

<p>Denominación de la materia</p> <p>PROGRAMACIÓN</p>	<p>N créditos ECTS = 36 carácter = MIXTO</p>
<p>Ubicación dentro del plan de estudios y duración</p> <p>La materia Programación está formada por 6 asignaturas de 6 créditos ECTS cada una de ellas, que se impartirán en los tres primeros años de la titulación. La distribución temporal planteada es la siguiente:</p> <ul style="list-style-type: none"> - INFORMÁTICA::PROGRAMACIÓN I, 1º semestre del 1º curso - PROGRAMACIÓN II e INFORMÁTICA::ALGORITMOS Y ESTRUCTURAS DE DATOS I, 2º semestre del 1º curso - ALGORITMOS Y ESTRUCTURAS DE DATOS II, 1º semestre del 2º curso - LÓGICA PARA LA COMPUTACIÓN, 1º semestre del 3º curso - TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES, 2º semestre del 3º curso 	
<p>Requisitos previos</p> <ul style="list-style-type: none"> • Conocimientos principales de la matemática • Capacidad de buscar información por medios propios • INFORMÁTICA::PROGRAMACIÓN I es requisito previo para PROGRAMACIÓN II • INFORMÁTICA::ALGORITMOS Y ESTRUCTURAS DE DATOS I es requisito previo para ALGORITMOS Y ESTRUCTURAS DE DATOS II • ALGORITMOS Y ESTRUCTURAS DE DATOS II es requisito previo para LÓGICA PARA LA COMPUTACIÓN • LÓGICA PARA LA COMPUTACIÓN es requisito previo para TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES 	
<p>Competencias a desarrollar</p> <p><i>Competencias específicas</i></p> <p>CORRESPONDIENTES AL MÓDULO DE FORMACIÓN BÁSICA:</p> <ol style="list-style-type: none"> 1. Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización. 3. Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería. 4. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería. 	

5. Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

CORRESPONDIENTES AL MÓDULO COMÚN A LA RAMA DE INFORMÁTICA:

7. Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
12. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
13. Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
14. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
19. Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
22. Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.

CORRESPONDIENTES AL MÓDULO DE TECNOLOGÍA ESPECÍFICA: INGENIERÍA DEL SOFTWARE

25. Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
26. Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.
27. Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.
28. Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.
30. Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.

CORRESPONDIENTES AL MÓDULO DE TECNOLOGÍA ESPECÍFICA: TECNOLOGÍAS DE LA INFORMACIÓN

32. Capacidad para seleccionar, diseñar, desplegar, integrar, evaluar, construir, gestionar, explotar y mantener las tecnologías de hardware, software y redes, dentro de los parámetros de coste y calidad adecuados.
33. Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas

basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas.

Competencias transversales

INSTRUMENTALES

- I1 Capacidad de análisis, síntesis y evaluación
- I2 Capacidad de organización y planificación
- I3 Comunicación oral y escrita en la lengua nativa
- I5 Capacidad de abstracción: capacidad de crear y utilizar modelos que reflejen situaciones reales.
- I6 Capacidad de diseñar y realizar experimentos sencillos y analizar e interpretar sus resultados.
- I7 Capacidad de buscar, relacionar y estructurar información proveniente de diversas fuentes y de integrar ideas y conocimientos.
- I8 Resolución de problemas
- I9 Capacidad de tomar decisiones.
- I10 Capacidad para argumentar y justificar lógicamente las decisiones tomadas y las opiniones.

PERSONALES

- P1 Capacidad de actuar autónomamente.
- P2 Capacidad de trabajar en situaciones de falta de información y/o bajo presión.
- P3 Capacidad de integrarse rápidamente y trabajar eficientemente en equipos unidisciplinares y de colaborar en un entorno multidisciplinar.
- P5 Capacidad de relación interpersonal.

SISTÉMICAS

- S1 Razonamiento crítico
- S3 Aprendizaje autónomo
- S4 Adaptación a nuevas situaciones
- S5 Creatividad
- S6 Liderazgo
- S7 Tener iniciativa y ser resolutivo.
- S9 Tener motivación por la calidad y la mejora continua.

Coordinación de los sistemas de evaluación en la materia

La materia es una unidad coherente desde el punto de vista disciplinar. Se llevará a cabo una coordinación vertical para garantizar la adquisición de las competencias y conocimientos establecidos.

El sistema de evaluación es común entre todas las asignaturas de la materia. En todos los grupos de estudiantes formados en una misma asignatura se realizarán actividades formativas similares.

