

- Cada pregunta de la **parte múltiple opción** contestada correctamente tiene un valor de **3,34** puntos. Cada pregunta incorrecta de la múltiple opción resta **1,11** puntos.
- Se aprueba con el 70% del puntaje.

## Múltiple Opción

- 1) La ingeniería de software:
- Es una disciplina madura y establecida con un elevado nivel de profesionalización, de forma análoga a la ingeniería civil.
  - Existe una fuerte relación entre la ingeniería de software y la ingeniería de sistemas, ya que el software que construimos va a formar parte de un sistema conformado por el software en sí, el hardware sobre el que corre y las personas que lo utilizan.
  - (b) y la creciente importancia y dependencia del software para el funcionamiento de la sociedad trae aparejado un creciente reclamo por niveles de calidad adecuados.**
  - A medida que se desarrolla, guarda cada vez menos relación con cuestiones legales, reglamentarias o éticas.
- 
- 2) Entre los atributos de calidad más relevantes para un producto de software para el control de la red de semáforos de una ciudad están:
- tiempo de respuesta, adecuación al uso, consumo de recursos, seguridad de la información (security)
  - tiempo de respuesta, adecuación al uso, consumo de recursos, seguridad de la integridad física (safety)**
  - adecuación al uso, facilidad de mantenimiento, amigabilidad de la interfaz de usuario, modularidad
  - seguridad de la información (security), facilidad de mantenimiento, facilidad de verificación, facilidad de adaptación
- 
- 3) Respecto a modelos de proceso de software:
- El esquema ETVX solo es adecuado para especificar modelos de proceso prescriptivos.
  - Una de las ventajas de un proceso en fases con liberaciones parciales en producción es que permite acortar el plazo para poder contar con resultados de utilidad, lo que a su vez permite atender la cada vez más buscada reducción del plazo para salir al mercado (time-to market).**
  - El proceso en cascada al día de hoy prácticamente ha dejado de ser utilizado en la industria, habiendo sido sustituido por otros modelos más eficientes como el modelo operacional.
  - Un proyecto se va a encarar siguiendo un proceso en fases con liberaciones parciales internas. Se ha definido que el proyecto se va a desarrollar en 3 fases. Podría resultar adecuado adoptar una estrategia de liberación paulatina para las 2 primeras fases y big-bang para la última.
- 
- 4) Dados los siguientes proyectos y modelos de proceso.
- Proyecto:
- A - de alto riesgo, muy complejo, alcance no del todo definido, larga duración (dos años o más).
  - B - Riesgo medio, alcance definido, subconjunto de funcionalidad definido y útil para los usuarios, duración media (ocho meses).
  - C - Riesgo bajo, simple, alcance definido, corta duración (dos meses).
- Proceso:
- 1 - Cascada
  - 2 - En Fases con Liberaciones Parciales en Producción
  - 3 - Espiral
- Elija la asignación que le parezca más adecuada.
- A3, B2, C1**
  - A1, B2, C3
  - A2, B1, C3
  - A1, B1, C2
-

- 5) Respecto a la liberación del software:
- Entre las principales ventajas de la estrategia big-bang en el caso de un sistema que sustituye a otro sistema anterior, están que evita los problemas que origina la coexistencia entre ambos sistemas y que se obtienen las ventajas del sistema nuevo desde el comienzo de la liberación.
  - (a) y la conversión de datos de un sistema anterior a uno nuevo que lo sustituye puede requerir un esfuerzo de orden comparable al proyecto de desarrollo del sistema nuevo.
  - (b) y al definir el alcance de un proyecto debe quedar claro si incluye o no tareas de la liberación del software tales como capacitación, entrenamiento y documentación para usuarios, instalación, soporte y solución de problemas, conversión de datos.
  - (c) y una explicación de que existan organizaciones que continúan soportando sus funciones críticas por un sistema legado (construido hace muchos años y en general con tecnologías hoy en día obsoletas), con costos muy elevados y crecientes de mantenimiento, es el importante esfuerzo que puede requerir la implantación de un sistema nuevo que lo sustituya.

