CD Examen Teoría **1 de septiembre de 2007**

Dr. Arno Formella

Apellidos, Nombre: _____

D.N.I: _____

Firma: _____

Prácticas presentadas: _____

_____

1	2	3	4	5	6	7	8	9	10	11	Suma
(6)	(6)	(4)	(8)	(4)	(6)	(6)	(5)	(5)	(5)	(5)	(60)

Pregunta 1: [6 Puntos]

Depurar un programa concurrente es difícil. ¿Por qué?

Pregunta 2: [6 Puntos]

¿Qué se entiende con "espera finita" de un proceso cuando quiere acceder a un recurso? ¿Cuáles son las propiedades de dicha espera finita que hay que tener en cuenta cuando se analiza el comportamiento en una aplicación concurrente concreta?

Pregunta 3: [4 Puntos]

Describe brevemente el patrón de diseño *ficha de terminación aíncrona* (o *magic cookie*).

Pregunta 4: [8 Puntos]

¿Cuáles son las condiciones que se tienen que cumplir para que se produzca un bloqueo entre procesos? Describe los tres métodos presentados en clase para solventar el problema del bloqueo. Razona brevemente sobre sus eficiencias en entornos distribuidos.

Pregunta 5: [4 Puntos]

¿Cómo se define formalmente que un programa concurrente es correcto? ¿Por qué se distingue entre corrección parcial y total?

Pregunta 6: [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 7: [6 Puntos]

Implementa en pseudo-código el sistema productor–consumidor con cola de comunicación finita donde varios productores y consumidores usan la misma cola.

Pregunta 8: [5 Puntos]

Reflexiona brevemente sobre las limitaciones para la programación concurrente en el uso de `synchronized` en Java.

Pregunta 9: [5 Puntos]

¿Cómo se puede conseguir que solo un hilo construye un objeto en Java (es decir, un segundo hilo que lo intente ya encuentra el objeto construido y lo nota)? ¿Cuándo hace falta una construcción de tal tipo?

Preguntas para las prácticas:

Pregunta 10: [5 Puntos]

En el ejercicio del PingPong vimos que el uso de `notify()` y `notifyAll()` resultaba en un número de posibles jugadas entre los jugadores en cierto intervalo de tiempo que dependía del número de jugadores participando en el juego.

- ¿Por qué eso era el caso?
- ¿Cómo conseguimos finalmente un número constante de jugadas, en cierto intervalo de tiempo, independientemente del número de jugadores participantes?

Pregunta 11: [5 Puntos]

En las prácticas hemos utilizado la implementación de una lista concurrente que disponía de las operaciones *insertar*, *borrar*, e *iterar*.

- Describe brevemente cuales fueran las técnicas usadas para que varios hilos pueden manipular la lista concurrentemente (por lo menos en aquellas situaciones en las cuales las modificaciones tienen lugar en posiciones suficientemente separadas en la lista), es decir, hacer uso de las tres operaciones, sin que exista la posibilidad de un bloqueo mutuo.
- Usamos dicha lista para implementar una tabla de hash simple. Describe brevemente cómo implementaste las operaciones principales de tal tablas que son *insertar*, *buscar*, y *borrar*.