Se empleará un sistema de calificación numérica de 0 a 10 según la legislación vigente (RD 1125/2003)

INFORMÁTICA::PROGRAMACIÓN I (6 ECTS; 150 horas) FORMACIÓN BÁSICA	PROGRAMACIÓN II (6 ECTS; 150 horas) OBLIGATORIA	INFORMÁTICA::ALGORITMOS Y ESTRUCTURAS DE DATOS I (6 ECTS; 150 horas) FORMACIÓN BÁSICA
ALGORITMOS Y ESTRUCTURAS DE DATOS II (6 ECTS; 150 horas) OBLIGATORIA	LÓGICA PARA LA COMPUTACIÓN (6 ECTS; 150 horas) OBLIGATORIA	TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES (6 ECTS; 150 horas) OBLIGATORIA

Denominación de la asignatura		Nº créditos ECTS= 6		
INFORMÁTICA::PROGRAMACIÓN I		carácter = FORMACIÓN BÁSICA		
		Materia = PROGRAMACIÓN		
Ubicación dentro del plan de estudios y duración				
<p>Esta asignatura es obligatoria en el primer semestre del primer curso.</p> <p>En esta asignatura se establecen las bases de la programación estructurada. La base adquirida es imprescindible para poder entender y desarrollar los conocimientos adquiridos en numerosas asignaturas a lo largo de los estudios y su vida profesional.</p> <p>Cualquiera de los tres perfiles profesionales que recogen los ámbitos de actuación más comunes de los Ingeniero/as Técnico/as Informático/as de hoy en día contempla la necesidad de poseer competencias relativas al desarrollo e implementación del software.</p>				
Resultados de aprendizaje				
<ul style="list-style-type: none"> • Adquirir las habilidades básicas para analizar un problema y conseguir desarrollar un programa en un lenguaje de alto nivel que permita solucionarlo. • Adquirir los conocimientos básicos de programación, independientes del lenguaje de programación utilizado. • Adquirir buenos hábitos de programación, primando la sencillez y legibilidad de los programas así como realizando, como paso previo a la programación, un análisis de la solución. • Adquirir un conocimiento detallado y práctico de las características y recursos del lenguaje de programación utilizado en la asignatura. • Conseguir la autonomía del alumno en el análisis y desarrollo de soluciones de cualquier tipo de problema, de complejidad simple a intermedia, de manera que disponga de estas habilidades cuando tenga que programar en cualquier entorno. • Usar las herramientas de un entorno de desarrollo de programación para crear y desarrollar aplicaciones. 				
Requisitos previos				
No se establece ninguno.				
Actividades formativas				
Actividad	ECTS	Metodologías	Competencias Específicas	Competencias Transversales

Clases de aula	1,2	Actividades introductorias Sesión magistral	3,4,5,7,12,13,25,28	I1, I2, I9, I10, P2, S1, S3, S4, S5, S7
Prácticas de laboratorio	2,6	Resolución de problemas y ejercicios	3,4,5,7,12,13,25,28	I1, I2, I9, I10, P2, S1, S3, S4, S5, S7
Trabajos y proyectos	0,7	Trabajos y proyectos	3,4,5,7,12,13,25,28	I2, I5, I8, P3, P5, S3, S5, S6, S7, S9
Evaluación	1,5	Otras (Actividades de recuperación para aquel alumnado que no haya superado la asignatura en la primera opción)	Todas las de la asignatura	Todas las de la asignatura

Procedimientos de evaluación

Procedimiento de evaluación ASISTENTES	Descripción (aspectos/criterios)
Trabajos y proyectos	El alumno tendrá que realizar un proyecto de programación, a partir de un guión proporcionado por el docente
Pruebas prácticas, de ejecución de tareas reales y/o simuladas	Se realizarán varias pruebas parciales, con las que se pretende comprobar si el alumno va alcanzando las competencias básicas de esta asignatura. Además, el alumno realizará distintas entregas correspondientes a resolución de problemas y/o ejercicios.

Procedimiento de evaluación NO ASISTENTES	Descripción (aspectos/criterios)
Resolución de problemas y/o ejercicios	Examen teórico-práctico que recogerá los contenidos correspondientes a la materia impartida a lo largo del semestre.

