

## 7 Ficha de la materia "Programación"

DENOMINACIÓN DE LA MATERIA  PROGRAMACIÓN	MÓDULO AL QUE PERTENECE
	CRÉDITOS ECTS 30
	CARÁCTER Mixto (ver asignaturas)
<b>DURACIÓN Y UBICACIÓN TEMPORAL DENTRO DEL PLAN DE ESTUDIOS</b>  Materia compuesta por 7 asignaturas programadas en el 2º, 3º, 4º, 5º y 6º semestre, tal y como se recoge a continuación en la tabla de asignaturas	
<b>COMPETENCIAS Y RESULTADOS DEL APRENDIZAJE QUE EL ESTUDIANTE ADQUIERE CON ESTA MATERIA</b>	
COMPETENCIAS	
<b>Competencias específicas</b>	
CE-1	Conocer profundamente los cimientos esenciales y fundacionales de la informática, abarcando tanto conceptos y teorías abstractas como los valores y los principios profesionales, subrayando los aspectos esenciales de la disciplina que permanecen inalterables ante el cambio tecnológico.
CE-2	Formalización y especificación de problemas reales cuya solución requiere el uso de la informática.
CE-3/4	Capacidad de elegir y usar los métodos analíticos y de modelización relevantes, y de describir una solución de forma abstracta.
CE-6	Comprender intelectualmente el papel central que tienen los algoritmos y las estructuras de datos, así como una apreciación del mismo.
CE-8	Poseer destrezas fundamentales de la programación que permitan la implementación de los algoritmos y las estructuras de datos en el software.
CE-9	Poseer las destrezas que se requieren para diseñar e implementar unidades estructurales mayores que utilizan los algoritmos y las estructuras de datos, así como las interfaces por las que se comunican estas unidades.
CE-13/18	Comprender lo que pueden y no pueden conseguir las tecnologías actuales, y las limitaciones de la informática, que implica distinguir entre lo que, inherentemente, la informática no es capaz de hacer y lo que puede lograrse a través de la ciencia y la tecnología futuras.
CE-19/20	Conocimiento de los tipos apropiados de soluciones, y comprensión de la complejidad de los problemas informáticos y la viabilidad de su solución.
CE-24	Elegir y usar los lenguajes de programación adecuados al tipo de aplicación a desarrollar.
CE-25	Concebir y diseñar la arquitectura de un sistema software.
CE-29	Diseñar, desarrollar y evaluar la seguridad de los sistemas, aplicaciones, servicios informáticos y sistemas operativos sobre los que se ejecutan, así como de la información que proporcionan.

CE-34	Crear prototipos, simulaciones o modelos que permitan la validación del sistema con el cliente.
CE-40	Comprender el concepto esencial de proceso en cuanto a su relación con la informática, especialmente la ejecución de los programas y la operación del sistema.
CE-42	Combinar la teoría y la práctica para realizar tareas informáticas.

**Competencias generales:**

CG-1/21	Capacidad de resolución de problemas aplicando conocimientos de matemáticas, ciencias e ingeniería.
CG-2/CE45	Capacidad para el aprendizaje autónomo y la actualización de conocimientos, y reconocimiento de su necesidad en el área de la informática.
CG-3/4	Saber trabajar en situaciones de falta de información y bajo presión, teniendo nuevas ideas, siendo creativo.
CG-5	Capacidad de gestión de la información.
CG-6	Capacidad de abstracción, análisis y síntesis.
CG-7/8/9/10/16/17	Capacidad para trabajar dentro de un equipo, organizando, planificando, tomando decisiones, negociando y resolviendo conflictos, relacionándose, y criticando y haciendo autocrítica.
CG-11/12/20	Capacidad para tomar iniciativas y espíritu emprendedor, el liderazgo, la dirección, la gestión de equipos y proyectos.
CG-19	Capacidad para usar las tecnologías de la información y la comunicación.

**RESULTADOS DEL APRENDIZAJE**

- Traducir especificaciones de tipos abstractos de datos (TADs) a implementaciones.
- Programar aplicaciones mediante librerías existentes de TADs, iteradores, etc.,
- Documentar clases y bibliotecas, tanto de manera pública (hacia el cliente).
- Realizar pruebas para asegurar el correcto funcionamiento de un TAD así como
- Programar aplicaciones, en al menos un lenguaje procedimental ampliamente utilizado, como puede ser C, que le permiten la comunicación con el sistema.
- Programar scripts que automaticen determinadas tareas o faciliten llevar a cabo pruebas funcionales de programas.
- Resolver problemas algorítmicos no triviales.
- Razonar sobre la complejidad algorítmica.
- Razonar sobre la terminación.
- Usar y definir estructuras de datos eficientes y adecuadas a cada problema.
- Conocer los fundamentos de la programación lógica y sus campos de aplicación.
- Modelar declarativamente la solución a un problema y expresarlo elegantemente con un programa lógico eficiente.
- Modelar mediante procedimientos finitos conjuntos y lenguajes infinitos.
- Adquirir destreza en la aplicación de los diferentes métodos de demostración.

- Distinguir y reconocer las distintas clases de lenguajes y sus autómatas asociados según la jerarquía de Chomsky.
- Conocer modelos de cómputo universales así como los límites de lo que puede o no ser computado mediante un algoritmo.
- Ser capaz de diseñar y construir un sistema para analizar léxica, sintáctica y semánticamente un código escrito en un determinado formato.
- Reconocer la concurrencia inherente a un sistema SW.
- Especificar propiedades relativas a la ejecución concurrente de un sistema SW.
- Desarrollar aplicaciones concurrentes, en al menos un lenguaje ampliamente utilizado, como puede ser Java.
- Analizar propiedades y riesgos atribuibles a la ejecución concurrente de un sistema SW.

