

- Cada pregunta de la **parte múltiple opción** contestada correctamente tiene un valor de 3,2 puntos. Cada pregunta incorrecta de la múltiple opción resta 1,066 puntos. Esta parte consta de 25 preguntas por lo que vale en total 80 puntos.
- El ejercicio 1 y el ejercicio 2 valen 10 puntos cada uno.
- Se aprueba con el 70% del puntaje.

Parte 1 Múltiple Opción

- 1) Entre las cualidades más relevantes para el software que gestiona el acceso a Internet de las XO del Plan Ceibal están:
 - a) visibilidad del proceso de construcción, resguardo de la integridad física (safety), facilidad de mantenimiento, robustez
 - b) facilidad de mantenimiento, modularidad, facilidad para ser comprendido, atractivo para el usuario
 - c) **seguridad de la información (security), robustez, tiempos de respuesta, uso de recursos**
 - d) amigabilidad de la interfaz de usuario, adecuación al uso, tiempos de respuesta, atractivo para el usuario.

- 2) Respecto a modelos de proceso:
 - a) Una de las principales desventajas de un proceso en fases con liberaciones intermedias en explotación, consiste en que no permite contar tempranamente con resultados útiles.
 - b) Una de las principales ventajas del modelo en cascada es su complejidad.
 - c) El modelo V se caracteriza por incorporar en el modelado de procesos el factor "riesgo".
 - d) **El proceso en cascada resulta más adecuado para proyectos de pequeño tamaño, corta duración y bajo riesgo que para proyectos de gran tamaño, larga duración y alto riesgo.**

- 3)
 - a) Conviene que la planificación de las actividades relacionadas con la liberación (puesta en explotación) de un producto comience una vez que se cuenta con la Especificación de Requerimientos en la línea base bajo control de cambios.
 - b) **La liberación (puesta en explotación) de un producto nuevo, en general involucra las actividades siguientes:**
 - Preparar la documentación de usuario
 - Entrenar/Capacitar a los usuarios
 - Eventualmente realizar migraciones de datos de un sistema anterior al nuevo
 - Instalar las versiones ejecutables requeridas.
 - c) La migración (o conversión) de datos de un sistema anterior a uno nuevo que lo sustituye en general es una tarea sumamente sencilla y que conviene encarar una vez que se dispone al menos de una versión beta del sistema nuevo.
 - d) La mejor estrategia de liberación es la paulatina.

- 4) Respecto al mantenimiento del sistema:
 - a) La mayoría de los sistemas legados de las organizaciones son poco sensibles a los cambios de contexto en la organización ya que los lenguajes estructurados (COBOL, CLIPPER, entre otros) facilitan el mantenimiento.
 - b) **Una arquitectura pensada para el futuro mantenimiento de un sistema podría ser determinante para que los atributos de calidad del software no se degradaran rápidamente con el posterior mantenimiento.**
 - c) El costo del mantenimiento en un proyecto disminuye con el tiempo, ya que el software se torna menos útil y por tanto no se realizan tantos pedidos de cambio.
 - d) El mantenimiento de software siempre requiere un equipo más especializado en el negocio de la organización que en el conocimiento de las tecnologías o la arquitectura del software.

- 5) Dentro del proceso de Requisitos:
 - a) Se encuentran las actividades de obtención, análisis y especificación de requisitos.
 - b) a) y uno de los artefactos construidos es el Documento de Visión.
 - c) **b) y pueden participan de este proceso diseñadores y verificadores**
 - d) c) y no participa de este proceso el cliente.

