2º Parcial de Teoría (TSR)

Este examen consta de 40 cuestiones. En cada caso sólo una de las respuestas es correcta. Para indicar la respuesta basta con rellenar la casilla correspondiente en la hoja de respuestas adjunta. Todas las cuestiones tienen el mismo valor. Si son correctas, aportan 0,25 puntos a la nota obtenida. Si son incorrectas descontarán 1/5 del valor correcto, es decir, -0,05 puntos. Conviene pensar cuidadosamente las respuestas.

Duración prevista para esta parte del examen: 2 horas.

1. El modelo de fallos bizantino...

A	...no debe asumirse para implantar un servicio distribuido real ya que los ordenadores no pueden tener ese comportamiento.
B	...asume que los procesos pueden tener un comportamiento arbitrario.
C	...no tiene sentido actualmente. Como sugiere su nombre, modela el comportamiento de ordenadores obsoletos, en lugar de ordenadores recientes.
D	...asume que los procesos pueden fallar parando y que esos fallos pueden ser detectados por otros procesos.
E	Todas las anteriores.
F	Ninguna de las anteriores.

2. Cuando diseñamos un algoritmo asumiendo el modelo de fallos de parada...

A	...el algoritmo resultante será sencillo ya que se asume que los procesos se comportan según su especificación hasta que fallan.
B	...habrá dificultades para implantar ese algoritmo pues los sistemas operativos y el middleware no garantizan un comportamiento perfecto de los procesos.
C	...se asume que los procesos fallan parando y que esos fallos pueden ser detectados por los procesos correctos.
D	...se asume que los canales de comunicación funcionan correctamente.
E	Todas las anteriores.
F	Ninguna de las anteriores.

3. Hay un fallo de partición de la red cuando...

A	...un proceso para.
B	...un proceso muestra un comportamiento arbitrario.
C	...se fragmenta y distribuye una base de datos entre múltiples ordenadores, pero de una forma incorrecta (p.ej., borrando algunas filas en varias tablas).
D	...el teorema CAP no ha sido respetado.
E	Todas las anteriores.
F	Ninguna de las anteriores.

4. En el modelo de "partición primaria":

A	Los subgrupos minoritarios (aquellos con menos de la mitad de los nodos) deben parar.
B	Todos los canales de comunicación funcionan correctamente.
C	Aplicamos el teorema CAP sacrificando la tolerancia al particionado.
D	Todos los subgrupos de nodos pueden continuar.
E	Todas las anteriores.
F	Ninguna de las anteriores.

5. La seguridad (*Safety*) es...

A	...un atributo cualitativo de la robustez.
B	...la probabilidad S(t) de que un sistema distribuido se recupere en el instante t si había fallado en el instante t'=0.
C	...un modelo de fallos.
D	...uno de los aspectos considerados en el teorema CAP.
E	Todas las anteriores.
F	Ninguna de las anteriores.

6. La replicación mejora el rendimiento de un servicio cuando:

A	...la mayor parte de las operaciones son de solo lectura.

B	...todas las operaciones implican escrituras.
C	...las réplicas están continuamente recuperándose de fallos en sus procesos.
D	...se usa un modelo de replicación pasiva, se pierde la conectividad de la red y las modificaciones del estado no pueden propagarse a las réplicas secundarias.
E	Todas las anteriores.
F	Ninguna de las anteriores.

7. En el modelo de replicación pasiva:

A	Múltiples réplicas reciben y procesan directamente las peticiones de los clientes.
B	Todas las réplicas interpretan un mismo papel.
C	Para implantarlo no se puede asumir el modelo de fallos arbitrarios.
D	Para implantarlo no se puede asumir el modelo de fallos de parada.
E	Todas las anteriores.
F	Ninguna de las anteriores.

8. Sobre los modelos de consistencia centrados en datos:

A	Causal es más estricto que caché.
B	Procesador es más estricto que causal.
C	Caché es más estricto que FIFO.
D	Secuencial es más estricto que caché.
E	Todas las anteriores.
F	Ninguna de las anteriores.

9. El acoplamiento mide:

A	El grado de dependencia entre los módulos de una aplicación.
B	La fiabilidad de una aplicación.
C	La continuidad de servicio.
D	Si cada una de las dimensiones consideradas en el teorema CAP se está garantizando.
E	Todas las anteriores.
F	Ninguna de las anteriores.