#### ASIGNATURAS DE QUE CONSTA

ASIGNATURA	CRÉDITOS ECTS	CARÁCTER	UBICACIÓN TEMPORAL	DEPARTAMENTO RESPONSABLE
Algoritmos y Estructura de Datos	6	Obligatoria	3er semestre	DLSIIS
Programación II	6	Obligatoria	2º semestre	DLSIIS
Programación para sistemas	3	Obligatoria	3er semestre	DLSIIS
Programación Declarativa: Lógica y Restricciones	3	Obligatoria	6er semestre	DIA
Lenguajes Formales, Autómatas y Computabilidad	6	Obligatoria	3er semestre	DIA
Procesadores de Lenguajes	3	Obligatoria	5º semestre	DLSIIS
Concurrencia	3	Obligatoria	4º semestre	DLSIIS

#### REQUISITOS PREVIOS QUE HAN DE CUMPLIRSE PARA PODER ACCEDER A LAS ASIGNATURAS DE ESTA MATERIA

ASIGNATURA	REQUISITOS
Algoritmos y Estructuras de Datos	Programación I, Programación II
Programación II	Programación I, Lógica, Matemática Discreta I.
Programación de Sistemas	Programación I
Programación Declarativa: Lógica y Restricciones	Programación I
Lenguajes Formales, Autómatas y Computabilidad	Matemática Discreta I, Matemática Discreta II, Lógica
Procesadores de Lenguajes	Algoritmos y Estructuras de Datos

Concurrencia

Lenguajes Formales, Autómatas y Computabilidad  
 Programación II  
 Lenguajes Formales, Autómatas y Computabilidad

**ACTIVIDADES FORMATIVAS, SU DISTRIBUCIÓN EN CRÉDITOS ECTS, SU METODOLOGÍA DE ENSEÑANZA-  
 APRENDIZAJE Y SU RELACIÓN CON LAS COMPETENCIAS QUE DEBE ADQUIRIR EL ESTUDIANTE**

	Actividades formativas									Métodos docentes					
	CT	S/T	ETAI	CP	T	L	EG	PA	PO	LM	EC	RE	ABP	AOP	AC
<b>Compe- tencias / Créditos</b>	<b>5</b>	<b>0,4</b>	<b>10</b>	<b>3</b>	<b>0,6</b>	<b>3</b>	<b>1</b>	<b>4</b>	<b>3</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
CE-1	x	x	x	x	x		x								
CE-2	x	x	x	x	x		x								
CE-3/4	x		x	x	x		x	x							
CE-6	x		x	x		x	x		x						
CE-8	x		x	x		x		x	x						
CE-9	x		x	x		x		x	x						
CE-13/18	x	x	x		x		x								
CE-19/20	x		x	x											x
CE-24	x	x	x	x	x				x						
CE-25	x	x	x	x					x						
CE-29	x	x	x	x					x						
CE-34				x					x						
CE-40	x	x	x	x					x						
CE-42															x

La tabla anterior muestra las actividades formativas, su distribución en créditos ECTS y su relación con las competencias que debe adquirir el estudiante. La relación entre los métodos docentes y competencias

se detallan más arriba en el punto 5.3.1, así como los códigos utilizados para abreviar en la tabla las actividades formativas y los métodos docentes.

### **ACTUACIONES DIRIGIDAS A LA COORDINACIÓN DE LAS ACTIVIDADES FORMATIVAS Y SISTEMAS DE EVALUACIÓN DENTRO DE ESTA MATERIA**

La coordinación en esta materia se va llevar a cabo por medio de la Comisión de Coordinación Vertical establecida para la misma, tal y como se describe en la sección 5. Planificación de las enseñanzas.

### **SISTEMA DE EVALUACIÓN DE LOS RESULTADOS DEL APRENDIZAJE ALCANZADOS Y SISTEMA DE CALIFICACIONES**

Se van a utilizar los siguientes métodos de evaluación:

- Pruebas objetivas (verdadero/falso, elección múltiple, emparejamiento de elementos...), de respuesta corta.
- Pruebas de respuesta larga, de desarrollo.
- Informes/memorias de prácticas.
- Trabajos y proyectos.
- Pruebas de ejecución de tareas reales y/o simuladas.
- Escalas de actitudes (para recoger opiniones, valores, habilidades sociales y directivas, conductas de interacción,...).

### **BREVE DESCRIPCIÓN DE LOS CONTENIDOS**

#### **Programación II**

Programación orientada a Objetos  
Concepto de Tipo Abstracto de Datos  
Manejo de Estructuras de Datos dinámicas -Documentación de código.

#### **Programación para Sistemas**

Programación para sistemas operativos  
Automatización de tareas mediante scripts

#### **Algoritmos y estructuras de datos**

Algoritmos básicos.  
Diseño de algoritmos.  
Implementación de tipos de datos.  
Análisis de terminación, corrección y complejidad.

#### **Procesadores de Lenguajes**

Análisis Léxico  
Análisis Sintáctico  
Análisis Semántico.

#### **Concurrencia**

Análisis y diseño de sistemas concurrentes  
Programación de aplicaciones concurrentes

Especificación de sistemas concurrentes

**Programación Declarativa: Lógica y Restricciones**

Programación lógica y resolución de restricciones

Lenguajes declarativos lógicos, ISO-Prolog

**Lenguajes formales, autómatas y computabilidad**

Teoría y diseño de lenguajes formales y gramáticas

Teoría y diseño de autómatas finitos y autómatas con pila

Máquinas de Turing y modelos de cómputo universales

Teoría de la computabilidad