

Instrucciones:

- Cada pregunta múltiple opción contestada correctamente tiene un valor de 3,34 puntos. Cada pregunta incorrecta de la múltiple opción resta 1,11 puntos.
- **El puntaje total del examen es 100 puntos y se aprueba con 60 o más puntos.**

## Múltiple Opción

1. Sobre la liberación de un sistema:

- a) La liberación de un sistema puede incluir actividades de entrenamiento a usuarios del mismo
- b) a) y cuando se sustituye un sistema por otro, es necesario planificar una estrategia de conversión, por ejemplo: big-bang, conversión paulatina, procesamiento en paralelo.
- c) b) y también puede llegar a ser necesario realizar una carga inicial de información histórica en el nuevo sistema
- d) c) y en cuanto a la documentación del sistema, es importante considerar su legibilidad, por ejemplo respecto a estructura y tamaño.

2. Con respecto a las revisiones e inspecciones:

- I. Como parte de la revisión se debe chequear la consistencia y completitud de los objetos que estén bajo revisión ya sean documentos o código. También es importante verificar que se sigan los estándares de calidad definidos.
- II. La documentación de los hallazgos de la revisión es opcional, ya que lleva un tiempo considerable y lo más importante es detectar y corregir los posibles defectos cuánto antes.
- III. Las inspecciones de código son un buen método para evaluar a los desarrolladores.
- IV. El proceso de revisión por lo general está dividido en 3 fases. En primer lugar se tiene una fase preparatoria en la cual, entre otras cosas, se conforma el equipo de revisión, se definen las fechas de la revisión y se entregan los objetos a revisar. Luego se realiza la reunión de revisión. Finalmente se cuenta con una fase post-revisión en la que se abordan los defectos encontrados en la revisión y se realizan los cambios que se consideran necesarios.
- V. Las revisiones son más fáciles de adoptar cuando se usan metodologías ágiles que cuando se usan metodologías tradicionales (basadas en planes).

Seleccione la opción correcta:

- a) Todas son verdaderas.
- b) Tres opciones son verdaderas.
- c) Sólo (I) es verdadera.
- d) Sólo (I) y (IV) son verdaderas.

3. Con respecto al mantenimiento de software:

- a) Cualquier modificación realizada a un sistema luego de entrar en operación se considera mantenimiento
- b) (a) y existe una clasificación que categoriza a los sistemas en: S, P y E
- c) (b) y un sistema que calcula la integral de una función en un intervalo es un sistema de tipo S
- d) (c) y un sistema de facturación electrónica es un sistema de tipo P.

4. Con respecto a las pruebas de módulos y a las técnicas de integración:

- I. Normalmente el *Stub* es el que suministra los datos de prueba y se usa para simular una llamada.
- II. Debido a que el *Stub* simula la actividad del componente omitido, es más costoso de realizar que un *Driver*.
- III. Big-Bang, Sandwich y por disponibilidad son estrategias incrementales, ya que para realizar la integración primero deben construirse los componentes.
- IV. Big-Bang es una buena elección ya que no es necesario construir *Stubs* ni *Drivers* y, además, es posible comenzar la integración con algunos módulos.
- V. La estrategia Bottom-up requiere de *Drivers* pero no de *Stubs*. Por su parte, Top-Down requiere de *Stubs* pero no de *Drivers*. Finalmente, la estrategia Sandwich busca probar primero los módulos más críticos.

Seleccione la opción correcta:

- a) Todas son correctas.
- b) Sólo 1 es correcta.
- c) Sólo 2 son correctas.
- d) Ninguna es correcta.

5. El MUM es un modelo de proceso:

- a) en cascada, dividido en fases, donde cada fase corresponde a una disciplina.
  - b) iterativo incremental, con liberaciones parciales indicadas
  - c) iterativo incremental, con evaluaciones parciales.
  - d) incremental en fases, pero no iterativo.
6. Un diseño de arquitectura cliente-servidor de tres niveles (presentación, lógica y datos), con un cliente fino:
- a) suele requerir menos infraestructura del lado del cliente comparando con esquemas de “cliente grueso”.
  - b) (a) y siempre es el modelo más flexible y escalable.
  - c) (a) pero suele ser difícil de administrar.
  - d) (a) y tiene como desventaja tiempos de respuestas más altos.
7. Respecto de la gestión de riesgos
- a) Es esperable que al inicio de un proyecto no exista ningún riesgo y que estos vayan surgiendo a medida de que el proyecto se desarrolla
  - b) En las etapas finales de un proyecto (liberación e implantación de la aplicación) ya no tiene sentido gestionar los riesgos.
  - c) Cuando un riesgo desaparece (probabilidad 0 de ocurrencia) necesariamente otro emerge.
  - d) Ninguna es correcta
8. Respecto a verificación y validación:
- a) la verificación busca comprobar que el sistema cumple con los requerimientos especificados (funcionales y no funcionales) y la validación que el software hace lo que el usuario espera.
  - b) a) y en técnicas de caja negra, las pruebas pueden derivarse solamente de la especificación, por lo que interesa conocer la funcionalidad y no así su implementación
  - c) b) y respecto a las técnicas estáticas, son efectivas en la detección temprana de defectos, no se utilizan para verificación, solo para validar.
  - d) c) y respecto a las técnicas dinámicas, sirven tanto para verificar como para validar.
9. Entre las características de un buen diseño se encuentra la independencia de los elementos de software definidos donde:
- a) la medida de esta independencia está dada por la cohesión y acoplamiento entre módulos, donde lo que se busca obtener es baja cohesión y alto acoplamiento.
  - b) la cohesión de un módulo refiere a cuan conectado está el módulo con el resto de los módulos definidos.
  - c) el acoplamiento de un módulo refiere a cuan focalizado está el módulo en resolver determinado aspecto del problema.
  - d) cuanto mayor es la independencia se facilitan aspectos como la comprensión y el mantenimiento del diseño.
10. Sobre la calidad del producto y sus factores:
- a) La calidad del proceso es un factor fundamental.
  - b) El ciclo de mejora de un proceso consta de tres etapas: Cambio, Medición y Análisis.
  - c) Otros factores que afectan a la calidad del producto son: la calidad del personal, la tecnología de desarrollo, los costos y plazos.
  - d) Todas las anteriores son correctas.

11. Todo jefe de proyecto debe lograr motivar a las personas que integren el equipo para que cada uno aporte sus mejores habilidades. En cuanto a la motivación es cierta la siguiente afirmación:
- Para satisfacer las necesidades de realización personal, es necesario darle a la persona responsabilidad por su trabajo, asignarle tareas exigentes (pero no imposibles) y proveer un programa de capacitación orientado a desarrollar las habilidades de cada persona.
  - Para lograr motivar a una persona basta con que sus necesidades fisiológicas estén cubiertas y se sienta seguro.
  - Cuando un grupo es cohesivo, la mayoría de los integrantes no se sentirán motivados.
  - La motivación no es algo que dependa de las personalidades individuales. Una misma técnica de motivación sirve para cualquier personalidad.
12. El modelo de desarrollo iterativo incremental, puede utilizarse con un enfoque ágil o uno dirigido por planes o uno mixto. En lo que refiere a este modelo:
- Uno de los problemas de este modelo es que la estructura del sistema puede tender a degradarse a medida que se agregan nuevos incrementos.
  - a). Debido a esto, la mejor solución para mejorar el software es utilizando la técnica de refactorización, independientemente del enfoque que se aplique (ágil o dirigido por planes).
  - a). Resolver este problema no es posible en enfoques dirigidos por planes.
  - Ninguna de las opciones es correcta
13. Sobre las técnicas de verificación de caja blanca, dado el siguiente fragmento de código y afirmaciones, seleccione la opción correcta

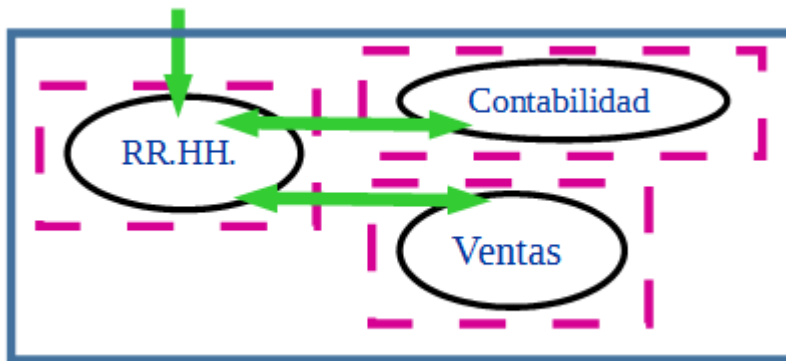
```
If (a > b) {
    print(a/b);
    x = a % 10;
    if (x != 0) {
        print(x);
    }
}
else {
    print(a*b);
}
```

- El caso de prueba  $\{b=0, a=12\}$  cumple el criterio de cubrimiento de sentencias.
  - Los casos de prueba  $\{b=0, a=0\}$ ,  $\{x=0, b=5, a=10\}$  cumplen el criterio de cubrimiento de decisión.
  - Los casos de prueba  $\{b=0, a=2\}$ ,  $\{x=0, b=0, a=0\}$  cumplen el criterio de cubrimiento de sentencias.
  - Ninguna de las anteriores es correcta.
14. Con respecto a la arquitectura de software:
- Una arquitectura de software es una descripción de cómo se organiza un sistema de software.
  - (a) donde determinadas propiedades de un sistema, como el rendimiento, seguridad y disponibilidad están influenciadas por la arquitectura utilizada.
  - (b). El cambio en la arquitectura de un producto ya construido suele requerir esfuerzo no despreciable.
  - (c) por lo que es conveniente realizar evaluaciones y validaciones durante su definición y construcción.
15. Construir un prototipo
- Sirve para relevar requisitos, porque puede usarse para definir la interfaz de usuario de forma que quienes la vayan a utilizar puedan experimentar su utilización
  - Resulta adecuado para relevar y validar requisitos del software, no así para evaluar aspectos del diseño del mismo
  - No sirve para relevar requisitos, porque consiste solamente en desarrollar una parte o un aspecto del producto para evaluar la factibilidad de implementar el enfoque elegido con la tecnología a utilizar.
  - Exige un esfuerzo adicional, que sólo se justifica si el prototipo puede evolucionar al producto final

16. Considere las siguientes afirmaciones:
- Es conveniente tener pruebas de regresión automatizadas

- II. Las pruebas unitarias son realizadas, en general, por los desarrolladores
  - III. Los casos de uso son un ejemplo de pruebas de caja blanca
  - IV. Es conveniente que las pruebas de sistema las realice un equipo especializado (verificadores), ya que en general manejan mejor las técnicas de prueba, y también pueden tener un buen conocimiento de los errores más comunes cometidos por los desarrolladores
  - V. Las pruebas de aceptación se realizan junto con el cliente, y tienen como objetivo validar el sistema
- a) Solamente las afirmaciones I, II, IV y V son verdaderas
  - b) Solamente las afirmaciones I y II son verdaderas
  - c) Solamente las afirmaciones I, II, III y IV son verdaderas
  - d) Todas las afirmaciones son verdaderas

17. Dada la siguiente realidad, con tres aplicaciones de RR. HH., Contabilidad y Ventas:



El usuario se comunica con el módulo de RR.HH. y este con los otros dos.

Consideramos dos opciones para trazar la frontera de la aplicación:

- Opción A (línea negra): la frontera del sistema engloba a los tres módulos
- Opción B (líneas punteadas): se trata de tres aplicaciones distintas, cada una con su frontera.

Se cumple que la cantidad de puntos de función en la opción A y la cantidad de puntos de función correspondiente a la suma de los PF de las tres aplicaciones consideradas por separado (opción B):

- a) Es igual
  - b) Es menor
  - c) Es mayor
  - d) No es posible saberlo
18. Con respecto al mantenimiento de software:
- a) Por lo general el software está más tiempo en la fase de mantenimiento que en la fase de desarrollo.
  - b) (a) y el mantenimiento perfectivo refiere a introducir cambios para prevenir fallas.
  - c) (a) y un problema muy común en el mantenimiento es el de la comprensión limitada del software que se está manteniendo.
  - d) Por lo general en el mantenimiento se suele destinar más tiempo a corregir fallas que a agregar nuevas funcionalidades o modificar existentes.
19. La Ingeniería de Software debe enfrentar varios desafíos:
- a) Uno de los principales es la demanda por aumentar los tiempos de entrega.
  - b) (a) y el desarrollo de software fiable.
  - c) Enfrentar la creciente diversidad y reducir los tiempos de entrega, entre otros.
  - d) Por ejemplo, según la ley de Moore, el costo del desarrollo de software crece exponencialmente.

20. Con respecto a los Casos de Uso:

- a) Un actor es otro sistema que interactúa con nuestro sistema pero que no realiza el caso de uso.
  - b) Un actor puede ser tanto una persona como otro sistema que interactúa con el nuestro. A su vez, las condiciones son aquellas condiciones que se deben cumplir para que el caso de uso pueda iniciarse.
  - c) Provee una descripción de alto nivel de los requisitos no funcionales del sistema.
  - d) Las poscondiciones no son elementos importantes de un caso de uso y pueden quitarse. En cambio las condiciones son un elemento esencial.
21. Son aspectos de un buen diseño de software:
- a) Completo.
  - b) a), trazable.
  - c) b), concreto.
  - d) c) y verificable
22. Dado el siguiente requerimiento de un sistema de control de gastos: “Todos los días a las 9:00 am se debe enviar por email al administrador del sistema un listado con todos los gastos e ingresos del día anterior”.
- a) Dado que el requerimiento no posee de interfaz gráfica no puede ser modelado con un caso de uso.
  - b) Dado que el interesado en la funcionalidad es el administrador es posible modelar el requerimiento como un caso de uso donde el Actor principal sea el Administrador.
  - c) a) y b) son correctas.
  - d) Ninguna de las anteriores es correcta.
23. En relación a la integración continua:
- a) Es muy poco utilizada en metodologías ágiles.
  - b) a) y se recomienda complementar con pruebas automatizadas.
  - c) En general, es una estrategia no recomendada para sistemas muy grandes o complejos.
  - d) Todas las anteriores son correctas.
24. La Madurez de una Organización se determina por:
- a) El desempeño del conjunto de sus procesos.
  - b) La capacidad del conjunto de sus procesos.
  - c) El promedio de proyectos exitosos a través de la aplicación del conjunto de sus procesos.
  - d) Ninguna de las anteriores.
25. La tarea de asegurar la calidad implica:
- a) identificar los atributos de calidad del producto
  - b) planificar la gestión de la calidad
  - c) revisar el ajuste a los procesos definidos que permitan obtener un producto de calidad
  - d) controlar la calidad de los productos obtenidos
26. Los requerimientos funcionales incluyen:
- a) Restricciones que especifican el comportamiento del producto, así como los que se derivan de los factores externos al sistema.
  - b) Restricciones que se derivan de políticas y procedimientos existentes en la organización del cliente y en la del desarrollador.
  - c) Declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares.
  - d) Todas las anteriores son correctas.

27. Indique cuál o cuáles de las siguientes técnicas sirven para evaluar/asegurar la calidad de los productos de software
- Revisiones e inspecciones.
  - Inventarios de pruebas.
  - Filtros de defectos.
  - Todas las anteriores.
28. Acerca de la Ingeniería de Software pueden enunciarse las siguientes afirmaciones:
- La Ingeniería de Software son los productos de software, incluidos los documentos y los modelos de desarrollo de software.
  - A diferencia de la Ciencia de la Computación, tiene que ver con los aspectos prácticos de desarrollo y distribución de software.
  - Aceptabilidad es un atributo de calidad, el cual es fundamental, ya que el cambio de software es un requisito inevitable.
  - La Ingeniería de Software es una disciplina dentro de la Ingeniería, que desarrolla todos los aspectos relacionados a la producción de software.
  - Las actividades fundamentales de la Ingeniería de Software son especificación, verificación y diseño de la arquitectura.
- Seleccione la opción correcta:
- Todas son correctas.
  - I, IV y V son correctas.
  - II y III son correctas.
  - II y IV son correctas.
29. La gestión de versiones en general incluye:
- Identificación de la versiones y de las liberaciones.
  - Desarrollo dependiente.
  - Registro histórico de cambios.
  - Dos opciones de las anteriores son correctas.
30. Respecto a casos de uso:
- Son independientes del método de diseño y lenguaje que se utilice para la implementación.
  - Se pueden utilizar como herramienta para el relevamiento de los requisitos.
  - El actor es un agente externo al sistema que participa en la historia de un caso de uso y está representado por el papel que cumple dentro del caso de uso. Solo seres humanos pueden ser actores en un casos de uso.
  - Las postcondiciones describen el estado del sistema luego de la ejecución exitosa del casos de uso.
  - La relación de extensión es la más común entre casos de uso. Permite definir casos de uso que «ejecuten» otros casos de uso.
  - La relación de inclusión entre casos de uso se refiere a un fragmento de un caso de uso que extiende, es decir, agrega comportamiento, a otro caso de uso.
- Las afirmaciones I), II) y III) y IV) son correctas.
  - Las afirmaciones, I), II), III), IV), V) y VI) son correctas.
  - Las afirmaciones I), II), IV), V) y VI) son correctas.
  - Las afirmaciones III), V) y VI) son incorrectas.