



## Ejercicios de repaso y autoevaluación

### 1. ¿Cómo surge la programación basada en componentes?

Surge basandose en la idea de reutilizacion del código, idea base de la programación orientada a objetos, que no existia en la anterior metodología de programación

### 2. Marque con una X los enunciados correctos.

	Respuestas HTTP
✓	La identificación de los componentes se lleva a cabo a través de la ingeniería del dominio.
✓	Existe un mercado de <i>software</i> al que uno pueda dirigirse para obtener los componentes.
✓	La calificación de componentes permite asegurar que la funcionalidad del componente encaja.
✓	Un componente queda caracterizado por su interfaz.
✓	Es necesaria una etapa de validación para determinar si un componente cumple o no los requisitos.

El libro dice que no, pero si que existe ver el caso de ODOO

### 3. Cuando un componente ha sido cualificado y es válido para formar parte del sistema, ¿qué dos escenarios posibles existen?

- ✓ a. El componente se integra o no.
- b. El componente se valida o no.
- c. El componente se reutiliza o no.
- d. El componente se codifica o no.

### 4. La composición de componentes es el proceso que se encarga de crear el sistema a partir de todos los componentes disponibles. Pero la integración no es algo sencillo. ¿Cuáles son los tipos de composiciones que se pueden hacer?

- ✓ a. Composición secuencial, jerárquica y aditiva.
- b. Composición secuencial, jerárquica y sustractiva.

- c. Composición secuencial, arbórea y aditiva.
- d. Composición paralela, jerárquica y aditiva.

**5. Complete los siguientes enunciados.**

Un Array es una colección de datos del mismo tipo almacenados consecutivamente en memoria.

Una Estructura es una colección de diferentes tipos de datos que se agrupan.

Un tipo Enumerado es una colección de elementos identificados alfanuméricamente en el momento de definir el tipo enumerado y en el que cada elemento corresponde a un número entero.

**6. Rellene la siguiente tabla de verdad en función de la operación.**

		Y	O
Operador1	Operador2	Conjunción	Disyunción
Falso	Falso	Falso	Falso
Falso	Cierto	Falso	Cierto
Cierto	Falso	Falso	Cierto
Cierto	Cierto	Cierto	Cierto

**7. ¿Cuáles son las tres estructuras de control que definen la programación estructurada?**

- ☒ a. Secuenciales, condicionales y concurrentes.
- ☒ b. Secuenciales, condicionales e iterativas.
- c. Repeticiones, permutaciones y condicionales
- d. Iterativas, predictivas y acumulativas.

**8. ¿Cuáles son las características de la programación estructurada?**

- Posee diseño modular
- Los módulos se diseñan en modo ascendente o descendente
- Cada módulo usa estructuras de tipo secuencial, condicional y /o iterativa
- Se complementan la modularidad y la estructuración

**9. ¿Cuál de estas afirmaciones es correcta?**

- ☒ a. Un *thread* o hilo es un flujo secuencial simple dentro de un proceso.

- b. En *Linux*, no se puede crear un nuevo proceso mientras haya otro en ejecución.
- ✓ c. El interbloqueo es una anomalía que sucede cuando varios procesos intentan acceder de forma simultánea. **si se añade a un recurso del sistema es correcta**
- ✓ d. Los semáforos son una muy buena solución al problema de la exclusión mutua.

**10. De las siguientes afirmaciones, indique cuál es verdadera o falsa.**

- a. Un monitor es una estructura de más alto nivel que el semáforo.  
☒ Verdadero  
☐ Falso
- b. Cada proceso, cuando se abre un fichero, añade una entrada en la tabla de ficheros asociada.  
☒ Verdadero  
☐ Falso
- c. La interrupción es el mecanismo mediante el cual es posible interrumpir la ejecución del programa ejecutado por la CPU.  
☒ Verdadero  
☐ Falso
- d. DMA no permite que los dispositivos accedan a la memoria del sistema.  
☐ Verdadero  
☒ Falso

**11. ¿En qué se basa el control de calidad del software?**

Se basa en un conjunto de actividades aplicables en las etapas del ciclo de vida del software desde el análisis hasta el mantenimiento que aseguran que el software está desarrollado y mantenido acorde al standard de programación elegido

---

---

---

**12. De las siguientes afirmaciones, indique cuál es verdadera o falsa.**

- a. Una métrica es una magnitud que permite su revisión o evaluación y corresponde a un posible atributo o requerimiento del *software*.  
☒ Verdadero  
☐ Falso

b. La captura de los requerimientos tiene como principal objetivo llegar a un entendimiento profundo de lo que debe y no debe hacer el sistema que se está desarrollando.

- ✓ ☐ Verdadero  
☐ Falso

c. La validación de requerimientos permite comprobar que estos definen el sistema que el cliente busca.

- ✓ ☐ Verdadero  
☐ Falso

d. En una revisión de requerimientos, el cliente o cualquier *stakeholder*, involucrado de manera formal o informal, debe poder verificar que el documento de requerimientos no presente anomalías ni omisiones.

- ✓ ☐ Verdadero  
☐ Falso

**13. ¿Cuál es la finalidad de un gestor de paquetes en un sistema operativo?**

Instala, desinstala, configura, y actualiza un software.

También verifica si el software a instalar tiene las dependencias que requiere y si necesita descargarlas e instalarlas, debe proveer ese servicio

**14. ¿Por qué es necesario el uso de optimizadores de código?**

Para mejorar la eficiencia del software, disminuir los recursos a utilizar en general para dotar al programa de mayor rapidez de ejecución, y para disminuir la cantidad de memoria que usa. Aunque con la evolución del hardware, cada vez va siendo menos necesaria esta optimización.

---

---

---

**15. ¿En qué consiste el despliegue de software?**

consiste en la instalación del software desarrollado en los dispositivos propios del cliente para los que ha sido desarrollado dicho software poder usar sus funcionalidades de forma completa.

---

---

---

---