Breve descripción de contenidos

BLOQUE I.- Fundamentos de informática
BLOQUE II.- Conceptos básicos de programación

BLOQUE III.- Algoritmos y tipos de datos

BLOQUE IV.- Programación modular

BLOQUE V.- Programación estructurada

BLOQUE VI.- Diseño de algoritmos iterativos

BLOQUE VII.- Tipos de datos estructurados

Denominación de la asignatura		Nº créditos ECTS= 6		
PROGRAMACIÓN II		carácter = OBLIGATORIA		
		Materia = PROGRAMACIÓN		
Ubicación dentro del plan de estudios y duración				
La asignatura se enmarca en el segundo semestre del primer curso. El objetivo de ésta es iniciar al alumno en la programación orientada a objetos. Los tres perfiles reconocidos en el Libro Blanco del Grado de Informática precisan de esta asignatura.				
Resultados de aprendizaje				
<ul style="list-style-type: none"> • Conocer ampliamente el lenguaje de programación orientado a objetos de mayor utilidad para la industria en la actualidad. • Conocer ampliamente el proceso de desarrollo asociado a un proyecto de complejidad básica realizado mediante programación orientada a objetos. • Desarrollar software de calidad aplicando los fundamentos del paradigma de orientación a objetos • Dominar la comunicación dentro del grupo de trabajo, y la capacidad de iniciativa y de toma de decisiones en el trabajo realizado. 				
Requisitos previos				
INFORMÁTICA::PROGRAMACIÓN I es requisito previo de PROGRAMACIÓN II .				
Actividades formativas				
actividad	ECTS	metodologías	competencias específicas	competencias transversales
Clases de aula	2,3	Sesión magistral	4,5,7,12,13,14, 25,28	I1, I2, I3, I5, I7, I8,P1, P5,S1, S3, S5, S7
Prácticas de laboratorio Clases prácticas	2,6	Resolución de problemas y/o ejercicios	4,5,7,12,13,14,25,28	I2, I3, I5, I7, I8, I9, I10, P1, P2, P5, S1, S3, S4, S5, S7, S9
Evaluación	1,1	Otras (Actividad de recuperación para los alumnos que no hayan superado la asignatura en la primera	Todas las de la asignatura	Todas las de la asignatura

opción)

Procedimientos de evaluación

Procedimiento de evaluación ASISTENTES	Descripción (aspectos/criterios)
Resolución de problemas y/o ejercicios	Se realizarán tareas continuas en grupo para la resolución de pequeños problemas de programación correspondientes a todos los temas de contenidos de la materia.
Pruebas de respuesta corta	Cada tema tendrá una prueba individual, con la que se pretende comprobar si el alumno ha alcanzado los objetivos de dicho tema.
Pruebas prácticas, de ejecución de tareas y/o simuladas	Se realizará una prueba individual de programación, con la cual se pretende evaluar la capacidad del alumno ante el desarrollo de software de calidad
Procedimiento de evaluación NO ASISTENTES	Descripción (aspectos/criterios)
Pruebas de respuesta corta	Se realizará una prueba individual de cada tema
Pruebas prácticas, de ejecución de tareas y/o simuladas	Se realizará una prueba individual de programación, con la cual se pretende evaluar la capacidad del alumno ante el desarrollo de software de calidad

Breve descripción de contenidos

Teoría

BLOQUE I.- Introducción al Desarrollo Orientado a Objetos.

BLOQUE II.- Fundamentos del modelo orientado a objetos.

BLOQUE III.- Archivos y pruebas.

Práctica

- Clases y objetos. Encapsulación
- Aplicación de herencia y composición.
- Aplicación del polimorfismo.
- Aplicación de genericidad.
- Aplicación de la gestión de errores.

Denominación de la asignatura INFORMÁTICA::ALGORITMOS Y ESTRUCTURAS DE DATOS I	Nº créditos ECTS= 6 carácter = FORMACIÓN BÁSICA Materia = PROGRAMACIÓN
Ubicación dentro del plan de estudios y duración Esta asignatura es obligatoria en el segundo semestre de primer curso. Es una continuación de la asignatura de programación impartida en primer curso. Esta asignatura capacita al alumno para enfrentarse a problemas de programación complejos imprescindibles para cursar las siguientes materias del plan de estudios.	
Resultados de aprendizaje <ul style="list-style-type: none"> • Concebir, desarrollar y utilizar de forma eficiente los tipos de datos y estructuras más adecuados a un problema. • Encontrar soluciones algorítmicas a problemas, comprendiendo la idoneidad y complejidad de las soluciones propuestas • Determinar la complejidad en tiempo y espacio de diferentes algoritmos. • Conocer la recursividad como herramienta de construcción de programas. • Programar aplicaciones de forma robusta, correcta y eficiente teniendo en cuenta restricciones de tiempo y coste, y eligiendo el paradigma y los lenguajes de programación más adecuados. • Conocer nuevas técnicas de programación, en particular el uso de la memoria dinámica y las estructuras de datos enlazadas que están en la base de muchas aplicaciones. • Usar las herramientas de un entorno de desarrollo de programación para crear y desarrollar aplicaciones • Saber analizar, especificar e implementar estructuras de datos lineales desde la perspectiva de los TAD. • Saber resolver problemas utilizando los TAD más apropiados. • Conocer el funcionamiento y las técnicas básicas de ordenación de la información y la consulta eficiente de la misma. 	
Requisitos previos Conocer los fundamentos básicos de programación estructurada. Conocer los conceptos básicos del paradigma de orientación a objetos.	
Actividades formativas	

