

INGENIERÍA TÉCNICA en INFORMÁTICA de SISTEMAS y de GESTIÓN
CÓDIGO CARRERA: 40=SISTEMAS y 41=GESTIÓN
CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

ASIGNATURA: INGENIERÍA DEL SOFTWARE

TIPO DE EXAMEN: A

FECHA: 24 de mayo de 2000

Hora: 11:30

Duración: 2 horas

MATERIAL: NINGUNO

¡ATENCIÓN! PONGA EL TIPO DE EXAMEN EN LA HOJA DE LECTURA ÓPTICA.

Conteste al test en la hoja de lectura óptica.

Sólo hay una respuesta correcta en cada pregunta.

PREGUNTAS DE TEST: El test es **ELIMINATORIO** (son necesarias 7 respuestas correctas para pasarlo),
(PUNTOS: 4)

- 1.- La "crisis del software" consiste en:
 - A.- La saturación de productos en el mercado.
 - B.- La falta de profesionales capacitados.
 - C.- Que la capacidad del hardware obliga al desarrollo de aplicaciones muy complejas.
 - D.- La gran cantidad de defectos del software.
- 2.- ¿Qué actividad del Ciclo de Vida está orientada a controlar la calidad?
 - A.- El análisis.
 - B.- Las revisiones y pruebas.
 - C.- La codificación.
 - D.- El diseño.
- 3.- ¿Cuál de las siguiente NO es una ventaja del "Análisis del Dominio"?
 - A.- Reutilización del software.
 - B.- Identificación de los elementos más relevantes.
 - C.- Facilita la comunicación entre analista y usuario.
 - D.- Que obtiene una solución particularizada, "a la medida" del problema.
- 4.- El proceso de la abstracción:
 - A.- Permite definir un módulo a partir de la "esencia" de un conjunto de elementos.
 - B.- Se usa para refinar el modelo desarrollado con DFD.
 - C.- Establece un conjunto de pruebas de "caja negra".
 - D.- Dota al diseño de una estructura jerárquica a través de la herencia.
- 5.- Con la genericidad se pretende:
 - A.- Diseñar cada elemento específico como un caso particular del genérico.
 - B.- Hacer invisible a otros elementos los que es irrelevante para su uso o puede ser cambiado en el futuro.
 - C.- Que los elementos específicos hereden las propiedades del genérico.
 - D.- Sincronizar las tareas particulares de un módulo genérico.
- 6.- La diferencia entre el acoplamiento Común y de Datos es:
 - A.- En el Común, las estructuras compartidas residen en un elemento externo al sistema.
 - B.- En el de Datos sólo se comparten las estructuras estrictamente necesarias para su uso.
 - C.- En el Común, todos los elementos de un módulo utilizan los mismos datos de entrada y de salida.
 - D.- En el Común, se agrupan elementos con funciones similares desde el punto de vista del usuario.
- 7.- Si una base de datos se organiza en tres niveles: externo, conceptual e interno; entonces:
 - A.- El interno emplea esquemas de usuario para organizar los datos.
 - B.- El conceptual emplea esquemas significativos desde la visión del usuario.
 - C.- El conceptual establece una organización lógica de los datos.
 - D.- Se emplea el Diseño para pasar de los esquemas de usuario al nivel conceptual.
- 8.- Los dos esquemas empleados en la recuperación de errores son:
 - A.- Detección y recuperación.
 - B.- La prevención y el ignorarlos.
 - C.- Semáforos y monitores.
 - D.- Hacia delante y hacia atrás.
- 9.- Un entorno SmallTalk:
 - A.- Es un entorno de la 4ª generación.
 - B.- Es un entorno asociado a la metodología de objetos.
 - C.- Es un entorno asociado a un lenguaje.
 - D.- Es un entorno basado en herramientas.
- 10.- En los entornos integrados, la integración de los datos se puede conseguir mediante:
 - A.- Transferencia basada en la comunicación.
 - B.- Semáforos.
 - C.- Sucesos o eventos.
 - D.- Una restricción del desarrollo.

