

Responda V (Verdadero) o F (Falso). Con tinta, no se puede utilizar lápiz.

1. Las restricciones en el presupuesto o calendario NO son el origen del cambio en la Gestión de Configuración del Sistema (GCS).
2. Un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único.
3. El equipo de software ágil se parece a la organización del paradigma abierto.
4. En la planeación de riesgos la estrategia denominada "plan de contingencia" establece que, siguiendo esta estrategia, la probabilidad que el riesgo se presente se reduce.
5. En la planeación temporal de un proyecto un "hito" es algo que se espera que esté hecho para alguna fecha, un logro que sea objetivo, fácil de evaluar y notable.
6. Del gráfico e PERT-CPM se puede obtener una ventana temporal para el desarrollo de cada actividad.
7. En el diseño de software se busca que los módulos tengan mucho acoplamiento y baja cohesión.
8. El refactoring de software es una técnica de reorganización que simplifica el diseño de un componente sin cambiar su función o comportamiento.
9. Los colores y el diseño visual son características centrales de la experiencia de usuario (UX).
10. El principio de consistencia de Jacob Nielsen en el desarrollo de interfaces indica que no debe haber ambigüedades en la terminología y lo visual.
11. El modelo de control centralizado para el diseño de una arquitectura de software se caracteriza por responder a eventos externos al subsistema.
12. Una de las desventajas de la arquitectura de sistemas distribuidos es la seguridad.
13. En la arquitectura distribuida interorganizacional cada nodo de procesamiento maneja su propia carga de trabajo, pero la carga informática general se comparte dinámicamente entre todos los nodos.
14. En la arquitectura Cliente-Servidor, los clientes y servidores son procesos diferentes. Además, los servidores pueden atender varios clientes donde un servidor puede brindar varios servicios y los clientes no se conocen entre sí.
15. La métrica LCD (líneas de código) es una métrica post mortem porque se calcula con datos de otros proyectos de software ya finalizados.
16. Si se realizan pruebas de software se garantiza la ausencia de defectos.
17. En las estrategias de prueba la verificación responde a la pregunta de ¿Estamos construyendo el producto correcto? de manera de asegurar que el software satisface las expectativas del cliente.
18. El mantenimiento de software es una actividad que incluye: corregir errores, mejorar las capacidades, eliminar funciones obsoletas y optimizar otras.
19. En las leyes de Lehman la "complejidad creciente" hace referencia a que la evolución de los programas es un proceso autorregulativo donde los atributos (tamaño, número de errores documentados, etc) son aproximadamente invariantes en el tiempo.
20. En el rejuvenecimiento del software la "re-documentación" representa un análisis estático del código para producir la documentación del sistema.

Responder en cada recuadro con tinta, no se puede utilizar lápiz.

**1. Indique 5 consejos para una buena comunicación del proyecto en formato presentación**

**2. Describa 4 pautas de criterios técnicos para un buen diseño de software.**

**3. Indique y describa 3 principios aplicados al auditor informático**