



## **PROGRAMA DE LA ASIGNATURA**

**CURSO ACADÉMICO 2007/2008**

**ASIGNATURA:** Interacción Persona-Computadora

**CODIGO:** 208

**CRÉDITOS:** 4,5

**CURSO:** 2º

**IDIOMA:** Español

**PROFESORES DE LA ASIGNATURA:**

Ángela Carrasco Fernández

Luis Rodríguez Baena

Mercedes Vargas Díez

angela.carrasco@upsam.net

luis.rodriguez@upsam.net

mercedes.vargas@upsam.net

**PLAN DE ESTUDIOS:** 2002

**TITULACIÓN:** Ingeniería Informática

**CARÁCTER:** Anual

**TIPO :** Obligatoria

## **I. OBJETIVOS**

El principal objetivo de la asignatura es formar a los alumnos y alumnas en el diseño, prototipado y evaluación de software interactivo mediante una aproximación centrada en el usuario y ofrecer al alumno la formación teórica y práctica necesaria para el desarrollo de sistemas interactivos.

La formación en los conceptos del diseño de interfaces se complementará con el estudio de entornos de programación para el diseño de aplicaciones interactivas, haciendo hincapié en la utilización y programación de los diferentes componentes de la interfaz.

## **II. PRERREQUISITOS**

El alumno deberá dominar los aspectos básicos de la programación, tanto en lo que concierne al diseño de algoritmos (asignaturas de Fundamentos de la Programación I y Fundamentos de la Programación II) como en el manejo de algún lenguaje de programación estructurado (Programación I y Programación II).

Así mismo es conveniente que a lo largo del curso adquiera conocimiento de los conceptos fundamentales de la programación orientada a objetos obtenidos en asignaturas del segundo curso de la titulación (Programación Orientada a Objetos o Programación de Gestión I)

## **III. CONTENIDO**

### **PRIMERA PARTE: INTERACCIÓN PERSONA-COMPUTADORA**

#### **TEMA 1. FUNDAMENTOS DE INTERACCIÓN PERSONA-ORDENADOR.**

- 1.1. Orígenes de la disciplina..
- 1.2. Orígenes de los sistemas de interacción.
- 1.3. La interfaz de usuario
- 1.4. Objetivos de la HCI.
  - 1.4.1. Usabilidad.
    - 1.4.1.1. Facilidad de aprendizaje.
    - 1.4.1.2. Flexibilidad..
    - 1.4.1.3. Consistencia..
- 1.5. Interdisciplinariedad de la HCI.
  - 1.5.1. Factores que afectan a la interacción.
  - 1.5.2. Disciplinas que afectan a la HCI.

#### **TEMA 2. ESTILOS DE INTERACCIÓN, PARADIGMAS Y METÁFORAS.**

- 2.1. Estilos de interacción.
  - 2.1.1. Interfaz de línea de órdenes.
  - 2.1.2. Sistemas basados en menús.
  - 2.1.3. Interacción por lenguaje natural.
  - 2.1.4. Sistemas basados en diálogos e interrogación.



- 2.1.5. Formularios y hojas de cálculo.
- 2.1.6. Interfaces de manipulación directa.
- 2.1.7. Interfaces *point-and-click*.
- 2.1.8. Interfaces tridimensionales.
- 2.2. Paradigmas de interacción.
  - 2.2.1. El ordenador personal.
  - 2.2.2. Interfaces gráficas: la interfaz WIMP.
  - 2.2.3. Computación cooperativa.
  - 2.2.4. El hipertexto y la WWW.
  - 2.2.5. Interacción asistida e interfaces basados en agentes.
    - 2.2.5.1. Asistentes.
    - 2.2.5.2. Agentes.
  - 2.2.6. Computación ubicua.
  - 2.2.7. Entornos virtuales y realidad virtual.
- 2.3. Metáforas.
  - 2.3.1. Definición.
  - 2.3.2. Tipos.
    - 2.3.2.1. Metáforas visuales.
  - 2.3.3. Ejemplos.