EJERCICIO (6 puntos)

Nota: No se corregirá la contestación que exceda las dimensiones de tres **caras** del tamaño A4-ISO (210x297 mm)

Se quiere construir una aplicación que permita a un usuario jugar al Nim contra la máquina. El juego comienza con un montón de n cerillas. Se juega por turnos entre dos contrincantes. En la primera jugada, se pueden retirar tantas cerillas como se desee, pero hay que dejar al menos una en el montón. En el resto de los turnos, hay que tomar al menos una cerilla y, como máximo, el doble de las que ha tomado el contrincante en el turno anterior. Gana el que toma la última cerilla del montón. El programa debe permitir seleccionar el número de cerillas inicial, el que toma el usuario en cada turno y representar en la pantalla en cada instante el turno, el número de cerillas que restan en el montón y el que han retirado los contrincantes en la jugada anterior. Además, debe calcular al principio del juego si existe una estrategia de juego óptima para el número de cerillas inicial seleccionado. Dicha estrategia la seguirá durante el juego retirando las cerillas que corresponda en cada turno. Si no existe una estrategia óptima, la máquina jugará para ganar en el menor número de turnos posible.

Haga una lista ordenando por tipos los requisitos del programa. Si es necesario, complete dichos requisitos.

3 puntos

Construya un modelo del sistema y represéntelo con la notación DFD. El nivel de detalle al que deberá llegar con la representación tiene que dejar claro qué hace el sistema.

3 puntos

INGENIERÍA TÉCNICA en INFORMÁTICA de SISTEMAS y de GESTIÓN
CÓDIGO CARRERA: 40=SISTEMAS y 41=GESTIÓN
CÓDIGO DE ASIGNATURA: 210=SISTEMAS y 208=GESTIÓN

ASIGNATURA: INGENIERÍA DEL SOFTWARE

TIPO DE EXAMEN: E

FECHA: 7 de junio de 2000

Hora: 11:30

Duración: 2 horas

MATERIAL: NINGUNO

¡ATENCIÓN! PONGA EL TIPO DE EXAMEN EN LA HOJA DE LECTURA ÓPTICA.

Conteste al test en la hoja de lectura óptica.

Sólo hay una respuesta correcta en cada pregunta.

PREGUNTAS DE TEST: El test es **ELIMINATORIO** (son necesarias 7 respuestas correctas para pasarlo),
(PUNTOS: 4)

- 1.- El Ciclo de Vida del software es:
 - A.- Un patrón del comportamiento de los diseñadores.
 - B.- Un modelo del proceso de construcción del Software.
 - C.- Un modelo de especificaciones establecido por el analista.
 - D.- Un esquema para integrar el sistema completo.
- 2.- El objetivo principal del análisis es:
 - A.- Determinar el mecanismo de funcionamiento de los módulos.
 - B.- Validar con el cliente el cumplimiento de las especificaciones.
 - C.- Establecer claramente las especificaciones del producto.
 - D.- Integrar los distintos componentes del sistema.
- 3.- El requisito del juego de las minas "Al destapar una casilla son minas alrededor, se destapan automáticamente las casillas colindantes" es del tipo:
 - A.- De operación.
 - B.- De verificación.
 - C.- De mantenimiento.
 - D.- Funcional.
- 4.- Las ventajas de un diseño modular son:
 - A.- Depuración, mantenimiento y sincronización.
 - B.- Facilitar las revisiones, las pruebas del sistema y la validación.
 - C.- Accesibilidad, previsión y consistencia.
 - D.- División del trabajo, claridad, reducción de costes y reutilización.
- 5.- Los diagramas de abstracciones son una notación:
 - A.- Basada en los datos.
 - B.- Con cohesión baja.
 - C.- Híbrida.
 - D.- Dotada del mecanismo de la herencia.
- 6.- La Programación Estructurada de Jackson es una técnica de diseño:
 - A.- Estructurado.
- B.- De descomposición funcional.
- C.- Orientada a objetos.
- D.- Para organizar los datos.
- 7.- Si se desea codificar un componente embebido de software, con especificaciones críticas de tiempo real, ¿qué tipo de lenguaje suele resultar más adecuado?
 - A.- Ensamblador o de bajo nivel.
 - B.- Uno de la tercera generación.
 - C.- LISP.
 - D.- Uno de tipado fuerte.
- 8.- Las estrategias principales para probar unidades son:
 - A.- Pruebas de caja negra y de caja transparente.
 - B.- Partición en clases equivalentes y análisis de valores límite.
 - C.- Cubrimiento lógico y pruebas de bucles.
 - D.- Intuición y comparación de versiones.
- 9.- El repositorio es un almacén común de información...
 - A.- Para construir entornos y herramientas según una metodología determinada.
 - B.- Que contiene un conjunto de herramientas homogéneas que trabajan en un formato único.
 - C.- Para organizar las distintas herramientas de un entorno de desarrollo.
 - D.- Para conseguir una descomposición modular independiente.
- 10.- ¿Cuál de los siguientes servicios NO es necesario para un repositorio CASE?
 - A.- Consulta y actualización.
 - B.- Intercambio de datos.
 - C.- Vistas.
 - D.- Ingeniería inversa.

