Tomados del examen de certificación como desarrollador de software de IEEE

1. Los elementos típicos del proceso de requerimientos son:

I. Análisis del problema
II. Diseño de software
III. Análisis de las necesidades del personal de staff
IV. Especificación de comportamiento externo.

[a] I y IV únicamente
[b] II and III únicamente
[c] I, III, y IV únicamente
[d] I, II, y III únicamente

2. ¿La Ingeniería de Dominio consiste de cuáles de los siguientes conjuntos de actividades?

a) Modelación de datos, adquisición de requerimientos, y verificación de requerimientos.
b) Análisis, modelación y desarrollo de infraestructura
c) Clasificación de componentes, diseño de biblioteca, y poblar la biblioteca.
d) Entender a la comunidad de usuarios, construir un modelo ER y establecer rastreo (traceability)

3. ¿Qué actividades necesitan completarse antes de realizar el alojamiento de los requerimientos del sistema?

I. Arquitectura del Sistema
II. Requerimientos del sistema y arquitectura del sistema
III. Requerimientos del sistema y requerimientos de software.

[a] I
[b] II
[c] I y III
[d] III

4. En un proyecto grande de tiempo real, los siguientes elementos (o componentes) del sistema fuero incluidos como parte del diseño arquitectónico:

I. Pre-procesamiento de la señal de entrada
II. Modulo principal de procesamiento
III. Interfaz de red

¿Dónde debería alojarse el requerimiento no funcional de confiabilidad?
[a] I
[b] I y II
[c] II y III
[d] I, II y III

5. De acuerdo al estándar IEEE Standard 830-1998, ¿cuál de las siguientes son características de una buena especificación de requerimientos de software?

I. Completa
II. Jerárquica
III. Verificable
IV. Probable (se puede probar)
V. Rastreable

[a] I, II, y IV
[b] I, III, y V
[c] II, III, y IV
[d] II, III, y V

6. ¿Cuál de los siguientes NO es un atributo de una especificación de requerimientos de software bien formada?

a) Todo lo que supuestamente debe hacer el software, está incluido en la especificación
b) Cada requerimiento incluido en la especificación tiene sólo una interpretación
c) Alguno de los requerimientos especifican una arquitectura de software objetivo
d) Todos los requerimientos son entendibles por clientes no especialistas en computación

7. Un diseño que puede modificarse fácilmente para correr en una variedad de ambientes de software y hardware es altamente:

[a] portable
[b] interoperable
[c] rentable
[d] usable

8. Los elementos una arquitectura de software de un sistema computacional incluyen

I. Componentes de software
II. Diagramas de clases
III. Conectores expresando las relaciones entre componentes de software.
IV. Diagramas Entidad Relación.

[a] I y II
[b] I y III
[c] I, III, y IV
[d] I, II, III, y IV

9. La descripción del diseño de software detalla todo lo siguiente EXCEPTO:

a) Descripción de input-output para elementos de software
b) Base lógica para el diseño de los elementos de software.
c) Ejecución conceptual, incluyendo flujo de datos y flujo de control
d) Modelo de ciclo de vida del software

A. ¿Cuál de estos diagramas NO es usado al realizar un diseño orientado a objetos?

[a] class
[b] activity
[c] use-case
[d] sequence

B. Un buen diseño realiza todo lo siguiente EXCEPTO:

a) Implementa todos los requerimientos explícitos e implícitos
b) Provee información para desarrolladores y "testers"
c) Menciona o incluye el domino de datos, funcionalidad y comportamiento desde una perspectiva de implementación
d) Define una metodología de revisión del software.

C. Todos los elementos siguientes son aspectos de diseño para un programa de un componente. EXCEPTO:

[a] Estructuras de control
[b] Algoritmos
[c] Requerimientos
[d] Estructuras de datos

D. La principal ventaja de la programación estructurada es:

a) Es más eficiente
b) Tiende a ser más confiable
c) Es más fácil de escribir
d) Puede ser descrita con un diagrama de flujo

E. El aspecto más importante de pruebas estructurales (caja blanca) es su habilidad para:

a) Revelar la presencia de defectos en varias partes del código
b) Establecer la exactitud del módulo.
c) Probar que cada elemento en el módulo es alcanzable
d) Probar que el módulo tiene una complejidad ciclomática baja

F. ¿Cuál de los siguientes son entradas para el proceso de pruebas de sistema?

I Requerimientos de Rendimiento del sistema
II Requerimientos funcionales del sistema.
III Código de programa
IV Especificación del diseño del sistema

[a] I y II
[b] I, II y IV
[c] I
[d] I, II, III y IV

10 Un test oracle:

[a] Genera predicciones de resultados esperados de las pruebas.
[b] Administra las corridas de pruebas a los programas.
[c] Genera datos de pruebas para los programas que serán probados.
[d] Cuenta el número de veces que un estatuto en particular ha sido ejecutado.

11 ¿Cuál de los siguientes tipos de planes de prueba es el que surgirá a partir del proceso de especificación de requerimientos?

a) Plan de pruebas de integración del sistema
b) Plan de pruebas de aceptación
c) Plan de pruebas de integración de subcomponentes
d) Pan de pruebas unitarias por módulo.

12 Un buen diseño basado en descomposición modular tiene como característica que sus módulos:

a) Presentan alto acoplamiento y baja cohesión
b) Presentan bajo acoplamiento y baja cohesión
c) Presentan bajo acoplamiento y alta cohesión
d) Presentan alto acoplamiento y alta cohesión

13 Cuál de las siguientes es la mejor medida de acoplamiento

a) Acoplamiento de control, ya que las estructuras se pasan información de control
b) Acoplamiento común, ya que las estructuras comparten la misma área de datos global
c) Acoplamiento por datos, ya que las estructuras se comunican a través de parámetros
d) Acoplamiento por estampas, ya que las estructuras comparten la misma estructura de datos

14) Clasifica los siguientes patrones como: **creational design patterns**, **structural design patterns** and **behavioral design patterns**:

(a) Abstract Factory	(d) Singleton
(b) Facade	(e) Observer
(c) Proxy	(f) Prototype