

Ingeniería del Software

Práctico N° 1 - El Producto y el Proceso (Capítulo 1 y 2 de Pressman)

Capítulo 1 - El Producto

1. ¿Qué factor ha precipitado la aparición de sistemas basados en computadoras más sofisticados y complejos?
 - ☐ El vasto incremento en la capacidad de almacenamiento y memoria de las computadoras.
 - ☐ Mayor variedad de opciones exóticas de entrada/salida.
 - ☐ Cambios profundos en las arquitecturas de las computadoras.
 - ☐ Todos los de arriba.
2. ¿Qué pregunta no resulta más de interés para los ingenieros de software modernos?
 - ☐ ¿Por qué el hardware de computadoras cuesta tanto?.
 - ☐ ¿Por qué el software toma un tiempo tan largo para ser finalizado?.
 - ☐ ¿Por qué cuesta tanto desarrollar una pieza de software?.
 - ☐ ¿Por qué los errores no pueden ser removidos de los productos a priori a ser entregados?.
3. Hoy en día la creciente potencia de las computadoras personales ha producido un abandono de la práctica de desarrollo en equipo de software.
 - ☐ Verdadero
 - ☐ Falso
4. El Software es un producto y puede manufacturarse usando la misma tecnología usada para otros artefactos de la ingeniería.
 - ☐ Verdadero
 - ☐ Falso
5. Decimos que el Software se deteriora más bien que se desgasta porque
 - ☐ El software sufre de la exposición a ambientes hostiles.
 - ☐ Los defectos son más probables de surgir después que el software ha sido usado a menudo.
 - ☐ Múltiples cambios de requerimientos introducen errores in la interacción de las componentes.
 - ☐ Las piezas de recambio del software son muy difíciles de adquirir en el mercado del software.
6. La mayoría del software continúa siendo personalizado porque
 - ☐ El reuso de componentes es común en el mundo del software.
 - ☐ Las componentes reusables son demasiado caras para usarlas.
 - ☐ El software es más fácil de construir sin usar las componentes de otro.
 - ☐ Las componentes de software no están comúnmente disponibles en el mercado.
7. La naturaleza de las aplicaciones de software pueden caracterizarse por
 - ☐ La complejidad de su información.

- ☐ El contenido de su información.
 - ☐ El determinismo de la información.
 - ☐ Las opciones "b" y "c".
8. Las aplicaciones de software modernas son tan complejas que es difícil desarrollar nombres de categorías mutuamente excluyentes.
- ☐ Verdadero
 - ☐ Falso
9. La crisis del software actual fue causada por el problema Y2K cuyas semillas fueron sembradas primero por programadores descuidados de principios de los '70.
- ☐ Verdadero
 - ☐ Falso
10. Los desarrolladores de software han tenido más éxitos que fracasos, pero las fallas del software recibieron mayor atención por parte de la prensa.
- ☐ Verdadero
 - ☐ Falso
11. Agregar más personas a un proyecto que ya está atrasado en su calendario es una buena manera de ponerse al día.
- ☐ Verdadero
 - ☐ Falso
12. Contar con herramientas de CASE modernas es más importante que contar con el hardware más nuevo para lograr una buena calidad y productividad del software.
- ☐ Verdadero
 - ☐ Falso
13. Los cambios no pueden ser fácilmente acomodados en la mayoría de los sistemas de software, a menos que el sistema haya sido diseñado previendo la posibilidad de cambios.
- ☐ Verdadero
 - ☐ Falso
14. Una declaración general de objetivos es todo lo que se necesita para empezar a desarrollar una pieza de software.
- ☐ Verdadero
 - ☐ Falso
15. Más allá de la naturaleza de los defectos del software, las revisiones técnicas formales son un sustituto inadecuado para la prueba de programas (testing).
- ☐ Verdadero
 - ☐ Falso
16. La Documentación no es más una parte necesaria del proceso de desarrollo del software porque nadie la lee.
- ☐ Verdadero
 - ☐ Falso