EJERCICIO (6 puntos) Nota: No se corregirá la contestación que exceda las dimensiones de tres caras del tamaño A4-ISO (210x297 mm)

Se quiere construir una aplicación que permita a un usuario jugar al Nim contra la máquina. El juego comienza con un montón n de cerillas. Se juega por turnos entre dos contrincantes. En la primera jugada, se pueden retirar tantas cerillas como se desee, pero hay que dejar al menos una en el montón. En el resto de los turnos, hay que tomar al menos una cerilla y, como máximo, el doble de las que ha tomado el contrincante en el turno anterior. Gana el que toma la última cerilla del montón. El programa debe permitir seleccionar el número de cerillas inicial, el que toma el usuario en cada turno y representar en la pantalla en cada instante el turno, el número de cerillas que restan en el montón y el que han retirado los contrincantes en la jugada anterior. Además, debe calcular al principio del juego si existe una estrategia de juego óptima para el número de cerillas inicial seleccionado. Dicha estrategia la seguirá durante el juego retirando las cerillas que correspondan en cada turno. Si no existe una estrategia óptima, la máquina jugará para ganar en el menor número de turnos posible.

Realice el diseño del programa utilizando notación de abstracciones y por aplicación del refinamiento progresivo.



ASIGNATURA: INGENIERÍA DEL SOFTWARE

TIPO DE EXAMEN: **A**

FECHA: 6 de septiembre de 2000 Hora: 11:30 Duración: 2 horas

MATERIAL: NINGUNO

¡ATENCIÓN! PONGA EL TIPO DE EXAMEN EN LA HOJA DE LECTURA ÓPTICA.

Conteste al test en la hoja de lectura óptica.

Sólo hay una respuesta correcta en cada pregunta.

PREGUNTAS DE TEST: El test es **ELIMINATORIO** (son necesarias 7 respuestas correctas para pasarlo),
(PUNTOS: 4)