### **TEMA 3. EVALUACIÓN**

- 3.1. Conceptos Ingeniería del Software de la Usabilidad.
  - 3.1.1. Desarrollo cíclico: análisis de requisitos, prototipado y evaluación.
  - 3.1.2. Diseño, implementación y lanzamiento.
- 3.2. Definición.
  - 3.2.1. La fase de evaluación del sistema.
  - 3.2.2. Objetivos de la evaluación.
  - 3.2.3. Estilos de evaluación:
    - 3.2.3.1. Estudios de laboratorio.
    - 3.2.3.2. Estudios de campo.
- 3.3. Métodos de evaluación.
- 3.4. Métodos de inspección.
  - 3.4.1. Recorrido cognitivo.
  - 3.4.2. Evaluación heurística.
    - 3.4.2.1. Principios heurísticos de usabilidad
- 3.5. Métodos de indagación.
  - 3.5.1. Observación de campo.
  - 3.5.2. *Focus Group*.
  - 3.5.3. Entrevistas.
  - 3.5.4. Cuestionarios.
  - 3.5.5. Grabación de uso.
- 3.6. Métodos de test.
  - 3.6.1. Medida de prestaciones.
  - 3.6.2. *Thinking aloud*.
  - 3.6.3. Interacción constructiva.
  - 3.6.4. Test retrospectivo.
  - 3.6.5. Método del conductor.
- 3.7. Elección de un método de evaluación.
- 3.8. Laboratorios de usabilidad.
- 3.9. Coste de la usabilidad

### **TEMA 4. DISEÑO Y PROTOTIPADO DE INTERFACES GRÁFICAS DE USUARIO.**

- 4.1. Principios fundamentales para el diseño de interfaces gráficas de usuario.
- 4.2. Las reglas de oro de Mandel.
  - 4.2.1. Colocar a los usuarios en el control de la interfaz.



- 4.2.2. Reducir la carga de memoria.
- 4.2.3. Interfaz de usuario consistente.
- 4.3. Diseño gráfico de interfaces.
  - 4.3.1. Objetivos del diseño gráfico.
  - 4.3.2. Organización perceptual de los objetos.
  - 4.3.3. Tipografía y estilo de escritura.
  - 4.3.4. Utilización del color.
  - 4.3.5. Utilización de imágenes.
- 4.4. Prototipado.
  - 4.4.1. Concepto de prototipo.
  - 4.4.2. Tipos de prototipos.
  - 4.4.3. Escenarios.
  - 4.4.4. Técnicas de prototipado.
    - 4.4.4.1. Prototipos en papel.
    - 4.4.4.2. Herramientas de diseño: PowerPoint y Visio.
    - 4.4.4.3. Herramientas de desarrollo: Visual Studio.
- 4.5. Principales elementos de una interfaz gráfica de escritorio
  - 4.5.1. Interacción por teclado.
  - 4.5.2. Ventanas.
  - 4.5.3. Menús.
  - 4.5.4. Botones.
  - 4.5.5. Cuadros de lista.
  - 4.5.6. Cuadrículas y tablas.
  - 4.5.7. Campos de texto.
  - 4.5.8. Ventanas secundarias y cuadros de mensaje.
  - 4.5.9. Otros elementos de la interfaz.
    - 4.5.9.1. Controles de fechas.
    - 4.5.9.2. Cuadros de grupo.
    - 4.5.9.3. Indicadores de progreso.
    - 4.5.9.4. Barras de desplazamiento.
    - 4.5.9.5. Fichas.
    - 4.5.9.6. Barras de herramientas y barras de estado.

## **TEMA 5. TEMAS AVANZADOS SOBRE USABILIDAD.**

- 5.1. Accesibilidad.
  - 5.1.1. Tipos de discapacidades y soluciones posibles.
    - 5.1.1.1. Ayudas tecnológicas.
  - 5.1.2. Accesibilidad en la Web.
    - 5.1.2.1. La iniciativa WAI
    - 5.1.2.2. Pautas de accesibilidad para el contenido en la Web
  - 5.1.3. Comprobación de la accesibilidad.
    - 5.1.3.1. Test de accesibilidad
- 5.2. Soporte al usuario.
  - 5.2.1. Requerimientos del soporte al usuario.
  - 5.2.2. Hipertexto e Hipermedia.
  - 5.2.3. Ayuda en línea.
- 5.3. Internacionalización.
  - 5.3.1. Definición de internacionalización del software.
  - 5.3.2. Definición de localización.
  - 5.3.3. Niveles de internacionalización y localización.
  - 5.3.4. Ventajas de hacer aplicaciones internacionales.
  - 5.3.5. Proceso de internacionalización/localización.
  - 5.3.6. Elementos relacionados con la internacionalización.



