

Programación semanal

En la programación semanal te presentamos un **reparto del trabajo de la asignatura** a lo largo de las semanas del cuatrimestre.

	CONTENIDO TEÓRICO	ACTIVIDADES (15 puntos)	CLASES EN DIRECTO
Semana 1	Tema 1: Introducción a la programación orientada a objetos 1.1. Introducción y objetivos. 1.2. Diseño modular de programas. 1.3. Principios de diseño de módulos. 1.4. El paradigma de orientación objetos. 1.5. Tarjetas CRC (Class, Responsibility Collaboration). 1.6. Modelado con objetos. 1.7. Cuaderno de ejercicios.	Se puntuará la asistencia a 2 clases presenciales virtuales a elegir a lo largo del cuatrimestre (0.25 puntos cada una) Test Tema 1 (0.15 puntos)	Presentación de la asignatura y clase del tema 1
Semana 2	Tema 2: Relaciones entre clases 2.1. Introducción y objetivos. 2.2. Relaciones entre clases. 2.3 Cardinalidad/multiplicidad de una relación. 2.4 Navegabilidad de una asociación.		Clase del tema 2
Semana 3	Tema 2: Relaciones entre clases (continuación) 2.5. Tipos de relaciones entre clases. 2.6. Cuaderno de ejercicios.	Test Tema 2 (0.15 puntos)	Clase del tema 2 (continuación)

	CONTENIDO TEÓRICO	ACTIVIDADES (15 puntos)	CLASES EN DIRECTO
Semana 4	Tema 3: Herencia, clases abstractas y polimorfismo 3.1. Introducción y objetivos. 3.2. Relación de generalización y especialización entre clases. 3.3. Principio de sustitución de Liskov. 3.4. Herencia	Laboratorio1 Modelado e implementación de clases (5 puntos)	Clase del tema 3 Clase laboratorio1
Semana 5	Tema 3: Herencia, clases abstractas y polimorfismo (continuación) 3.5. Clases abstractas 3.6. Polimorfismo 3.7. Ligadura de métodos 3.8. Cuaderno de ejercicios.	Test tema 3 (0,15 puntos)	Clase tema 3 (Continuación)
Semana 6	Tema 4. Gestión de errores mediante excepciones 4.1. Introducción y objetivos. 4.2. Tipos de errores en un programa. 4.3. Gestión clásica de errores mediante código de retorno. 4.4. Concepto de excepción. 4.5. El modelo de errores de Java. 4.6. Tratamiento de excepciones. 4.7. Lanzar una excepción (<i>throw</i>). 4.8. Manejadores de excepciones. 4.9. Declaración de excepciones en métodos (excepciones comprobadas). 4.10. Cuaderno de ejercicios.	Test Tema 4 (0,15 puntos)	Clase del tema 4

	CONTENIDO TEÓRICO	ACTIVIDADES (15 puntos)	CLASES EN DIRECTO
Semana 7	<p>Tema 5. Interfaces gráficas de usuario (I)</p> <p>5.1. Introducción y objetivos.</p> <p>5.2. Introducción a las interfaces gráficas de usuario.</p> <p>5.3. Interacción persona ordenador (HCI Human-Computer Interaction).</p> <p>5.4. Historia de las interfaces gráficas de usuario.</p> <p>5.5. Prototipado.</p> <p>5.6. Aspectos de diseño de las interfaces gráficas usuario.</p> <p>5.7. Principios de diseño de interfaces de usuario.</p> <p>5.8. Estilos de interacción.</p> <p>5.9. Componentes de un sistema de ventanas.</p> <p>5.10. Elementos comunes de una interfaz gráfica de usuario.</p> <p>5.11. Cuaderno de ejercicios.</p>	<p>Test Tema 5</p> <p>(0,15 puntos)</p>	<p>Clase del tema 5</p>
Semana 8	<p>Tema 6. Interfaces gráficas de usuario (II)</p> <p>6.1. Introducción y objetivos.</p> <p>6.2. Patrones de diseño e interfaces gráficas de usuario.</p>		<p>Clase del tema 6</p>
Semana 9	<p>Tema 6. Interfaces gráficas de usuario (II) (continuación)</p> <p>6.3. Programación dirigida por eventos.</p> <p>6.4. Implementación de interfaces gráficas en Java.</p> <p>6.5. Cuaderno de ejercicios.</p>	<p>Laboratorio 2 Diseño e implementación interfaz gráfica de usuario</p> <p>(5 puntos)</p> <p>Test Tema 6</p> <p>(0,15 puntos)</p>	<p>Clase del tema 6 (continuación)</p> <p>Clase resolución laboratorio 1</p> <p>Clase laboratorio 2</p>