- 1.- Desde el punto de vista de la asignatura, el software incluye:
 - A.- Las personas que desarrollan algún elemento de él.
 - B.- Los procedimientos de operación del producto.
 - C.- Los equipos físicos necesarios.
 - D.- El lenguaje empleado en la codificación.
- 2.- ¿Cuál de los siguientes puntos NO es un objetivo del mantenimiento?
 - A.- Mejorar o ampliar las prestaciones del producto.
 - B.- Modificar la aplicación para adecuarla al entorno de trabajo.
 - C.- Paliar el deterioro del producto software.
 - D.- Corregir errores no detectados durante el desarrollo.
- 3.- La notación DFD en el análisis:
 - A.- Tiene un carácter estático.
 - B.- Es dinámico pues refleja los procesos de transformación de la información.
 - C.- Es dinámico por representar la evolución del sistema a través de sus estados.
 - D.- Es dinámico por caracterizar las relaciones entre las entidades del sistema.
- 4.- La utilidad de la ocultación en el diseño se refleja en:
 - A.- Que se mantiene en secreto la forma de trabajar de otros programadores del equipo.
 - B.- Que se mantiene ocultas las metodologías empleadas por otros equipos de desarrollo.
 - C.- Que los módulos pueden actuar de diferente manera sin perder su naturaleza.
 - D.- La facilidad de depuración y mantenimiento.
- 5.- La metodología de diseño que da más importancia a las estructuras de datos es:
 - A.- Diseño orientado a las abstracciones.
 - B.- El diseño orientado a los datos.
 - C.- Diseño orientado a objetos.
 - D.- El método de Abbott.
- 6.- Los diagramas Entidad-Relación en el diseño:
 - A.- Tienen carácter dinámico.
 - B.- Es una notación híbrida.
 - C.- Tienen carácter estático.
 - D.- Permiten describir el diseño funcional y procedimental.
- 7.- El método de Abbott:
 - A.- Establece una manera metódica de descomposición modular basada en abstracciones.
 - B.- Puede considerarse como una ampliación de la técnica de refinamiento progresivo.
 - C.- Es una técnica de diseño estructurado.
 - D.- Parte del análisis de las estructuras de los datos de entrada/salida.
- 8.- La capacidad de un lenguaje para manejar excepciones:
 - A.- Se produce siempre en lenguajes de tipado débil.
 - B.- Está orientada a transferir el control de la ejecución cuando se produzcan situaciones anormales.
 - C.- Permite construir sistemáticamente pruebas de caja negra.
 - D.- Permite la sincronización entre tareas.
- 9.- El Repositorio CASE es un almacén común con la información necesaria para un grupo de herramientas o entorno de desarrollo. Necesita:
 - A.- Integración del control.
 - B.- Un equipo de verificación y validación.
 - C.- Herramientas de 4ª generación.
 - D.- Un servicio de metamodelo.
- 10.- Lo fundamental de los lenguajes de codificación de tercera generación es:
 - A.- Suelen ser herramientas para aplicaciones específicas.
 - B.- Son lenguajes que permiten programar simbólicamente.
 - C.- Están pensados para la enseñanza.
 - D.- Que utilizan y aplican las metodologías de programación estructurada.

EJERCICIO (6 puntos) Nota: No se corregirá la contestación que exceda las dimensiones de tres caras del tamaño A4-ISO (210x297 mm)

Realizar el análisis del siguiente sistema, descrito en lenguaje natural, mediante un modelo descrito con la notación DFD y, donde sea necesario, Entidad-Relación.

Se trata de automatizar el sistema de citas de una clínica. La clínica está atendida por cinco doctores que acuden a la consulta tres días a la semana. La asignación de citas se realiza según el horario de cada doctor excepto en el caso de que le surja una urgencia a dicho doctor. Si es así, las citas se desplazan a los siguientes huecos tras consultar por teléfono a los pacientes. La asignación de citas se actualiza en una base de datos en la que se mantiene la planificación de cada doctor, las citas, los datos de los pacientes, expediente clínico, etc.



ASIGNATURA: INGENIERÍA DEL SOFTWARE

TIPO DE EXAMEN: **E**

FECHA: 9 de septiembre de 2000 Hora: 16:00 Duración: 2 horas

MATERIAL: NINGUNO

¡ATENCIÓN! PONGA EL TIPO DE EXAMEN EN LA HOJA DE LECTURA ÓPTICA.

Conteste al test en la hoja de lectura óptica.

Sólo hay una respuesta correcta en cada pregunta.

