

Este examen consta de 15 preguntas con un total de 15 puntos. Tres preguntas incorrectas restan un punto. Sólo una opción es correcta a menos que se indique algo distinto. No está permitido el uso de calculadora. La duración máxima de este examen será de 30 minutos.

Apellidos: _____ SOLUCIÓN _____ Nombre: _____ Grupo: _____

- 1** [1p] ¿Cuál es el principal motivo por el que no se dispone de un reloj global en los sistemas distribuidos?
- ☐ a) La necesidad de mantener un buen nivel de concurrencia entre procesos.
 - ☐ b) No existe tecnología para fabricar relojes con la precisión necesaria.
 - ☐ c) Las limitaciones de los lenguajes de programación.
 - ☒ d) La latencia impredecible de las comunicaciones.
- 2** [1p] La frase «La red es homogénea» es una falacia porque...
- ☐ a) es posible utilizar diferentes modos de codificar los mensajes.
 - ☐ b) existen muchos fabricantes distintos de dispositivos de red.
 - ☒ c) existen muchas tecnologías de comunicaciones diferentes.
 - ☐ d) podemos usar muchos protocolos distintos.
- 3** [1p] ¿Por qué los sistemas distribuidos ofrecen un buen nivel de escalabilidad?
- ☒ a) Al tratarse de sistemas multiproceso, es sencillo crear nuevos procesos cuando crece la cantidad de usuarios o peticiones.
 - ☒ b) Su naturaleza descentralizada permite añadir fácilmente nuevos recursos conforma aumente la carga de trabajo.
 - ☐ c) Los sistemas distribuidos no escalan bien.
 - ☐ d) Pueden crecer indefinidamente.
- 4** [1p] ¿Con cuál de los siguientes está más relacionado el concepto de «interface»?
- ☒ a) API
 - ☐ b) librería
 - ☐ c) servidor
 - ☐ d) proceso
- 5** [1p] ¿Cómo gestiona un SSDD la heterogeneidad?
- ☐ a) Impide la integración de recursos heterogéneos para lograr un ecosistema homogéneo.
 - ☐ b) Impide la integración de recursos homogéneos para lograr un ecosistema heterogéneo.
 - ☒ c) Ofrece acceso a recursos heterogéneos presentándolos como si fueran homogéneos.
 - ☐ d) Ofrece acceso a recursos homogéneos presentándolos como si fueran heterogéneos.
- 6** [1p] ¿En qué aspectos de la implementación de un protocolo nos puede ayudar «Google Protocol Buffers»?
- ☐ a) sincronización de mensajes.
 - ☒ b) marshalling y unmarshalling.
 - ☐ c) creación de interfaces.
 - ☐ d) gestión de versiones.
- 7** [1p] Suponga que realiza una implementación en Python del cliente y servidor y utiliza los métodos de serialización/deserialización para los mensajes de petición y de respuesta proporcionados por el módulo struct. ¿Cuál de los siguientes afirmaciones es FALSA?
- ☐ a) Si se modifica el formato del mensaje a enviar, es necesario modificar también el código de la aplicación cliente/servidor.
 - ☒ b) El sistema es menos propenso a errores.
 - ☐ c) Es más eficiente que otros formatos de representación de datos como XML o JSON.
 - ☐ d) No permite desarrollo evolutivo.
- 8** [1p] ¿Cuál de las siguientes alternativas de serialización permite obtener mensajes de menor tamaño?
- ☐ a) XML
 - ☐ b) JSON
 - ☒ c) Python struct
 - ☐ d) protocol buffers

- 9** [1p] Está implementando un middleware para hacerle la competencia a ZeroC Ice, su primer cliente le pide mecanismo de RPC no bloqueante y no fiable. Al usar enlaces de comunicaciones de pago por byte transmitido, le pide que minimize los bytes transmitidos ¿que modelo de intercambio de mensajes es mas adecuado?
- ☒ a) Request (R) ☐ c) Request-Reply-Ack (RRA)
- ☐ b) Request-Reply (RR) ☐ d) Serialización JSON
- 10** [1p] Como programador de aplicaciones distribuidas, su empresa empieza a usar un middleware RPC ¿De qué aspectos ya no se tendrá que preocupar?
- ☐ a) Serialización
- ☐ b) Conexión y desconexión
- ☐ c) Uso de las primitivas del interfaz socket del sistema operativo
- ☒ d) No me tengo que ocupar de ninguna de las otras opciones
- 11** [1p] Señale la opción que se realiza. En una invocación RPC, el proceso llamado que hace las funciones de servidor y ejecuta el *stub* del servidor:
- ☒ a) Serializa el resultado de la llamada al procedimiento local
- ☐ b) Serializa los argumentos de la llamada
- ☐ c) Deserializa el resultado de la llamada al procedimiento local
- ☐ d) Envía el mensaje de petición
- 12** [1p] Señale la opción que se realiza. En una invocación RPC, el proceso que llama y que hace las funciones de cliente y ejecuta el *stub* del cliente:
- ☒ a) Construye el mensaje de petición de invocación
- ☐ b) Deserializa los argumentos de la llamada
- ☐ c) Serializa y envía el resultado de la llamada al procedimiento local
- ☐ d) Todas las funciones anteriores son realizadas
- 13** [1p] Qué semántica requiere en su implementación, de un mecanismo de ACK pero no control de duplicados
- ☒ a) at-least-once ☐ c) maybe
- ☐ b) at-most-once ☐ d) exactly-once
- 14** [1p] Qué semántica podría implementar sólo y exclusivamente con el protocolo de la capa de transporte UDP sin ningún mecanismo adicional (ni ACK, ni duplicado de filtros, etc.)
- ☒ a) maybe ☐ c) exactly-once
- ☐ b) at-most-once ☐ d) at-least-once
- 15** [1p] El portmapper simplifica:
- ☒ a) La vinculación cliente/servidor ☐ c) La deserialización/serialización de argumentos
- ☐ b) El intercambio de mensajes entre cliente y servidor ☐ d) La semántica de implementación