Capítulo 2 - El Proceso

1. ¿Cuál de los items listados abajo no es una capa de la ingeniería del software?
 - ☐ Proceso
 - ☐ Manufacturación
 - ☐ Métodos
 - ☐ Herramientas
2. ¿Cuáles son las tres fases genéricas de la ingeniería del software?
 - ☐ Definición, Desarrollo, Mantenimiento
 - ☐ Qué, Cómo, Dónde
 - ☐ Programación, Depuración, Mantenimiento
 - ☐ Análisis, Diseño, Prueba
3. ¿Cuál de estos términos es el nombre de un nivel en el Modelo de Madurez de Capacidad?
 - ☐ Ad hoc
 - ☐ Repetible
 - ☐ Reusable
 - ☐ Organizado
4. ¿Cuál de estos items debería ser tenido en cuenta para elegir un marco de proceso de desarrollo de software?
 - ☐ Personas
 - ☐ Producto
 - ☐ Proyecto
 - ☐ Todos los de arriba.
5. El modelo lineal secuencial de desarrollo de software es
 - ☐ Una propuesta razonable cuando los requerimientos están bien definidos.
 - ☐ Una buena propuesta cuando se requiere rápidamente un programa funcionando.
 - ☐ La mejor propuesta para usar en proyectos con equipos de desarrollo grandes.
 - ☐ Un modelo pasado de moda que ya raramente se utiliza.
6. El modelo lineal secuencial de desarrollo de software también es conocido como
 - ☐ Modelo del ciclo de vida clásico
 - ☐ Modelo de la Fuente
 - ☐ Modelo Espiral
 - ☐ Modelo de Caos
7. El modelo de prototipado de desarrollo de software es
 - ☐ Una propuesta razonable cuando los requerimientos están bien definidos.
 - ☐ Una propuesta útil cuando el cliente no puede definir claramente los requerimientos.
 - ☐ La mejor propuesta para usar en proyectos con equipos de desarrollo grandes.
 - ☐ Un modelo de riesgo que raramente produce un producto significativo.

8. El modelo de desarrollo rápido de aplicación es
- ☐ Otro nombre para el desarrollo basado en componentes.
 - ☐ Una propuesta útil cuando el cliente no puede definir claramente los requerimientos.
 - ☐ Una adaptación de alta velocidad del modelo lineal secuencial.
 - ☐ Todos los de arriba.
9. Los modelos de proceso del software evolutivos
- ☐ Son iterativos por naturaleza.
 - ☐ Pueden acomodarse fácilmente a los cambios de requerimientos del producto.
 - ☐ Generalmente no producen sistemas que son descartados.
 - ☐ Todos los de arriba.
10. El modelo incremental de desarrollo de software es
- ☐ Una propuesta razonable cuando los requerimientos están bien definidos.
 - ☐ Una buena propuesta cuando un producto operativo con funcionalidad básica es requerido rápidamente.
 - ☐ La mejor propuesta para usar en proyectos con equipos de desarrollo grandes.
 - ☐ Un modelo revolucionario que no es usado para productos comerciales.
11. El modelo en espiral de desarrollo de software
- ☐ Termina con la entrega del producto de software.
 - ☐ Es más caótico que el modelo incremental.
 - ☐ Incluye evaluación de riesgos del proyecto durante cada iteración.
 - ☐ Todos los de arriba.
12. El modelo de desarrollo basado en las componentes es
- ☐ Apropiado solo para el diseño de hardware de computadora.
 - ☐ No puede soportar el desarrollo de componentes reusables.
 - ☐ Depende de la tecnología de objetos para soportarlo.
 - ☐ No rentable de acuerdo a resultados obtenidos por métricas de software cuantificables conocidas.
13. El modelo de métodos formales de desarrollo de software hace uso de métodos formales para
- ☐ Definir la especificación para sistemas basados en computadoras.
 - ☐ Desarrollar sistemas basados en computadoras libres de defectos.
 - ☐ Verificar la correctitud de sistemas basados en computadoras.
 - ☐ Todos los de arriba.
14. Las técnicas de cuarta generación
- ☐ Permiten al software ser desarrollado sin pruebas.
 - ☐ Eliminan la necesidad de costosas actividades de recolección de requerimientos.
 - ☐ Pueden reducir el tiempo requerido para desarrollar software.
 - ☐ Son mejor usadas para construir sistemas pequeños por personas sin experiencias en programación.