

## 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 (20 puntos)
Semana 1	<b>Tema 1. Introducción a la programación orientada a objetos</b> 1.1. Introducción y objetivos 1.2. Introducción a la programación orientada a objetos 1.3. Diseño de clases 1.4. Introducción a UML para el modelado de los problemas	<b>Test Tema 1</b> (0,2 puntos)
Semana 2	<b>Tema 2. Relaciones entre clases</b> 2.1. Introducción y objetivos 2.2. Abstracción y herencia 2.3. Conceptos avanzados de herencia 2.4. Polimorfismo 2.5. Composición y agregación 2.6. This y super  <b>Tema 3. Introducción a los patrones de diseño para problemas orientados a objetos</b> 3.1. Introducción y objetivos 3.2. ¿Qué son los patrones de diseño? 3.3. Patrón Factory 3.4. Patrón Singleton 3.5. Patrón Observer 3.6. Patrón Composite	<b>Laboratorio #1:</b> Diseño e implementación de clases (7,5 puntos)  <b>Foro:</b> Parones de Diseño de Software: «Entendiendo los parones desde su implementación y Utilidad» (2,6 puntos)  <b>Test Tema 2</b> (0,2 puntos)  <b>Test Tema 3</b> (0,2 puntos)
Semana 3	<b>Tema 4. Excepciones</b> 4.1. Introducción y objetivos 4.2. Excepciones 4.3. Captura y gestión de excepciones 4.4. Lanzamiento de excepciones 4.5. La clase Throwable 4.6. Creación de excepciones	<b>Test Tema 4</b> (0,2 puntos)
Semana 4	<b>Tema 5. Interfaces de usuarios</b> 5.1. Introducción y objetivos 5.2. AWT 5.3. Gestores de posicionamiento 5.4. ¿Qué son los eventos? 5.5. Adaptadores  <b>Tema 6. Interfaces de usuarios (II)</b> 6.1. Introducción y objetivos 6.2. Eventos: definición y captura 6.3. Desarrollo de interfaces de usuario 6.4. SWING	<b>Laboratorio #2:</b> Diseño e implementación de interfaz de usuario (7,5 puntos)  <b>Test Tema 5</b> (0,2 puntos)  <b>Test Tema 6</b> (0,2 puntos)

	CONTENIDO TEÓRICO	ACTIVIDADES (20 puntos)
Semana 5	<b>Tema 7. Introducción a la programación concurrente</b> 7.1. Introducción y objetivos 7.2. Introducción a la programación concurrente 7.3. El concepto de proceso e hilo 7.4. Interacción entre procesos o hilos 7.5. Los hilos en Java 7.6. Ventajas e inconvenientes de la programación concurrente	<b>Test Tema 7</b> (0,2 puntos)
Semana 6	<b>Tema 8. Sincronización</b> 8.1. Introducción y objetivos 8.2. Sincronización 8.3. Mecanismos de bloqueo 8.4. Mecanismos de comunicación 8.5. Monitores 8.6. Bibliotecas de Java para concurrencia  <b>Tema 9. Gestión de hilos</b> 9.1. Introducción y objetivos 9.2. Ciclo de vida de un hilo 9.3. La clase Thread 9.4. Planificación de hilos 9.5. Grupos hilos 9.6. Hilos de tipo demonio	<b>Test Tema 8</b> (0,2 puntos)  <b>Test Tema 9</b> (0,2 puntos)
Semana 7	<b>Tema 10. Problemas comunes dentro de la programación concurrente</b> 10.1. Introducción y objetivos 10.2. El problema de los productores consumidores 10.3. El problema de los lectores y escritores 10.4. El problema de la cena de los filósofos  <b>Tema 11. Documentación de <i>software</i></b> 11.1. Introducción y objetivos 11.2. ¿Por qué es importante documentar el <i>software</i> ? 11.3. Documentación de diseño 11.4. Uso de herramientas para la documentación: JavaDoc	<b>Test Tema 10</b> (0,2 puntos)  <b>Test Tema 11</b> (0,2 puntos)
Semana 8	<b>Tema 12. Pruebas de <i>software</i></b> 12.1. Introducción y objetivos 12.2. Introducción a las pruebas del <i>software</i> 12.3. Prueba de unidad 12.4. Prueba de integración 12.5. Prueba de validación 12.6. Prueba del sistema	<b>Test Tema 12</b> (0,2 puntos)

CONTENIDO TEÓRICO	ACTIVIDADES (20 puntos)
Semana 9	Semana de repaso
Semana 9	Semana de exámenes

**NOTA**

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