PREGUNTAS DE TEST: El test es **ELIMINATORIO** (son necesarias 7 respuestas correctas para pasarlo),
(PUNTOS: 4)

- 1.- En relación con la calidad del software ¿cuál de las siguientes afirmaciones es cierta?
 - A.- La evaluación del nivel de calidad alcanzado para un criterio determinado se realiza con las métricas.
 - B.- La valoración significativa de la calidad se hace con las métricas.
 - C.- Los criterios de calidad se determinan según unos factores de calidad.
 - D.- Uno de los factores de calidad es la genericidad.
- 2.- En un documento de especificación de requisitos nunca puede aparecer:
 - A.- El diccionario de datos
 - B.- Una descripción en pseudocódigo del funcionamiento de cada módulo
 - C.- Modelos entidad-relación
 - D.- Diagramas de flujos de datos
- 3.- El modelo del sistema
 - A.- Recoge los módulos funcionales.
 - B.- Establece los métodos de verificación.
 - C.- Debe recoger los criterios de validación.
 - D.- Es la base del repositorio.
- 4.- El concepto de ocultación en el diseño...
 - A.- Persigue el objetivo de ocultar el contenido de los programas al resto de la organización.
 - B.- Evita que el usuario conozca el funcionamiento de un módulo.
 - C.- Permite que un elemento sirva para varias funciones.
 - D.- Facilita la depuración y el mantenimiento.
- 5.- El mecanismo de la herencia:
 - A.- Está ligado, sobre todo, al polimorfismo por sobrecarga.
 - B.- Facilita enormemente la reutilización del software.
 - C.- Se da en la metodología de diseño estructurado.
 - D.- Genera una cohesión lógica en el diseño.
- 6.- ¿Cuál de los pasos siguientes NO pertenece a la técnica de diseño de Jackson (JSP)?
 - A.- Construir una estructura de programa basada en estructuras de datos de Entrada/Salida.
 - B.- Analizar el problema y describir las estructuras de datos que se procesan.
 - C.- Identificar términos significativos para el diseño en las descripciones de las especificaciones del sistema.
 - D.- Definir tareas y situarlas en los módulos apropiados.
- 7.- Cuando los módulos pueden acceder a datos comunes de manera que los pueden manejar y estructurar sin tener en cuenta el resto de los módulos, tenemos acoplamiento:
 - A.- De datos.
 - B.- Externo.
 - C.- Por contenido.
 - D.- Común.
- 8.- El grado de acoplamiento entre módulos:
 - A.- Es débil en el acoplamiento común.
 - B.- Es moderado en el acoplamiento por contenido.
 - C.- Es una medida de la interrelación que hay entre estos.
 - D.- Es fuerte en el acoplamiento de datos.
- 9.- En cuanto al nivel de tipado de los lenguajes de programación:
 - A.- En el débil, el compilador se ocupa de realizar todas las conversiones necesarias.
 - B.- En el semirrígido se comprueban todos los tipos en CASI todos los casos.
 - C.- En el fuerte se puede evitar la comprobación de tipos mediante la compilación separada.
 - D.- FORTRAN es un lenguaje típico de nivel 0.
- 10.- ¿Cuál NO es una arquitectura típica de los entornos orientados al proceso?
 - A.- Los bancos de trabajo (upper CASE).
 - B.- Infraestructura de integración fija con colección personalizada de herramientas (NIST/ECMA).
 - C.- Entorno integrado con repositorio común (PCTE).
 - D.- Basada en integración mediante "software bus" (ESF).

EJERCICIO (6 puntos) Nota: No se corregirá la contestación que exceda las dimensiones de tres **caras** del tamaño A4-ISO (210x297 mm)

Realizar el análisis del siguiente sistema, descrito en lenguaje natural, mediante un modelo descrito con la notación DFD y, donde sea necesario, Entidad-Relación.

Se trata de automatizar el sistema de citas de una clínica. La clínica está atendida por cinco doctores que acuden a la consulta tres días a la semana. La asignación de citas se realiza según el horario de cada doctor excepto en el caso de que le surja una urgencia a dicho doctor. Si es así, las citas se desplazan a los siguientes huecos tras consultar por teléfono a los pacientes. La asignación de citas se actualiza en una base de datos en la que se mantiene la planificación de cada doctor, las citas, los datos de los pacientes, expediente clínico, etc.