

Apellidos, Nombre: \_\_\_\_\_

D.N.I., Firma: \_\_\_\_\_

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

**Pregunta 1:** [4 Puntos]

Explica brevemente las ventajas de la programación concurrente que se ofrecen 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 razona cómo un programador debe tener en cuenta dichas propiedades en el desarrollo de un programa concurrente.

**Pregunta 3:** [6 Puntos]

Describe brevemente como el hardware del procesador con una operación atómica de “test-and-set” 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 los métodos que se pueden aplicar para que no se produzca un bloqueo entre procesos?

**Pregunta 5:** [8 Puntos]

Describe brevemente los patrones de diseño *guardián* y *monitor*.

**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 y se elimina de la cola; 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.) Observa que dicho sistema puede ser útil para implementar un mecanismo de priorización entre los procesos; comenta.

**Pregunta 7:** [4 Puntos]

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

**Pregunta 8:** [4 Puntos]

Describe la semántica y el uso de `volatile` disponible en Java.

**Pregunta 9:** [8 Puntos]

Razona brevemente sobre una posible aplicación del ejemplo pingPONG de las prácticas, donde queremos que los hilos involucrados (el “arbitro”, el “ping”, y el “pong”, y el “pang”, etc.) produzcan la siguiente salida:

```
arbitro: empezamos
ping:    ping
pong:    PONG
pang:    pang
pung:    pung
...      (y así alternando durante 10 segundos)
arbitro: vamos a terminar
...      (como mucho una salida para cada jugador)
ping:    he terminado
pong:    he terminado
...
Arbitro: juego terminando
```

Describe cómo lograr un paso directo de un hilo a otro sin despertar innecesariamente todos los hilos esperando para que se cumpla la condición adecuada.

**Pregunta 10:** [10 Puntos]

Describe brevemente, pero de forma precisa, cómo se puede implementar las operaciones *insertar*, *borrar*, e *iterar* sobre una lista concurrente de tal manera que no se produzca ningún bloqueo infinito y que varios hilos puedan acceder a la lista con operaciones concurrentes no-conflictivas.