- 6) Algunos de los problemas de mantenimiento de sistemas son:

- problemas del personal, por ejemplo: Comprensión limitada, cambios en las prioridades de la gerencia.
- Problemas Técnicos, por ejemplo: Especificaciones de diseño inadecuadas
- Necesidades conflictivas, por ejemplo: Solucionar lo antes posible un problema en un sistema que el personal de mantenimiento no domina
- costo: todos los problemas contribuyen a elevar los costos de mantenimiento

Seleccionar la opción correcta:

- Sólo A es correcta
- Sólo A y B son correctas
- Sólo A, B y C son correctas
- Todas son correctas

- 7) Respecto al mantenimiento de software:
- Es responsabilidad del Comité de Control de Cambios (CCC) controlar los cambios que se solicitan y aplican a un software que está en Producción.
  - (a) y el CCC debiera integrar la visión de Cliente, Usuarios y Desarrolladores para asegurar que el software evoluciona atendiendo a las necesidades de Cliente y Usuarios y los cambios propuestos son factibles y consistentes.
  - (b) y dejar registros del esfuerzo de mantenimiento correctivo por componente permite evaluar la conveniencia de re-escribir o re-diseñar un componente para mejorar su calidad y reducir los costos de mantenimiento.
  - (c) y prácticamente no existen herramientas que asistan a los desarrolladores en el mantenimiento de software.

- 8) Indique cuál de las siguientes afirmaciones sobre las técnicas de verificación basadas en el flujo de control del programa, es verdadera
- El "Criterio de Cubrimiento de Sentencias" cubre en general más caminos que el "Criterio de Cubrimiento de Decisión".
  - El "Criterio de Condición Múltiple" es considerado por Myers un criterio aceptable,
  - (b) porque asegura verificar todos los caminos posibles.
  - Las técnicas verificación basadas en el flujo de control del programa pertenecen a las técnicas de verificación de "Caja Negra".

- 9) En el proceso de Verificación y Validación las técnicas estáticas de verificación
- son efectivas en la detección temprana de defectos.
  - (a) y sirven para verificar no solo el código sino también los requerimientos y el diseño.
  - (b) y son muy utilizadas como técnicas de validación.
  - (b) y permiten tener en cuenta el ambiente donde será ejecutado el software.

10) Dado el siguiente segmento de código:

```
void pinturasInco(char color[20])
{
    bool pintar = false;
    while (pintar == false){
        if (color == "blanco"){
            color = "rojo";
        }
        else if (color == "negro"){
            color = "blanco";
        }
        else{
            color = "violeta";
            pintar = true;
        }
    }
}
```

el caso de prueba CP={color='negro'} cumple los siguientes criterios:

- 1) Cubrimiento de sentencias
- 2) Cubrimiento de decisión
- 3) Cubrimiento de condición
- 4) Cubrimiento de condición múltiple

Seleccione la opción correcta:

- a. Sólo 1 es correcta
- b. Sólo 1 y 2 son correctas
- c. Sólo 1, 2 y 3 son correctas
- d. **Todas las opciones son correctas**

11) Alguno/uno de los objetivos de la verificación de software son/es:

- a. Corregir defectos
- b. **Descubrir defectos**
- c. Identificar los defectos luego de provocada una falla
- d. Todas las anteriores

12) Varios estudios empíricos (Fagan, Ackerman, Jones, Basili) indican que:

- a. **Las inspecciones son muy efectivas, pudiendo detectar más del 60% de los defectos de un producto**
- b. Las inspecciones son costosas comparadas con las técnicas de verificación dinámica
- c. Las inspecciones no detectan defectos de funcionalidad en el software
- d. Todas las anteriores

13) Un idiom

- a. **Describe como implementar aspectos particulares de elementos o de las relaciones entre ellos usando las características de un lenguaje particular.**
- b. Describe una estructura recurrente de elementos de diseño interconectados que soluciona un problema general de diseño dentro de un contexto particular
- c. provee un esquema para refinar los elementos de un sistema de software o las relaciones entre ellos.
- d. expresa un esquema de organización estructural para sistemas de software.

14) La arquitectura de software está compuesta por:

- a. Los subsistemas que componen el sistema,
- b. (a) y las interfaces de los subsistemas,
- c. **(b) y las reglas de interacción entre los subsistemas,**
- d. (c) y los casos de uso del sistema.