## **SEGUNDA PARTE: LENGUAJES DE PROGRAMACIÓN VISUAL**

### **TEMA 6. CONCEPTOS DE .NET.**

- 6.1. El entorno .NET Framework.
  - 6.1.1. Arquitectura .NET Framework.
  - 6.1.2. Common Language Runtime (CLR).
  - 6.1.3. La especificación de lenguaje común CLS.
- 6.2. Estructura de una aplicación .NET.
  - 6.2.1. El lenguaje intermedio de Microsoft (MSIL).
  - 6.2.2. Ensamblados, manifiestos y metadatos.
    - 6.2.2.1. Contenido de un ensamblado.
    - 6.2.2.2. Manifiesto de un ensamblado.
- 6.3. Modelo de ejecución de un aplicación .NET.
  - 6.3.1. El compilador JIT.
  - 6.3.2. Verificación.
  - 6.3.3. Dominios de una aplicación.
- 6.4. La biblioteca de clases Framework .NET.
  - 6.4.1. Espacios de nombres.

### **TEMA 7. ELEMENTOS DEL LENGUAJE VB.NET.**

- 7.1. Elementos del lenguaje.
  - 7.1.1. Caracteres.
  - 7.1.2. Identificadores.
  - 7.1.3. Comentarios.
  - 7.1.4. Palabras reservadas.
- 7.2. Estructura de un programa VB.NET.
  - 7.2.1. Soluciones y proyectos.
  - 7.2.2. Tipos de módulos. Estructura de los módulos.
    - 7.2.2.1. Instrucciones Option.
    - 7.2.2.2. Instrucciones Imports.
    - 7.2.2.3. Declaraciones.
    - 7.2.2.4. El método main.
- 7.3. Tipos de datos.
  - 7.3.1. Tipos de datos primitivos, de referencia y Object.
    - 7.3.1.1. Almacenamiento en memoria.
  - 7.3.2. Tipos de datos numéricos.
  - 7.3.3. Los datos Boolean y Char.
  - 7.3.4. El tipo Date.
  - 7.3.5. La clase String.
    - 7.3.5.1. Constructores, propiedades y métodos de String.
  - 7.3.6. El tipo Object.
  - 7.3.7. Estructuras.
    - 7.3.7.1. Declaración de estructuras.
  - 7.3.8. Literales.
  - 7.3.9. Variables.
    - 7.3.9.1. Declaración de variables.
    - 7.3.9.2. Visibilidad de las variables.
  - 7.3.10. Constantes y enumeraciones.
- 7.4. Operadores y expresiones.
  - 7.4.1. Operadores aritméticos.
  - 7.4.2. Operadores de concatenación.
  - 7.4.3. Operadores de asignación.
  - 7.4.4. Operadores lógicos.
    - 7.4.4.1. Operadores de relación.
    - 7.4.4.2. Operadores lógicos.