-
- 6) La técnica de obtención de requisitos:
- Investigar antecedentes - tiene como ventaja una perspectiva ilimitada.
 - Entrevista individual/Grupal - resulta costosa pero rica y orientada a las personas.**
 - Encuesta - es útil para validar asunciones pero no permite obtener datos estadísticos sobre preferencias del cliente.
 - Tormenta de ideas - no permite dejar volar la imaginación, sólo debatir y criticar.
-
- 7) Con respecto a los casos de uso:
- Es una técnica para entender y describir requisitos.
 - a) y describen como el sistema debe comportarse desde el punto de vista del usuario.**
 - a) y son de gran ayuda para identificar aspectos no funcionales.
 - Todas las anteriores.
-
- 8) En cuanto a las características de una buena especificación SRS (IEEE), los requisitos
- Son Correctos, o sea que sean validados por el cliente.
 - a) y No Ambiguos, todo requisito tiene una única interpretación.
 - b) y Verificables, o sea que existe un proceso finito de costo accesible para determinar que el sistema lo cumple.**
 - Ninguna de las anteriores.
-
- 9) De los Modelos o Vistas del Sistema se puede afirmar que:
- Las Tablas de decisión, las Redes de Petri y los Diagramas de Flujo de Datos son descripciones dinámicas.
 - Las Redes de Petri permiten describir concurrencia y sincronización
 - Los Diagramas de Flujo de Datos permiten visualizar cómo fluye la información por el sistema.
 - Todas las anteriores**
-
- 10) El PERT no resulta una herramienta muy adecuada para la Ingeniería de Software porque:
- Las duraciones estimadas en Ing. SW son generalmente fijas
 - Las duraciones no son variables independientes, como en otras disciplinas**
 - La aleatoriedad de las duraciones hace que, aunque se atrasen tareas que no están en el camino crítico, éstas nunca van a pasar a formar parte del mismo
 - El camino crítico no siempre tiene mayor duración que otros caminos no críticos
-
- 11) Indique las dificultades que pueden existir para estimar el esfuerzo en forma precisa:
- No se tiene historia de la empresa
 - Entornos, tecnología o lenguaje desconocidos
 - Definición de requerimientos (usada para estimar) imprecisa
 - Todas las anteriores**
-
- 12) El Plan de Proyecto incluye:
- El alcance
 - El plan de gestión de Recursos Humanos
 - El cronograma
 - Todas las anteriores**
-
- 13) En un grafo de actividades:
- No se consideran las relaciones de precedencia entre las actividades y no se permiten circuitos de precedencia.
 - El fin tardío de una actividad es lo más tarde que puede terminar una actividad, afectando la duración total del proyecto.
 - La holgura total de una actividad indica cuánto se puede atrasar su comienzo sin afectar la duración total del proyecto.**
 - Es un modelo simplificador, que permite marcar el comienzo de una actividad en un punto determinado de la ejecución de otra.
-

-
- 14) Sobre los estilos de arquitectura de software
- Capas jerárquicas, en su modelo estricto, debe proveer protocolos que establezcan como interactuará cada par de capas. Los cambios en las interfaces solo afectan a un par de capas. No siempre es fácil estructurar las capas.
 - La descomposición orientada a objetos promueve la reutilización de componentes y los cambios en las interfaces afectan a todos los que la usan.
 - Los sistemas distribuidos son altamente escalables, tolerantes a fallas pero son complejos y difíciles de gestionar.
 - Todas las anteriores**
-
- 15) Relacionado a los principios de diseño
- “Dividir y conquistar” consiste en dividir en piezas simples un problema complejo con el principal objetivo de poder delegar estas piezas a distintos programadores.
 - “Abstracción” implica describir el comportamiento exterior de un componente, evitando entrar en los detalles internos.**
 - La “Modularidad” se observa en sistemas que pueden implementarse separadamente, pero esta fuertemente ligada a lenguajes de programación que permitan aplicar este principio.
 - Tanto (a) como (b) son correctas
-
- 16) En Arquitecturas de Sistemas Distribuidos
- El procesamiento de la información es distribuido entre varias computadoras.
 - a) y una de sus ventajas es que se comparten recursos y una desventaja es la dificultad que implica gestionarlas.**
 - Una de sus ventajas es que promueve la reutilización de componentes y una desventaja es su complejidad.
 - todas las anteriores y un ejemplo es la arquitectura cliente servidor.
-
- 17) Los aspectos clave para el diseño de la interfaz de usuario son:
- La forma de interacción entre el usuario y el sistema
 - La forma en la cual se despliegan los mensajes de error
 - La forma de cada botón de interacción
 - La forma de presentar los datos al usuario
- Se cumplen solamente I y III
 - Se cumplen solamente I, II y IV
 - Se cumplen solamente II y III
 - Se cumplen solamente I y IV**
-
- 18) Respecto de la verificación:
- Uno de los objetivos principales de la verificación es descubrir defectos.
 - a) y evaluar la calidad de los productos.**
 - b) y para que un defecto sea considerado como tal, éste debe haber producido una falla.
 - c) y para descubrir defectos se pueden realizar pruebas y revisiones de código.
-
- 19) Respecto de las pruebas de software
- El enfoque de pruebas de caja blanca tiene la debilidad de que si se quiere utilizar un criterio fuerte (trayectorias linealmente independientes, todos los usos, etc.), diseñar los casos de prueba es muy costoso.
 - El enfoque de pruebas de caja negra tiene la desventaja de que si los requerimientos del sistema no son completos no se puede realizar.
 - a) y existen herramientas para ejecutar casos de prueba.**
 - b) y para diseñar pruebas de caja negra no es necesario conocer la estructura interna del código.
-
- 20) ¿Quién Verifica?
- El encargado de realizar las pruebas unitarias, de integración y del sistema es razonable que sea el propio equipo de desarrollo.
 - a) y el resto de las pruebas en general un equipo especializado
 - El encargado de realizar las pruebas unitarias y de integración es razonable que sea el propio equipo de desarrollo.
 - c) y el resto de las pruebas en general las realiza un equipo especializado porque entre otras cosas el autor de un programa tiende a cometer los mismos errores al probarlo**
-