	CONTENIDO TEÓRICO	ACTIVIDADES (15 puntos)	CLASES EN DIRECTO
Semana 10	<p>Tema 7. Introducción a la programación concurrente</p> <p>7.1 Introducción y objetivos.</p> <p>7.2. Introducción a la programación concurrente.</p> <p>7.3. Unidades de ejecución y planificación: procesos e hilos.</p> <p>7.4. Concurrencia, paralelismos y ejecución concurrente paralela.</p> <p>7.5. Soporte para la ejecución de múltiples hilos.</p> <p>7.6. Modelos de concurrencia e intercomunicación.</p> <p>7.7. Lenguajes de programación y la programación concurrente.</p> <p>7.8. Cuaderno de ejercicios.</p>	<p>Test Tema 7</p> <p>(0,15 puntos)</p>	<p>Clase del tema 7</p>
Semana 11	<p>Tema 8. Sincronización de hilos</p> <p>8.1 Introducción y objetivos.</p> <p>8.2. Tipos de interacciones en ejecución concurrente.</p> <p>8.3. Sincronización de hilos.</p> <p>8.4. Exclusión mutua.</p> <p>8.5. Mecanismos de sincronización.</p>	<p>Actividad grupal: Programación concurrente aplicada al modelado de programas</p> <p>(3 puntos)</p>	<p>Clase del tema 8</p> <p>y</p> <p>presentación de la actividad grupal</p>
Semana 12	<p>Tema 8. Sincronización de hilos (Continuación)</p> <p>8.6. Problemas de control y sincronización de hilos.</p> <p>8.7. Sincronización de hilos en Java.</p> <p>8.8. Monitores en Java.</p> <p>8.9. Tipos de datos colección seguros.</p> <p>8.10. Cuaderno de ejercicios.</p>	<p>Test Tema 8</p> <p>(0,15 puntos)</p>	<p>Clase del tema 8 (continuación)</p>

	CONTENIDO TEÓRICO	ACTIVIDADES (15 puntos)	CLASES EN DIRECTO
Semana 13	Tema 9. Gestión y planificación de hilos 9.1. Introducción y objetivos. 9.2. Introducción a la planificación de procesos e hilos. 9.3. Políticas de planificación de hilos. 9.4. Soporte de ejecución de hilos. 9.5. Patrón de diseño concurrente: ThreadPool. 9.6. Cuaderno de ejercicios.		Clase del tema 9 Clase de resolución Laboratorio 2
Semana 14	Tema 10. Estrategias y técnicas de prueba de software 10.1. Introducción y objetivos. 10.2. Introducción a las pruebas software . 10.3. Estrategias de pruebas. 10.4. Diseños de casos de pruebas.		Clase del tema 10
Semana 15	Tema 10. Estrategias y técnicas de prueba de software (continuación) 10.5. Pruebas de caja blanca. 10.6. Técnicas de prueba de caja blanca. 10.7. Técnicas de prueba de caja negra. 10.8. Pruebas unitarias. 10.9. Cuaderno de ejercicios.	Test Tema 10 (0,15 puntos)	Sesión de explicación del modelo de examen Clase del tema 10 (Continuación) Clase de resolución de la actividad Grupal
Semana 16	Semana de exámenes		

NOTA

Esta **Programación semanal** puede ser modificada si el profesor lo considera oportuno para el enriquecimiento de la asignatura.