- 7.4.5. Prioridad de los operadores.
  - 7.4.6. Conversión de tipos.
    - 7.4.6.1. Funciones de conversión.
  - 7.5. Instrucciones de entrada/salida.
    - 7.5.1. Entrada salida por consola.
    - 7.5.2. Cuadros de entrada y cuadros de mensajes.
  - 7.6. Control de flujo.
    - 7.6.1. Instrucciones, líneas y bloques.
    - 7.6.2. Instrucciones de control.
      - 7.6.2.1. Instrucciones selectivas.
      - 7.6.2.2. Instrucciones repetitivas.
      - 7.6.2.3. Declaración de variables de bloque.
  - 7.7. Programación modular.
    - 7.7.1. Procedimientos Sub.
    - 7.7.2. Procedimientos Function.
    - 7.7.3. Accesibilidad de los procedimientos.
    - 7.7.4. Paso de argumentos.
      - 7.7.4.1. Paso por valor y referencia.
      - 7.7.4.2. Argumentos opcionales.
      - 7.7.4.3. Argumentos ParamArray.
    - 7.7.5. Procedimientos de evento.
  - 7.8. Arrays.
    - 7.8.1. La clase Array.
    - 7.8.2. Declaración e inicialización de arrays.
    - 7.8.3. Arrays dinámicos. Redimensión de un array.
    - 7.8.4. Arrays en procedimientos.
    - 7.8.5. Copia y borrado de arrays.
    - 7.8.6. Búsqueda y ordenación.
  - 7.9. Una introducción a la POO en VB.NET.
    - 7.9.1. Declaración de clases.
      - 7.9.1.1. Métodos.
      - 7.9.1.2. Propiedades.
      - 7.9.1.3. Constructores.
      - 7.9.1.4. Tiempo de vida de un objeto: recolección de basura.
      - 7.9.1.5. Visibilidad.
    - 7.9.2. Herencia.
      - 7.9.2.1. Sobrecarga.
      - 7.9.2.2. Clases abstractas.
  - 7.10. Manejo de excepciones.
    - 7.10.1. Control no estructurado y control estructurado de excepciones.
    - 7.10.2. El bloque Try.
      - 7.10.2.1. La cláusula Catch.
      - 7.10.2.2. La cláusula Finally.
    - 7.10.3. Excepciones creadas por el usuario.
- TEMA 8. DISEÑO DE UNA APLICACIÓN WINDOWS EN VB.NET.**
- 8.1. Programación en Windows.
    - 8.1.1. El modelo de programación orientado a eventos.
      - 8.1.1.1. Controladores de evento
  - 8.2. Proyectos y tipos de proyectos.
    - 8.2.1. Estructura de un aplicación Windows Forms.
  - 8.3. El entorno de programación.
  - 8.4. Tareas comunes.
    - 8.4.1. Manejo de texto.
    - 8.4.2. Color.



- 8.4.3. Fuentes.
- 8.4.4. Tamaño y posición.
- 8.4.5. Control del teclado y el ratón.
- 8.4.6. Control del foco de entrada..
- 8.5. El objeto Form.
  - 8.5.1. Ciclo de vida de un formulario. Eventos que intervienen.
  - 8.5.2. Mover y cambiar el tamaño de un formulario.
  - 8.5.3. Aspecto del formulario.
  - 8.5.4. Mostrar formularios.
- 8.6. Controles principales de Windows Forms
  - 8.6.1. Label y LinkLabel
  - 8.6.2. TextBox.
  - 8.6.3. Button.
  - 8.6.4. CheckBox y RadioButton.
  - 8.6.5. ListBox, CheckedListBox y ComboBox.
  - 8.6.6. PictureBox.
  - 8.6.7. HScrollBar, VScrollBar, TrackBar.
  - 8.6.8. NumericUpDown y DomainUpDown.
- 8.7. Menús.
  - 8.7.1. Menús desplegados.
  - 8.7.2. Menús contextuales.
- 8.8. Aplicaciones MDI.
- 8.9. Cuadros de diálogo comunes.
- 8.10. Otros controles.
  - 8.10.1. Timer.
  - 8.10.2. Panel, GroupBox y TabControl.
  - 8.10.3. ToolBar y StatusBar.
  - 8.10.4. ImageList, TreeView y ListView.
  - 8.10.5. RichTextBox.
- 8.11. Otros objetos útiles.
  - 8.11.1. Clipboard.
  - 8.11.2. Application.

## **TEMA 9. ADO.NET.**

- 9.1. Introducción al acceso a datos.
  - 9.1.1. Tecnologías de acceso a datos de Microsoft (DAO, RDO, ODBC, OLEDB, ADO, ADO.NET).
- 9.2. Proveedores de datos.
- 9.3. El modelo de objetos de ADO.NET.
  - 9.3.1. Clase Connection.
  - 9.3.2. Clase Command.
  - 9.3.3. Clase DataReader.
  - 9.3.4. Clase DataAdapter.
  - 9.3.5. Clase DataSet.
  - 9.3.6. Espacios de nombres para las clases de ADO.NET.
- 9.4. Modo conectado y modo desconectado.
- 9.5. Conexión a la base de datos.
- 9.6. Trabajo en modo conectado.
  - 9.6.1. El objeto Command.
    - 9.6.1.1. Comandos de recuperación.
    - 9.6.1.2. Comandos de acción.
  - 9.6.2. Recuperación de registros: el objeto DataReader.
- 9.7. Trabajo en modo desconectado.
  - 9.7.1. Adaptadores de datos.
  - 9.7.2. Conjuntos de datos.



