

Evaluación Final

Parte teórica

Ej. 1. Además de la calidad, la Ingeniería del Software debe confrontarse con otros desafíos. Enumérelos y explíquelos brevemente.

Ej. 2.

- (a) Describa el enfoque "ETVX" utilizado en diversas etapas de los distintos procesos.
- (b) Enumere los modelos de procesos de desarrollo que conoce y compárelos teniendo en cuenta las ventajas, desventajas y contextos de aplicación.

Ej. 3. ¿Qué características debe tener una especificación de requerimientos? Describa tres de ellas. Elija tres de las características restantes y explique qué problemas ocurrirían si la especificación no las poseyera.

Ej. 4.

- (a) ¿Qué describen las vistas de módulo y la de asignación de recursos?
- (b) Enumere los estilos arquitectónicos que conoce para la vista de componentes y conectores. Elija tres de ellos y describa para qué tipos de sistema son más adecuados cada uno.

Ej. 5. Supongamos que la estimación del esfuerzo de un proyecto resulta en 36 personas por mes. Sabemos que incrementar arbitrariamente el personal no llevará a un incremento lineal en la productividad. ¿Cuál es el máximo sugerido de personas involucradas en dicho proyecto? ¿Qué valores dictan las distintas normas que conoce?

Ej. 6.

- (a) ¿Por qué es deseable tener bajo acoplamiento?
- (b) ¿Cuáles son los factores más importantes que influyen en el acoplamiento y qué deben cumplir para lograr un diseño con bajo acoplamiento?

Ej. 7.

- (a) ¿Cuáles son los tres tipos de cohesión que se distinguen en diseño orientado a objetos? Describa muy brevemente en qué se enfocan cada uno de ellos.
- (b) Explique qué establece el principio abierto-cerrado.

Ej. 8.

- (a) Describa el modelo de proceso de codificación dirigido por test. ¿Qué problemas se deben tener en cuenta en este tipo de desarrollo?
- (b) Usted se encuentra en una reunión para la revisión del código: ¿en qué cosas deberá enfocar su atención con el fin de tratar de mejorar la calidad del código? (Enumere al menos 6).

Ej. 9.

- Defina testing de caja blanca y de caja negra.
- Dé un pequeño ejemplo de programa erróneo que pase un test por particionado por clases de equivalencias y *no* pase un test por análisis de valores límites. Justifique.
- Dé un pequeño ejemplo de programa erróneo para el cual haya un test que cumpla el criterio de cobertura de nodos que detecte el error y un test que cumpla el criterio de cobertura de bifurcación que *no* lo detecte. Justifique.

Parte práctica

Ej. 11. Considere el siguiente problema, y produzca un diagrama de clases correspondiente al análisis orientado a objetos de una solución para el mismo:

Se desea producir el software necesario para el manejo de una red bancaria compuesta de un conjunto de cajeros automáticos, que serán compartidos por un consorcio de bancos. Cada banco dispone de su propio servidor provisto de software propio, que lleva la información sobre sus cuentas y procesa las transacciones que actúan sobre dichas cuentas. A este servidor están conectadas las estaciones de cajero, que son propiedad del banco y en las que operan cajeros humanos, que pueden crear cuentas e introducir transacciones sobre ellas.

Los cajeros automáticos aceptan tarjetas de crédito, interactúan con el usuario, se comunican con un servidor central para llevar a cabo las transacciones, entregan dinero en efectivo al usuario e imprimen recibos. El sistema deberá llevar correctamente el registro de las transacciones efectuadas, cumplir características aceptables de seguridad y manejar correctamente accesos concurrentes a la misma cuenta.

Ej. 12. Sobre el problema del ejercicio anterior, describa al menos tres escenarios de uso (sin descuidar casos excepcionales) mediante casos de uso.

Ej. 13. Considere el siguiente problema y produzca un DFD correspondiente al diseño estructurado para la funcionalidad asociada a la detección de un suceso de sensor:

Se desea producir el software para el manejo de un sistema de seguridad hogareño. El software debe permitir al usuario configurar el sistema de seguridad, controlar todos los sensores conectados al sistema de seguridad, e interactuar con el usuario a través de un teclado numérico y teclas de función contenidas en el panel de control del sistema.

Durante la instalación, el panel de control del sistema de seguridad se usa para programar y configurar el sistema. A cada sensor se le asigna un número y tipo, se programa una contraseña maestra para activar y desactivar el sistema, y se introducen números de teléfono a marcar cuando un sensor detecte un suceso.

Cuando se reconoce un suceso de sensor, el software invoca una alarma audible asociada al sistema. Después de un tiempo de espera especificado por el propietario durante las actividades de configuración del sistema, el software marca un número de teléfono de un servicio de control, proporciona información acerca de la localización, e informa de la naturaleza del suceso detectado. El número será remarcado cada 20 segundos hasta obtener una conexión telefónica.