos están esperando en la cola, cualquier dos de ellos pueden cambiar su orden en la cola, siempre cuando ambos procesos involucrados están mutuamente de acuerdo. (Se puede asumir que un proceso usa el recurso solamente durante un tiempo finito.)

Pregunta 7: [4 Puntos]

Describe el mecanismo de sincronización disponible en Java basado en el uso de la palabra reservada `synchronized`.

Pregunta 8: [4 Puntos]

Describe el uso de los métodos `wait()`, `notify()` y `notifyall()` disponibles en Java. ¿Cómo se implementaría un “region crítica condicional” con dicho mecanismo?

Pregunta 9: [10 Puntos]

Describe brevemente, pero de forma precisa, como se puede implementar las operaciones *insertar*, *borrar*, y *iterar* sobre la lista concurrente de tal manera que no se produzca ningún bloqueo infinito.