10. No interesa una cohesión débil porque:

A	Siempre genera defectos, errores y fallos.
B	Implica una pérdida de disponibilidad.
C	La funcionalidad de cada operación no queda clara. Esto evita que los módulos puedan reutilizarse en otras aplicaciones.
D	Asegura consistencia fuerte entre réplicas.
E	Todas las anteriores.
F	Ninguna de las anteriores.

11. En un sistema distribuido con acoplamiento bajo:

A	Los mensajes de petición serán pequeños, generalmente.
B	Se minimiza la interacción entre componentes.
C	Hay un alto grado de localidad en el acceso a datos.
D	Cada operación solo necesita unos pocos argumentos.
E	Todas las anteriores.
F	Ninguna de las anteriores.

12. Los almacenes NoSQL...

A	...aseguran la consistencia de datos utilizando transacciones ACID.
B	...suelen ser más escalables que los sistemas gestores de bases de datos relacionales.
C	...no proporcionan persistencia de datos.
D	...utilizan un lenguaje de interrogación basado en el operador JOIN.
E	Todas las anteriores.
F	Ninguna de las anteriores.

13. Los almacenes clave-valor:

A	Son ejemplos de sistemas gestores de bases de datos relacionales.
B	Usan un esquema basado en objetos con un número variable de atributos.
C	Dos ejemplos son MongoDB y SimpleDB.
D	Dos ejemplos son Cassandra y PNUTs.
E	Todas las anteriores.
F	Ninguna de las anteriores.

14. Los almacenes de registros extensibles:

A	Usan esquemas basados en tablas con un número variable de columnas.
B	Usan particionado vertical y particionado horizontal de las tablas para mejorar su escalabilidad.
C	Un ejemplo es Bigtable.
D	Un ejemplo es Cassandra.
E	Todas las anteriores.
F	Ninguna de las anteriores.

15. Sobre el teorema CAP:

A	Relaciona consistencia, disponibilidad y tolerancia al particionado.
B	Su resultado solo tiene sentido en sistemas distribuidos sincrónicos.
C	Establece que la consistencia fuerte y la alta disponibilidad no pueden proporcionarse simultáneamente.
D	Relaciona consistencia de datos, atomicidad y persistencia.
E	Todas las anteriores.
F	Ninguna de las anteriores.

16. Sobre las consecuencias del teorema CAP:

A	Para implantar un servicio altamente disponible y tolerante a particiones, necesitamos utilizar consistencia eventual (o final).
B	Para implantar un servicio con consistencia fuerte (secuencial) y altamente disponible, no podremos permitir particiones de la red.
C	La consistencia eventual es prácticamente obligatoria para implantar servicios escalables, pues estos deben ser altamente disponibles y tolerar particiones.
D	El modelo de "partición primaria" se asume para sacrificar la disponibilidad, gestionar particiones y asegurar consistencia fuerte.
E	Todas las anteriores.
F	Ninguna de las anteriores.

17. Las tres dimensiones de la escalabilidad son:

A	Consistencia, disponibilidad y tolerancia a particiones.
B	Fiabilidad y los dos tipos de seguridad (*security* y *safety*).
C	Hardware, firmware y software.
D	Interfaz de usuario, lógica de negocio y datos persistentes.
E	Todas las anteriores.
F	Ninguna de las anteriores.

18. La escalabilidad vertical consiste en:

A	Adaptar la capacidad de cómputo a la carga actual.
B	Asegurar continuidad de servicio durante las actualizaciones de software.
C	Incrementar el número de nodos en los que un servicio está ejecutándose.
D	Incrementar la capacidad de cómputo de un nodo determinado.
E	Todas las anteriores.
F	Ninguna de las anteriores.

19. La escalabilidad horizontal consiste en:

A	Adaptar la capacidad de cómputo a la carga actual.
B	Asegurar continuidad de servicio durante las actualizaciones de software.
C	Incrementar el número de nodos en los que un servicio está ejecutándose.
D	Incrementar la capacidad de cómputo de un nodo determinado.
E	Todas las anteriores.
F	Ninguna de las anteriores.