- 9.7.2.1. El método Fill.
- 9.7.3. La clase DataTable.
  - 9.7.3.1. Filtrar y ordenar registros en un objeto DataTable.
- 9.7.4. La clase DataView.
- 9.7.5. Relaciones entre tablas.
- 9.7.6. Actualización de registros.
  - 9.7.6.1. Modificación, inserción y eliminación.
  - 9.7.6.2. Sincronización con el origen de datos.
  - 9.7.6.3. Ordenes de actualización.
- 9.8. Acceso a datos con aplicaciones Windows Forms.

## **IV. BIBLIOGRAFÍA**

### **A. Básica**

**LORES, J (ed):** *La interacción persona-ordenador*. Lerida: Asociación Interacción Persona Ordenador, 2001. Versión electrónica en <<http://griho.udl.es/ipo/libro.html>>

**BALENA, F.;** *Programación avanzada con Microsoft Visual Basic.NET*. Madrid: McGraw-Hill, 2003

### **B. Complementaria y de consulta**

**CHARTE, F.:** *Visual Basic .2005*. Madrid: Anaya, 2005.

**DIX, A.; FINLAY, J.; ABOWD, G; BEALE, R. :** *Human-Computer interaction. 3ª ed.* Hemel Hempstead, Hertfordshire: Prentice-Hall, 1998.

**GRANOLLERS, T.; LORÉS. J.; CAÑAS, J.J.;** *Diseño de sistemas interactivos centrados en el usuario*. Barcelona; Editorial UOC, 2005.

**KRUG, S.:** *No me hagas pensar. Una aproximación a la usabilidad en la Web*. Madrid: Pearson Educación, 2001.

**NIELSEN, J. ; THAIR, M,** *Usabilidad de páginas de inicio : análisis de 50 sitios Web*. Madrid: Pearson Alhambra, 2003

**NIELSEN, J.;** *Usabilidad, diseño de sitios Web* Madrid: Pearson Alhambra, 2000

**NORMAN, D:** *La psicología de los objetos cotidianos*. San Sebastián: Ed. Nerea, 1998.

**SHARP, H.; ROGERS, Y.; PREECE, J.;** *Interaction Design. Beyond Human-Computer Interaction. 2ª Ed.* Chichester, Hoboken, NJ: Wiley, 2007.

**SHNEIDERMAN, B.; PLAISANT, C.;** *Diseño de interfaces de usuario. Estrategias para una interacción persona-computadora efectiva*. Mexico: Addison Wesley, 2006.

**SILER, B.; SPOTTS, J.:** *Visual Basic .NET. Edición Especial*. Madrid: Prentice-Hall. 2003.

**SNYDER, C.;** *Paper Prototyping*. San Francisco: Morgan Kaufman, 2003.

### **C. Web**

*Microsoft Development Network*

[http://msdn2.microsoft.com/es-es/library/2x7h1hfk\(VS.80\).aspx](http://msdn2.microsoft.com/es-es/library/2x7h1hfk(VS.80).aspx)

*Microsoft Development Network Library*

<http://msdn.microsoft.com/library/default.asp>

*Programación en Castellano*

<http://www.programacion.com/direcciones/VisualBasic/>

*PortalVB*

<http://www.portalvb.com/>

*La web del Guille*

<http://www.elguille.info/default.aspx>



*The Open Source Development Environment for .NET*

<http://www.icsharpcode.net/OpenSource/SD/>

Entorno de desarrollo de código abierto para .NET más ligero que Visual Studio .NET. Requiere tener instalado .NET Framework 1.1.

*Asociación Interacción Persona-Ordenador*

<http://www.aipo.es>

*Cadius*

Portal dedicado a la arquitectura de información, la usabilidad, diseño de interacción y demás disciplinas centradas en el usuario. Dispone de una lista de correo muy activa.

<http://www.cadius.org>

*Fundación Sidar*

Fundación dedicada a la accesibilidad, sobre todo en la Web. Además de contener artículos, directivas y normativas sobre accesibilidad, dispone de una sección dedicada a la traducción de las especificaciones del World Wide Web Consortium.

<http://www.sidar.org>

*Alzado.org*

Alzado es una publicación electrónica con artículos referentes a usabilidad y accesibilidad.

<http://www.alzado.org>

*No solo usabilidad*

Magazín electrónico que pretende servir de herramienta para la difusión, divulgación e intercambio de conocimiento entre desarrolladores e investigadores Web.