21)

- a) Un caso de prueba se compone de entradas, condiciones de ejecución y el resultado esperado.
 - b) a) y con las técnicas de caja blanca se puede pasar por alto un requerimiento no implementado.
 - c) b) y luego de someter al sistema a pruebas de caja negra complementadas con pruebas de caja blanca, con casos bien diseñados, completos, y con valores de entrada estratégicamente seleccionados es posible afirmar la ausencia de defectos
 - d) Ninguna de las anteriores
-

22) Seleccione la que no es una estrategia de integración

- a) Bottom-up
 - b) Top-down
 - c) Sándwich
 - d) Deductiva
-

23) Entre las actividades de la Gestión de la Configuración se encuentran:

- a) Identificar elementos que se pondrán bajo la gestión de configuración
 - b) a), gestionar el cambio de estos elementos
 - c) a) y ubicarlos bajo la línea base de forma de minimizar los cambios
 - d) b) y realizar auditorías de seguimiento que permitan evaluar la aplicación de los procedimientos de gestión
-

24) Sobre el análisis post-mortem:

- a) Se ejecuta una vez terminado el proyecto, y como resultado se elabora un informe que debe ser entregado el cliente para mejorar su nivel de satisfacción.
 - b) Busca mejorar la organización, indagando cuáles fueron los puntos fuertes y las oportunidades de mejora en un proyecto
 - c) Evalúa productos y procesos, pero se excluyen los recursos humanos ya que no busca encontrar culpables.
 - d) Tanto (b) como (c) son correctas
-

25) En la representación continua del modelo de mejora de procesos CMMI:

- a) Cada nivel estructura el proceso de mejora como un conjunto de áreas de proceso relacionadas
- b) a) y por lo menos se requieren las correspondientes a las categorías Ingeniería y Gestión de Proyectos
- c) Cada nivel estructura el proceso de mejora como una mejora incremental de los procesos correspondientes a una o más áreas de proceso
- d) c) y por lo menos se requieren las correspondientes a las categorías Ingeniería, Gestión de Proyectos y Gestión de Procesos

Ejercicio 1 (10 puntos)

Un método recibe dos circunferencias. Cada circunferencia se representa con la posición del centro en un plano (x, y) y el radio. El método tiene como precondition que ambas circunferencias tengan una posición válida y un radio válido (mayor que cero).

El método retorna como resultado la distancia entre los centros de las circunferencias e indica con un entero cuántas intersecciones existen entre las circunferencias.

```
public resultadoCircunferencias ejercicioIIS (posx1, posy1, radio1, posx2, posy2, radio2)
```

Indicar qué tipos de casos de prueba son interesantes para probar el método ejercicioIIS.

Como máximo se podrán indicar 8 tipos de casos de prueba. Seleccione los más adecuados para el problema.

Ejercicio 2 (10 puntos)

Se desea automatizar un garaje de 5 plazas de forma tal de que si éste se encuentra lleno se encienda una luz indicándolo y no suba la barrera. En caso contrario deberá estar encendida otra luz indicando libre.

Hay una célula fotoeléctrica y una barrera a la entrada y lo mismo a la salida.

- a) Modele el sistema con un diagrama de estados
- b) ¿Qué problema le encuentra a este modelo?
- c) ¿Con qué otra herramienta podría superarse este problema y modelarlo mejor?