

APELLIDOS:		NOMBRE:	
DNI:		FIRMA:	

Este bloque tiene una puntuación máxima de **10 puntos** (que aportará 2.5 puntos a la nota global).

Indique, para cada una de las siguientes 50 afirmaciones, si éstas son verdaderas (V) o falsas (F). **Cada respuesta vale: correcta= 0.2, errónea= -0.2, vacía=0.** Al final de cada pregunta dispone de un espacio reservado para justificar mejor su respuesta, en caso de considerarlo necesario.

1. Un sistema distribuido:

V	Es una clase de sistema concurrente; aunque no todos los sistemas concurrentes son distribuidos.
V	Ofrece la imagen de un sistema coherente y único.
F	Proporcionará diferentes tipos de transparencia. Entre ellos: transparencia en el rendimiento, en la escalabilidad, en su disponibilidad, en la seguridad...
F	Es un sistema de tiempo real que necesitará un análisis de planificabilidad.

JUSTIFICACIÓN:

2. La comunicación basada en mensajes:

F	Es un tipo particular de memoria compartida y requiere acceso en exclusión mutua.
F	Será sincrónica cuando el canal pueda mantener los mensajes durante un intervalo indefinido.
V	Se utiliza en su variante asincrónica para implantar el servicio de correo electrónico.
V	Se utiliza en su variante sincrónica no persistente para implantar las llamadas a procedimiento remoto.

JUSTIFICACIÓN:

3. El algoritmo de Cristian:

F	Proporciona la base necesaria para implantar cualquier algoritmo descentralizado.
V	Permite sincronizar el reloj local de un nodo cliente con el mantenido por un nodo servidor.
F	Es uno de los algoritmos de elección de líder más eficientes.
F	Permite identificar los mensajes en tránsito por cada uno de los canales de comunicación de un sistema distribuido.

JUSTIFICACIÓN:

4. Sobre los relojes lógicos de Lamport:

F	Son necesarios para deshacer los empates en el algoritmo de Chandy y Lamport.
V	Ordenan de manera parcial los eventos que hayan ocurrido en un sistema distribuido.
F	Permiten determinar en todos los casos si dos eventos de una traza han sido concurrentes o no.
V	Si $C(a)=C(b)$, entonces $a b$.

JUSTIFICACIÓN:

5. Sobre los servicios de nombres en un sistema distribuido:

F	Se necesitan para asegurar la exclusión mutua.
V	Suelen utilizar una estructura jerárquica para facilitar su escalabilidad.
V	Pueden retornar identificadores para facilitar la gestión de entidades móviles.
V	LDAP es un ejemplo de servicio de directorio basado en atributos.

JUSTIFICACIÓN:

6. Un sistema "peer-to-peer":

F	Es un ejemplo de arquitectura software en niveles.
V	Emplea una arquitectura de sistema basada en distribución horizontal.
F	Es un ejemplo de arquitectura de sistema centralizada.
V	Es un ejemplo de sistema distribuido con buenas escalabilidades de tamaño y distancia.

JUSTIFICACIÓN:

7. Sobre tiempo lógico:

V	Si "a ---> b", entonces $C(a) < C(b)$.
V	Si "a ---> b", entonces la ejecución del evento "a" siempre sucederá antes que la ejecución del evento "b".
F	Si "a b", entonces la ejecución de los eventos "a" y "b" siempre sucederá en el mismo instante de tiempo.
V	Sean $VT(a)=[3,4,4]$ y $VT(b)=[5,4,8]$ los relojes vectoriales de dos eventos "a" y "b" en un sistema formado por tres procesos P1, P2 y P3. Podemos afirmar que "b" no ha sido ejecutado por el proceso P2.

JUSTIFICACIÓN:

8. Sobre el mecanismo de invocación a objeto remoto (ROI):

V	Es el utilizado en Java RMI.
V	Proporciona transparencia de ubicación.
V	Admite paso de parámetros por referencia en sus invocaciones.
F	Proporciona transparencia de fallos.

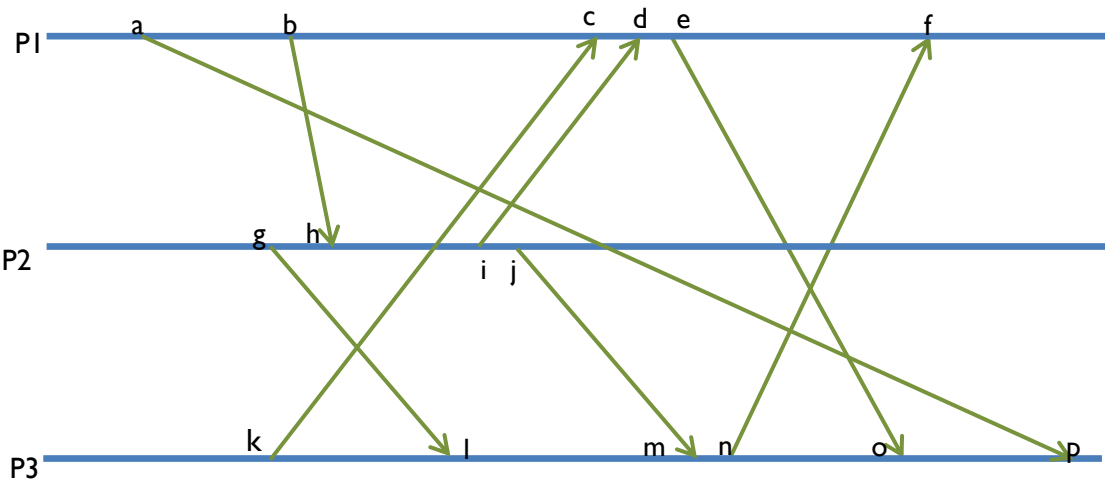
JUSTIFICACIÓN:

9. Sobre los algoritmos descritos en el tema 9 ("Sincronización en sistemas distribuidos"):

F	Se describieron dos algoritmos basados en anillos: uno de exclusión mutua y otro de recolección del estado global.
V	El algoritmo de Chandy y Lamport recoge el estado global del sistema.
F	El algoritmo Bully resuelve el problema de la exclusión mutua en un sistema distribuido.
V	Algunos algoritmos de exclusión mutua pueden utilizar relojes lógicos e identificadores para deshacer los empates que surjan.

JUSTIFICACIÓN:

10. Dado el siguiente conjunto de eventos en un sistema distribuido, asumiendo que no hay otros eventos previos:



V	El reloj vectorial de “p” es $VT(p)=[5,4,6]$ y el de “f” es $VT(f)=[6,4,4]$.
V	Los eventos “c” y “m” son concurrentes.
V	El reloj de Lamport de “d” es $C(d)=5$ y el de “m” es $C(m)=6$.
V	A partir de la figura podemos afirmar que “c \rightarrow o” (“c ocurre antes que o”), pues existe un camino dirigido que va desde “c” hasta “o”.
V	Si el reloj vectorial de un evento “x” fuera $VT(x)=[5,4,1]$ y el de “f” fuese $VT(f)=[6,4,4]$, entonces podríamos afirmar que “x \rightarrow f”.

JUSTIFICACIÓN:

11. Sobre la gestión de recursos:

V	Cuando una entidad pasa a no tener ninguna referencia en el sistema, se convierte en un residuo y debería ser localizada y eliminada.
V	El nombre de un punto de entrada es una dirección.
F	Las direcciones no pueden reutilizarse.
F	Un subservicio de localización guarda las correspondencias entre nombres e identificadores.

JUSTIFICACIÓN:

12. Los sistemas “cloud”:

V	Soportan el paradigma de “pago por uso”.
V	Son un ejemplo de arquitectura de sistema descentralizada.
F	Son un ejemplo de arquitectura software orientada a objetos.
F	Son un ejemplo de arquitectura software híbrida.
V	Permiten instalar y/o utilizar aplicaciones distribuidas sin necesidad de realizar una gran inversión en <i>hardware</i> .

JUSTIFICACIÓN:

APELLIDOS:		NOMBRE:	
DNI:		FIRMA:	

Este bloque tiene una puntuación máxima de **10 puntos** (que aportará 0.5 puntos a la nota global).

Indique, para cada una de las siguientes 10 afirmaciones, si éstas son verdaderas (V) o falsas (F). **Cada respuesta vale: correcta= 1, errónea= -1, vacía=0.** Al final de cada pregunta dispone de un espacio reservado para justificar mejor su respuesta, en caso de considerarlo necesario.

1. Sobre la práctica de los filósofos comensales:

F	Se ha implementado una solución al problema, basada en la asimetría entre los filósofos que resuelve el problema rompiendo la condición de exclusión mutua.
V	Se ha implementado una solución al problema, basada en la asimetría entre los filósofos que resuelve el problema rompiendo la condición de espera circular.
F	La solución basada en la clase PhiloBothOrNone resuelve el problema de interbloqueos rompiendo la condición de no expropiación.
V	La solución basada en la clase PhiloBothOrNone resuelve el problema de interbloqueos rompiendo la condición de retención y espera.
V	Se ha desarrollado una solución al problema, basada en la limitación del número de comensales que resuelve el problema de interbloqueos rompiendo la condición de espera circular.

JUSTIFICACIÓN:

2. Sobre la práctica 4:

V	Los procesos ChatServer y <code>rmiregistry</code> pueden arrancar en el mismo ordenador.
F	El primer paso de todo cliente es conectarse al servidor de chat, tras lo cual contacta con el servidor de nombres.
V	Podemos lanzar varios clientes en una misma máquina o en máquinas diferentes.
V	Los ChatClient deben obtener la lista de canales invocando algún método del ChatServer.
V	El proceso ChatClient no se registra en <code>rmiregistry</code> .

JUSTIFICACIÓN: