

- 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

- Entre las dificultades que se presentan para tomar medidas sobre productos de software se pueden mencionar:
  - a) Hay una resistencia de las personas a tomar medidas porque éstas son siempre objetivas.
  - b) Es difícil tomar medidas directas sobre un producto de software.
  - c) **(b) y se miden ciertos atributos internos y se asume que están relacionados con los externos que se quieren evaluar.**
  - d) Siempre es conveniente tomar estas medidas una vez haya finalizada la construcción del producto.

---

- Las pruebas de sistema:
  - a) Deben enfocarse en testear las interacciones entre los distintos componentes y objetos que conforman un sistema.
  - b) **a) y por este motivo, tomar como base los casos de uso es una manera efectiva de realizar pruebas de sistema.**
  - c) b) y en caso de ejecutar todas las pruebas sin error, se garantiza el correcto funcionamiento del sistema.
  - d) c) y en general, son más sencillas de automatizar que las pruebas unitarias.

---

- Indique cual de las siguientes NO es una ventaja de Tubos y Filtros
  - a) **Fue conceptualizado para el diseño y construcción de sistemas con alto grado de interacción con el usuario.**
  - b) Es sencillo agregar nuevas transformaciones.
  - c) Es posible reutilizar los filtros.
  - d) Es sencillo conceptualizar el diseño como secuencias de procesamientos de datos.

---

- En los métodos ágiles los requerimientos son desarrollados de forma incremental, el software es entregado en incrementos cortos y rápidos . Estas características requieren un enfoque particular de gestión de proyectos. Un ejemplo es el enfoque de Scrum, alguna de las características de ese método son:
  - a) Scrum consiste en 4 fases: Planificación, Diseño, Desarrollo y Cierre de Proyecto.
  - b) La fase central, los ciclos de sprint, concentra las actividades de desarrollo y testing, pero no abarca la entrega del ejecutable.
  - c) **Los sprints son normalmente de 2 a 4 semanas. Al inicio de un Sprint se revisa el product backlog, donde, a su vez, el cliente puede incluir nuevos requerimientos.**
  - d) Los Sprint son por naturaleza de largo variable, y se van definiendo a lo largo del propio Sprint. Pueden ir de 2 a 4 semanas según la cantidad de requerimientos que se tenga que desarrollar en el mismo.

---

- Respecto a la ingeniería de software:
  - a) No aplica a procesos de desarrollo ágiles dado que estos se centran principalmente en la construcción y no en la planificación del proyecto.
  - b) Si comenzamos desarrollando un prototipo evolutivo el cual se va refinando según las necesidades del cliente, no es necesario planificar el proyecto ni seguir un proceso de desarrollo de software.
  - c) **Definir el proceso de desarrollo, la planificación y estimar el esfuerzo inicial reduce los riesgos en cuanto a no cumplir con los objetivos del proyecto.**
  - d) Dos de las anteriores son correctas.

---

- Para el tema mantenimiento de software, seleccione la opción correcta:
  - a) Lehman afirma que si los sistemas no cambian, se vuelven menos útiles.
  - b) **a) y la complejidad de los sistemas tiende a crecer a menos que se tomen acciones para disminuirla.**
  - c) La edad del equipo de mantenimiento, es uno de los factores que tiene mayor incidencia sobre la calidad del mantenimiento.
  - d) En general, respecto la distribución del esfuerzo de mantenimiento, se invierte el 50% del esfuerzo a mantenimiento correctivo.

7. Sommerville define al objetivo de la verificación como:
- Comprobar que el sistema cumple con los requerimientos funcionales, y los no funcionales no se verifican.
  - Comprobar que el sistema cumple con los requerimientos funcionales y no funcionales.**
  - Comprobar que el software cumple las expectativas del cliente.
  - Comprobar que estamos construyendo el producto correcto.

8. La ingeniería de software se ha visto afectada por:
- la necesidad por reducir los tiempos de salida al mercado.
  - (a), así como por la preocupación por el proceso de software y la arquitectura de software .**
  - (b) A su vez, el desarrollo y éxito de algunas startups han demostrado la invalidez de determinados procesos de desarrollo.
  - (c), principalmente por la presión creciente por reducir los costos de producción del software.

9. Dadas las siguientes afirmaciones sobre Diseño del Sistema:

- Uno de los objetivos del diseño orientado a objetos es conseguir una alta cohesión entre las clases y un bajo acoplamiento.
- Bajo acoplamiento significa tener en un mismo módulo elementos que estén fuertemente vinculados para que los mismos sean más fáciles de entender y modificar.
- El principal objetivo del principio Abierto-Cerrado es promover la construcción de sistemas que sean fáciles de modificar.
- La cohesión indica, por ejemplo, el nivel de relación existente entre las distintas operaciones de un módulo.

Seleccione la opción correcta:

- Solo las afirmaciones I), II) y III) son correctas.
- Solo las afirmaciones I), III) IV) son correcta.**
- Solo las afirmaciones II) y IV) son correctas.
- Todas las afirmaciones son correctas

10. Acerca de patrones y estilos de diseño de software puede decirse que:

- Los Idioms son soluciones de organización a nivel de sistema.
- Los estilos arquitectónicos son soluciones útiles para problemas específicos que dependen del lenguaje de programación.
- Un patrón de diseño describe cómo implementar aspectos particulares de elementos usando características de un lenguaje particular.
- A partir de la definición de los principios organizativos del sistema, el uso de un estilo arquitectónico permite seleccionar una solución entendible y probada a ciertos problemas.**

11. ¿Cuáles son las ventajas de utilizar subsistemas dentro de un producto de software?

- Reutilización, abstracción, posibilidad de reemplazo dinámico.
- La posibilidad de paralelización de la implementación de los diferentes subsistemas.
- La posibilidad de aplicar más de un patrón arquitectónico, ya que de otra manera no sería posible.
- Dos de las anteriores son correctas.**

12. En cuanto a la planificación del proyecto

- Debe realizarse de forma exhaustiva al comienzo del proyecto, dado que es el mejor momento para planificar todas las tareas, actividades y recursos que se necesitarán.
- El nivel de planificación que se puede alcanzar al inicio del proyecto depende, entre otras cosas, de la volatilidad de los requisitos.**
- Es independiente de los requisitos, dado que todo sistema se puede llevar a una arquitectura en capas. La planificación consiste en estimar el esfuerzo necesario para desarrollar y probar cada una de las capas.
- (c). Finalizada la integración, se planifica la etapa de pruebas del sistema y la liberación.

13. Indique la opción INCORRECTA sobre el tema requisitos:

- Los requisitos de un sistema son descripciones de lo qué debería hacer el sistema.
- Los requisitos se dividen en tres grandes áreas: de producto, organizacionales y externos.
- La mejor manera de especificar un requisito es con un caso de uso.**
- Realizar entrevistas para recabar requisitos es costoso pero se puede obtener información muy valiosa.

---

14. Sobre los de Casos de Uso:

- a) El diagrama de Casos de Uso provee una representación de bajo nivel de los requisitos de usuario.
  - b) **El diagrama de Casos de Uso Provee una representación de alto nivel de los requisitos de usuario.**
  - c) (a) y la descripción es un elemento esencial de un caso de uso.
  - d) (b) y el identificador del caso de uso pueden no ser único.
- 

15. Dadas las siguientes afirmaciones respecto a prototipos:

- I. Algunos de los objetivos de los prototipos son: validar los requisitos, encontrar errores y omisiones.
- II. Los mock-ups son prototipos utilizados para validar la factibilidad de la arquitectura del sistema.
- III. Las pruebas de concepto son prototipos horizontales, en los que se implementa una funcionalidad para realizar un estudio de factibilidad y mitigar riesgos.
- IV. Los prototipos evolutivos buscan la obtención rápida del mismo para resolver incertidumbre y responder dudas, en ejemplo de este tipo de prototipo son los wireframes.

Seleccione la opción correcta:

- a) Dos afirmaciones son correctas.
  - b) **Una afirmación es correcta.**
  - c) Todas las afirmaciones son correctas.
  - d) Ninguna afirmación es correcta.
- 

16. Indique que afirmación es correcta sobre la especificación de requisitos.

- a) Las especificaciones estructuradas son una aproximación a los requisitos donde la libertad del escritor de los requisitos es limitada y los requisitos son escritos de una manera estándar.
  - b) **a) y esto resulta adecuado para algunos tipos de requisitos ej: sistemas de control embebidos.**
  - c) b) y esto significa que los requisitos son escritos en frases en lenguaje natural.
  - d) Ninguna afirmación es correcta.
- 

17. Un estándar provee un cuerpo de conocimiento que:

- I. Se aplica obligatoriamente en una disciplina
- II. Guía el desempeño de una disciplina profesional.
- III. Se elabora por consenso entre personalidades, organizaciones y universidades reconocidas en la materia
- IV. Se impone desde la dirección de las empresas.
- V. Es de suma utilidad para evaluar al personal de una empresa.

Seleccione la opción correcta:

- a) Todas son correctas.
  - b) Únicamente I y IV son correctas.
  - c) Únicamente II, III y V son correctas.
  - d) **Únicamente II y III son correctas.**
- 

18. Sobre las distintas estrategias para realizar las pruebas de integración:

- a) La estrategia Big Bang es recomendable para sistemas grandes, ya que se invierte menos tiempo en realizar pruebas de integración por módulo.
  - b) Utilizando la estrategia Bottom-Up los módulos se van integrando a medida que están disponibles, teniendo en cuenta siempre probar aquellos más críticos lo antes posible.
  - c) **Las pruebas no incrementales suelen ser más costosas que las incrementales ya que requieren la implementación de una gran cantidad de drivers y stubs.**
  - d) Utilizando la estrategia de integración por disponibilidad, puede resultar complejo identificar el origen de una falla ya que se prueban todos los módulos a la misma vez.
- 

19. Con respecto a los requisitos

- a) No es necesario gestionar los cambios de los requisitos a partir de que son relevados.
  - b) Los requisitos de software son importantes únicamente al inicio del desarrollo de un sistema.
  - c) Entre las técnicas de obtención de requisitos se tienen: Casos de Uso, Tormenta de ideas, Revisiones y Encuestas.
  - d) **Un requisito puede ser verificable aunque no se pueda demostrar formalmente.**
-

---

20. Sobre la integración continua:

- a) No es recomendable si utilizamos metodologías ágiles de desarrollo.
  - b) Se comporta muy bien para sistemas muy grandes o complejos.
  - c) **Se recomienda complementar con pruebas automatizadas.**
  - d) No permite resolver rápidamente posibles conflictos entre desarrolladores.
- 

21. Las pruebas de performance:

- a) Buscan demostrar que el sistema cumple con todos sus requisitos funcionales.
  - b) Se ejecutan una vez que el sistema está completamente integrado.
  - c) **b) y un ejemplo son las pruebas de stress, que implican someter al sistema a grandes cargas o esfuerzos considerables.**
  - d) c) y se deben detener estas pruebas cuando el sistema soporta las cargas especificadas en su diseño.
- 

22. En cuanto a la cantidad de personal requerido para un proyecto de desarrollo:

- a) puede calcularse dividiendo el esfuerzo estimado entre el tiempo de desarrollo deseado.
  - b) Será el mismo independientemente de la fase de desarrollo del proyecto.
  - c) **Cuanto más personas trabajan en un proyecto, el esfuerzo total no aumenta proporcionalmente, sino que es aún mayor, debido al incremento en las líneas de comunicación y el tiempo que esta consume.**
  - d) La incorporación de personal en últimas fases es una buena estrategia para adelantar la fecha de finalización del proyecto.
- 

23. La incertidumbre de las estimaciones en general tiende a:

- a) Disminuir a medida que el proyecto avanza, independientemente del proceso que se siga.
  - b) Aumentar a medida que el proyecto avanza, si la tecnología es desconocida o los requisitos son inestables.
  - c) **Mantenerse sin converger, si el negocio es desconocido y no se reduce el riesgo de haber malentendido los requisitos.**
  - d) Todas son correctas.
- 

24. Entre los objetivos del proceso de liberación de software encontramos:

- a) Entregar a todos los usuarios la documentación técnica del sistema.
  - b) Definir los atributos de calidad que el sistema debe cumplir en lo que refiere a la amigabilidad, usabilidad del sistema.
  - c) (b) y luego capacitar a los usuarios y administradores del sistema.
  - d) **Que los usuarios y administradores del sistema entiendan cómo utilizar el mismo, logrando esto por ejemplo a través de talleres y/o cursos de capacitación.**
- 

25. Sobre la gestión de la configuración:

- a) Los beneficios y costos de la Gestión de la Configuración están relacionados al alcance y al grado del formalismo.
  - b) a) donde el alcance corresponde al número de objetos comprendidos en la gestión de la configuración.
  - c) **b) y grado del formalismo es el control con el cual se realizan las tareas de la gestión de la configuración.**
  - d) a) donde el alcance corresponde al control con el cual se realizan las tareas de la gestión de la configuración.
- 

26. El mantenimiento de software puede involucrar:

- a) Adaptación a cambios en el ambiente, como ser cambios de hardware, y/o sistema operativo donde el software funciona.
  - b) Corrección de errores introducidos durante el desarrollo del software.
  - c) Adición de nuevas funcionalidades, debidas a cambios en los requerimientos.
  - d) **Todas las opciones son correctas.**
- 

27. Respecto a los Modelos de Proceso de Software:

- a) Son abstracciones del proceso de software que se utilizan para explicar los diferentes enfoques del desarrollo de software.
  - b) (a). El desarrollo incremental es un ejemplo de modelo de proceso.
  - c) **(b) El modelo prescriptivo indica cómo debería llevarse a cabo el proceso, en contraposición a los modelos descriptivos.**
-

d) (c) y no es conveniente combinar los modelos de proceso dado que cada uno define un conjunto de actividades estructuradas.

---

28. En el libro de referencia (Sommerville), se establecen algunas ventajas de la inspección de software sobre el testing (o pruebas dinámicas), entre las cuales encontramos:

- a) Con las inspecciones es posible detectar un espectro más amplio de defectos, como ser el cumplimiento de estándares o algoritmos implementados de manera poco eficiente.
- b) Las inspecciones permiten validar el software junto con el cliente.
- c) a) y b) son correctas.
- d) No existen dichas ventajas.

---

29. Las empresas recurren a la evaluación y mejora de sus procesos para:

- a) Elevar la calidad de sus productos de software.
  - b) Vencer a la competencia y ampliar el mercado.
  - c) (a) y reducir los costos y los plazos de salida al mercado.
  - d) Obtener certificaciones internacionales exclusivamente.
-

30. Dadas las siguientes actividades de un proyecto, en caso de necesitar acortar la duración del proyecto, ¿qué actividad intentarías reducir?

Actividad	Actividad anterior	Duración
Inicio	Ninguna	0
A	Inicio	1
B	Inicio	2
C	Inicio	6
D	A	10
E	B, C	1
F	C	2
G	D	3
H	E	9
I	F	1
Final	G, H, I	0

- a) B
- b) D
- c) F
- d) C