15) En la etapa de diseño de software:

- a. Se refina la arquitectura
  - b. (a) y el diseño es correcto si satisface los requerimientos del sistema
  - c. (b) y se logra mantener alto acoplamiento entre módulos relacionados
  - d. (c) y siempre debe preservar la arquitectura
- 

16) El diseño:

- a. Es un "plano" de una solución para el sistema
  - b. El objetivo del diseño es encontrar el diseño perfecto
  - c. Facilita la reutilización siempre que tenga una baja cohesión
  - d. Ninguna de las anteriores
- 

17) Se entiende por configuración:

- a. Los elementos que componen toda la información producida como parte del proceso de ingeniería de software
  - b. (a), como ser datos y código
  - c. (b). Es recomendable que estos elementos sean versionados por herramientas automáticas ya que la complejidad de su gestión puede ser elevada
  - d. El código y documentos producidos como parte del proceso de ingeniería de software
- 

18) Son actividades de SCM:

- a. Controlar versiones
  - b. (a) e identificar elementos de configuración
  - c. Auditar el desarrollo
  - d. Todas las anteriores
- 

19) La definición de un proceso (según Watt Humphrey) consta de:

- a. El nombre del proceso, una lista de actividades y los roles involucrados
  - b. (a) y las entradas a las actividades con sus condiciones de entrada, las salidas de las actividades y sus condiciones de salida así como las mediciones involucradas
  - c. (a) y una estimación de cuánto tiempo lleva ejecutar el proceso
  - d. Todas las anteriores
- 

20) La estructura de áreas de proceso del modelo CMMI consta de:

- a. Metas específicas y genéricas que son requeridas y que se espera que sean logradas aplicando las prácticas correspondientes
  - b. Las mejores prácticas establecidas para el área que tienen que ser aplicadas en su totalidad
  - c. Una lista detallada de los objetivos del área y cómo lograrlo
  - d. Una guía informativa de las prácticas más comunes y los productos obtenidos como resultado de su aplicación
- 

21) El desempeño de un proceso es:

- a. Una medida de la madurez de la organización
  - b. La probabilidad de que los resultados obtenidos luego de su ejecución coincidan con los estimados
  - c. Una medida de los resultados reales obtenidos luego de su ejecución
  - d. Ninguna de las anteriores
- 

22) En cuanto a la herramienta de valor ganado:

- a. se usa para estimar el fin del proyecto pero si no existe una planificación previa con la cual comparar no sirve para nada.
  - b. se usa para hacer seguimiento y saber si el proyecto va atrasado o adelantado
  - c. no es suficiente para conocer si el proyecto va atrasado o adelantado, es preciso conocer cuál es el camino crítico.
  - d. Todas las anteriores
-

23) Al estimar esfuerzo total del proyecto por analogía con proyectos en la memoria organizacional de la empresa, se debería tomar en cuenta

- a. que tengan características similares (pe. peso de algoritmia, complejidad, interfaz gráfica )
  - b. que usen tecnologías similares, pero el lenguaje de desarrollo no importa
  - c. no importa el tamaño del proyecto, mientras la composición del equipo sea similar.
  - d. todas las anteriores
- 

24) Al planificar una iteración en un proceso de desarrollo con iteraciones:

- a. no es necesario calcular las horas que lleva cada actividad para saber si es factible
  - b. no es necesario realizar un perfil de uso de recursos
  - c. se debe hacer un perfil de uso de recursos individual para saber si el plan es factible
  - d. alcanza con hacer un perfil de uso de recursos grupal
- 

25) El estudio de factibilidad debe tomar en cuenta:

- a. la factibilidad técnica
  - b. el presupuesto asignado y los recursos disponibles
  - c. los plazos comprometidos, si es el caso
  - d. todas las anteriores
- 

26) Los requisitos no cambian:

- a. una vez que están estables
  - b. una vez que fueron subidos a línea base,
  - c. una vez negociados y validados con el cliente
  - d. ninguna de las anteriores
- 

27) En el documento de especificación de requisitos:

- a. se debe incluir solo lo que el cliente pidió
  - b. se deben incluir solo los requerimientos no funcionales
  - c. no deben ir las decisiones de diseño tomadas por el equipo de desarrollo
  - d. ninguna de las anteriores
- 

28) El procedimiento de control de cambios definido se debe aplicar:

- a. para cada cambio en los requisitos solicitado
  - b. para cada cambio solicitado, una vez que la especificación de los requisitos está en línea base
  - c. solo si se trata de un cambio de gran impacto
  - d. para cada detalle que se quiera cambiar al documento de especificación de requisitos.
- 

29) Al relevar requerimientos:

- a. se deben buscar antecedentes,
  - b. (a) y por antecedentes se entienden documentos existentes
  - c. (b) y sistemas análogos existentes en el mercado
  - d. (c) y por ejemplo leyes que determinen el funcionamiento del sistema
- 

30) Las entrevistas para relevar requerimientos:

- a. hay que prepararlas
- b. (a) y atenerse a los puntos de la agenda planificada
- c. (b) y hacerse no solo a los clientes, sino también a los usuarios
- d. deben utilizarse siempre por ser la mejor técnica de relevamiento.