<http://www.nosolousabilidad.com/>

*Usalo*

*Bitácora dónde se divulgan ideas y consejos sobre Usabilidad y sus disciplinas afines: Accesibilidad, Arquitectura de la Información, Findability, HCI, Diseño, Experiencia de Usuario...*

<http://usalo.es/>

*Useit.com*

Useit es la página personal de Jacob Nielsen, uno de los gurús más importantes en usabilidad. En ella se incluyen columnas, como la famosa alertbox y artículos del autor.

<http://www.useit.com>

*Cuerpo de conocimiento de usabilidad*

Versión española de Usability Body Of Knowledge (<http://usabilitybok.org>), proyecto dedicado a la creación de una referencia que represente el conocimiento colectivo de la profesión de usabilidad.

<http://es.usabilitybok.org/>

*Comunicación Persona-Máquina*

Página de la asignatura de Comunicación Persona-Máquina de la Escuela Universitaria de Ingeniería Técnica de la Universidad de Oviedo realizada por la profesora Ana Belén Martínez.

<http://petra.euitio.uniovi.es/asignaturas/com.per.maq/>

*Special Interest Group Computer Human Interaction de la ACM (SIGCHI)*

<http://www.acm.org/sigchi/>

*HCI Bibliography : Human-Computer Interaction Resources*

Bibliografía del SIGCHI sobre interacción persona-computadora

<http://www.hcibib.org/>



### *Bad Human Factors*

Página con ejemplos de objetos difíciles de utilizar o mal diseñados. Una página para pasar un rato divertido.

<http://www.baddesigns.com/>

### *Interface Hall of Shame*

Una página con ejemplos de errores de diseño encontrados en aplicaciones reales.

<http://homepage.mac.com/bradster/iarchitect/>

### *Windows XP - Guidelines for Applications*

Incluye un enlace al archivo WindowsXPDesignGuidelines.exe con la guía de diseño visual para Windows XP.

<http://www.microsoft.com/whdc/Resources/windowsxp/default.mspx>

### *Windows Vista User Experience*

<http://msdn2.microsoft.com/en-us/library/aa511258.aspx>

### *UI Design Guidelines*

[http://msdn2.microsoft.com/en-us/library/bb166159\(VS.80\).aspx](http://msdn2.microsoft.com/en-us/library/bb166159(VS.80).aspx)

### *Java Look & Feel Design Guidelines*

Guía de estilo para el desarrollo de aplicaciones en Java.

<http://java.sun.com/products/jlff/ed2/book/index.html>

### *Apple Human Interface Guidelines*

Guía de estilo para el desarrollo de aplicaciones en Macintosh OSX.

<http://developer.apple.com/documentation/macos8/HumanInterfaceToolbox/HumanInterfaceGuide/humaninterfaceguide.html>

### *IBM Web Design Guidelines*

Guía de estilo de IBM para aplicaciones web.

<http://www-03.ibm.com/easy/page/572>

### *Web Style Guide: design principles for creating web sites*

Guía de estilo para sitios web de la Universidad de Yale.

<http://www.webstyleguide.com/index.html>

## **V. MÉTODOS DOCENTES**

Para la primera parte de la asignatura se impartirán sesiones teóricas intercaladas con ejercicios prácticos. En la segunda parte de la asignatura se combinarán las sesiones teóricas con prácticas de laboratorio.

## **VI. EVALUACIÓN**

La evaluación de la asignatura se realizará a partir de las notas obtenidas en el examen teórico y las notas obtenidas a lo largo del periodo de docencia del curso. La nota final será la nota media obtenida en cada uno de los cuatrimestres (siempre y cuando hubieran superado ambos) o, en el caso de presentarse al examen final., la nota obtenida en éste.

## **Prácticas**

El alumno deberá entregar las prácticas indicadas por el profesor, siendo éste un requisito indispensable para la superación de la asignatura.



### **Examen**

Se realizarán un examen parcial liberatorio en la convocatoria de febrero que abarcará la materia impartida en el primer cuatrimestre.

Aquellas personas que hubieran superado este examen realizarán un segundo parcial en la convocatoria de junio. El resto de los alumnos realizarán un examen final en la convocatoria de junio que abarcará toda la materia de la asignatura.

En el caso de suspender la parte teórica en la convocatoria de junio, se podrán presentar a un examen final en la convocatoria de septiembre.

En Madrid, a 28 de mayo de 2007