20. Los cuatro mecanismos complementarios para alcanzar escalabilidad de tamaño son:

A	Fiabilidad, disponibilidad, mantenibilidad y seguridad.
B	Reparto de tareas, reparto de datos, replicación y uso de cachés.
C	Consistencia estricta, particionado horizontal, replicación activa y tolerancia a particiones.
D	Elasticidad, computación en la nube, computación grid y computación P2P.
E	Todas las anteriores.
F	Ninguna de las anteriores.

21. Propiedades de los algoritmos descentralizados:

A	Cuando un proceso falla, el algoritmo se bloquea.
B	Los procesos toman decisiones utilizando información local.
C	Los procesos asumen que existe un reloj global.
D	Uno de los procesos tiene información completa sobre el estado del sistema.
E	Todas las anteriores.
F	Ninguna de las anteriores.

22. El particionado de tablas (*sharding*) mejora la escalabilidad porque...

A	Es un mecanismo de reparto de datos.
B	Proporciona equilibrado de carga.
C	Incrementa el grado de concurrencia, así el servicio resultante puede procesar un mayor número de peticiones simultáneamente.
D	Con un diseño apropiado, no necesita ningún paso de sincronización.
E	Todas las anteriores.
F	Ninguna de las anteriores.

23. Un servicio es elástico cuando…

A	Es fiable y altamente disponible.
B	Es robusto y utiliza un modelo de consistencia rápido.
C	Es escalable y se adapta dinámica y autónomamente.
D	Es seguro y tolerante a defectos (*fault-tolerant*).
E	Todas las anteriores.
F	Ninguna de las anteriores.

24. Para implantar un servicio elástico, necesitamos:

A	Un sistema de monitorización de la carga actual.
B	Un sistema reactivo que automatice la reconfiguración del servicio, tomando decisiones de escalado.
C	Un sistema reactivo que tenga en cuenta el SLA.
D	Un sistema de monitorización del rendimiento actual.
E	Todas las anteriores.
F	Ninguna de las anteriores.

25. Los objetivos principales de la seguridad (*security*) son:

A	Asegurar la persistencia y consistencia de los datos.
B	Confidencialidad, integridad, disponibilidad y contabilidad.
C	Seguridad (*safety*), fiabilidad, disponibilidad y mantenibilidad.
D	Transparencia, continuidad de servicio, rendimiento y eficiencia.
E	Todas las anteriores.
F	Ninguna de las anteriores.

26. El objetivo de una política de seguridad es:

A	Asegurar la corrección de un sistema de seguridad.
B	Implantar un sistema de seguridad.
C	Especificar un sistema de seguridad.
D	Evaluar la robustez de un sistema.
E	Todas las anteriores.
F	Ninguna de las anteriores.

27. Sobre los mecanismos de seguridad:

A	Son técnicas y herramientas utilizadas para implantar la seguridad.
B	Hay tres clases principales: físicos, relacionados con autenticación y relacionados con autorización (es decir, control de acceso).
C	Un ejemplo es el uso de contraseñas.
D	Un ejemplo es la palabra de protección en los sistemas de ficheros UNIX.
E	Todas las anteriores.
F	Ninguna de las anteriores.

28. Una amenaza es:

A	Una debilidad en dispositivos, protocolos, programas o políticas de un sistema.
B	La probabilidad A(t) de que un sistema realice sus funciones en el instante t si ha estado funcionando correctamente desde el instante t'=0.
C	Un modelo que especifica qué divergencias están permitidas en las réplicas de un determinado dato.
D	Un conjunto de reglas que especifica qué acciones están autorizadas para los agentes de un sistema determinado.
E	Todas las anteriores.
F	Ninguna de las anteriores.

29. Ejemplos de vulnerabilidades en las políticas de seguridad:

A	Denegación de servicio.
B	*Man in the middle*.
C	*SYN floods*.
D	Falta de planes para recuperación de desastres.
E	Todas las anteriores.
F	Ninguna de las anteriores.

30. Ejemplos de vulnerabilidades de configuración:

A	Amenazas desestructuradas.
B	Permitir contraseñas débiles.
C	Rastreadores de paquetes.
D	*Phishing*.
E	Todas las anteriores.
F	Ninguna de las anteriores.

31. Sobre los mecanismos en los protocolos criptográficos:

A	Un MAC asegura no repudio.
B	Un certificado es un mecanismo que proporciona contabilización.
C	Las funciones unidireccionales (cripto-hashing) se utilizan en los MAC y en los certificados.
D	El cifrado simétrico necesita dos claves distintas y complementarias, una para cifrar y otra para descifrar.
E	Todas las anteriores.
F	Ninguna de las anteriores.

32. Protocolos criptográficos. Distribución de claves:

A	Con cifrado simétrico, solo necesitamos distribuir la clave privada.
B	Con cifrado asimétrico, solo necesitamos distribuir la clave pública.
C	En el cifrado simétrico, la fuga de información no es problemática.
D	En el cifrado asimétrico, necesitamos canales secretos para distribuir claves.
E	Todas las anteriores.
F	Ninguna de las anteriores.

33. En el patrón básico (y sincrónico) de petición/respuesta:

A	Las peticiones de los clientes pueden entregarse en un orden “no FIFO” al servidor.
B	El cliente puede enviar otra petición antes de recibir la respuesta para la actual.
C	Si el servidor cae antes de responder una petición de un cliente, ese cliente quedará bloqueado.
D	Este patrón se implanta utilizando sockets PUSH y PULL en ZeroMQ.
E	Todas las anteriores.
F	Ninguna de las anteriores.

34. Para desplegar el patrón básico petición/respuesta, considerando sockets ZeroMQ:

A	Los clientes realizan el bind() y los servidores el connect().
B	Los servidores realizan el bind() y los clientes el connect().
C	Tanto los clientes como los servidores usan un único socket. Ambos realizan un bind(), pero también un connect() sobre otros sockets.
D	No se necesita ningún socket para implantar este patrón arquitectónico.
E	Todas las anteriores.
F	Ninguna de las anteriores.

35. El patrón arquitectónico PUSH-PULL básico es un patrón de comunicaciones…

A	Bidireccional.
B	Sincrónico.
C	Para multienvío (*multicast*).
D	Asincrónico.
E	Todas las anteriores.
F	Ninguna de las anteriores.

36. En el patrón arquitectónico cliente/servidor avanzado con múltiples clientes y múltiples servidores interconectados mediante una cola intermedia:

A	Cada servidor es un único punto de fallo para el servicio.
B	La cola intermedia es un elemento estable. En su configuración más simple, sus sockets realizarán el bind().
C	La cola intermedia utiliza sockets PUSH-PULL.
D	Los clientes utilizan sockets SUB.
E	Todas las anteriores.
F	Ninguna de las anteriores.

37. Los *heart-beats* se utilizan en algunas arquitecturas cliente-servidor para:

A	Detectar fallos en los clientes.
B	Monitorizar la carga actual.
C	Mejorar la escalabilidad de los servidores.
D	Implantar protocolos criptográficos.
E	Todas las anteriores.
F	Ninguna de las anteriores.

38. Los reintentos se utilizan en algunas arquitecturas cliente-servidor avanzadas para:

A	Detectar fallos en los servidores.
B	Implantar un mecanismo de equilibrado de carga.
C	Mejorar la escalabilidad de los servidores.
D	Detectar fallos en los clientes, combinándolos con *timeouts*.
E	Todas las anteriores.
F	Ninguna de las anteriores.

39. Para implantar un mecanismo de recuperación en caso de fallo de una petición no idempotente, necesitamos:

A	Identificar claramente cada mensaje de petición, con un par <id-emisor, id-petición>.
B	Antes de servir cada petición, el servidor debe buscarla en su "almacén de respuestas".
C	Si encontramos una petición en el "almacén de respuestas", la respuesta es tomada de allí y enviada al cliente.
D	Tras servir cada petición, el servidor copia su mensaje de respuesta en el "almacén de respuestas" local.
E	Todas las anteriores.
F	Ninguna de las anteriores.

40. En los patrones arquitectónicos cliente/servidor avanzados del tema 9, los servidores se replican de la siguiente manera:

A	Utilizando el modelo de replicación activo.
B	Exigiendo que todas sus operaciones sean idempotentes.
C	Utilizando el modelo de replicación pasivo.
D	Tanto el proceso primario como los secundarios utilizan un módulo de *"heart-beats"* para detectar fallos en los clientes.
E	Todas las anteriores.
F	Ninguna de las anteriores.