actividad	ECTS	metodologías	competencias específicas	competencias transversales
Clases de aula	1,8	Sesión magistral	4,5,7,12,13,14,22,25,26,27,28,33	I1, I5, I8, P1, S1, S3, S7
Prácticas de laboratorio	2,6	Prácticas de laboratorio	3,4,5,7,12,13,14,22,25,26,27,28,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Clases prácticas	0,5	Resolución de problemas y/o ejercicios	4,5,7,12,13,14,22,25,26,27,28,33	I1, I5, I8, P1, S1, S3, S7
Evaluación	0,2	Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	3,4,5,7,12,13,14,22,25,26,27,28,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Trabajos y proyectos	0,1	Trabajos y proyectos	3,4,5,7,12,13,14,22,25,26,27,28,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Evaluación	0,8	Otras (Actividad de recuperación para los alumnos que no hayan superado la asignatura en la primera opción)	Todas las de la asignatura	Todas las de la asignatura

Procedimientos de evaluación

Procedimiento de evaluación ASISTENTES	Descripción (aspectos/criterios)
Resolución de problemas y/o ejercicios	Actividad en la que se formulan problemas y/o ejercicios relacionados con los algoritmos y estructuras de datos. El alumno debe desarrollar en Java las soluciones adecuadas y correctas de forma individual.
Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	Actividades de aplicación de los conocimientos a situaciones concretas y de adquisición de habilidades básicas y procedimentales relacionadas con la materia objeto de estudio. El alumno debe desarrollar el análisis y resolución de los problemas de manera individual/grupal.
Trabajos y proyectos	Entrega de actividades de forma individual /grupal. El número de entregas no puede ser inferior al 90% de las totales.
Procedimiento de evaluación NO ASISTENTES	Descripción (aspectos/criterios)

Pruebas de respuesta corta Resolución de problemas y/o ejercicios	Examen de teoría/problemas que recogerá los contenidos correspondientes a la materia impartida durante el semestre. Para calcular la calificación final de la asignatura es necesario que en este examen se obtenga una nota superior o igual a 4
Pruebas prácticas	Examen práctico sobre un computador, que podrá ser sustituido por examen escrito en función de la viabilidad de realizar dicho examen sobre los ordenadores Para calcular la calificación final de la asignatura es necesario que en este examen se obtenga una nota superior o igual a 4
<p>Breve descripción de contenidos</p> <p>BLOQUE I.- Análisis de eficiencia de algoritmos</p> <p>BLOQUE II.- Algoritmos de búsqueda y ordenación</p> <p>BLOQUE III.- Diseño de algoritmos recursivos</p> <p>BLOQUE IV.- Estructuras de datos dinámicas</p> <p>BLOQUE V.- Tipos abstractos de datos</p> <p>BLOQUE VI.- Técnicas de verificación y pruebas</p> <p><i>Prácticas de laboratorio</i></p> <p>Utilizando una herramienta de desarrollo de software se resolverán problemas relacionados con la teoría.</p>	

Denominación de la asignatura		Nº créditos ECTS= 6		
ALGORITMOS Y ESTRUCTURAS DE DATOS II		carácter = OBLIGATORIA		
		Materia = PROGRAMACIÓN		
Ubicación dentro del plan de estudios y duración				
Esta asignatura es obligatoria en el primer semestre de segundo curso. Es una continuación de la asignatura Algoritmos y estructura de datos I y sirve para complementar y ampliar los conocimientos del alumno en el diseño de estructuras de datos y algoritmos para la solución de problemas no triviales de forma eficiente y correcta.				
Resultados de aprendizaje				
<ul style="list-style-type: none"> • Saber analizar, especificar e implementar las estructuras de datos y las colecciones no lineales desde la perspectiva de los TAD. • Saber resolver problemas utilizando la estructura de datos no lineal más apropiada, en función de los recursos necesarios (tiempo de ejecución, espacio requerido, etc.) • Capacitar al alumno para la resolución de problemas utilizando esquemas algorítmicos básicos. • Saber que los esquemas algorítmicos se consideran una metodología en la cual se deben seguir procesos sistemáticos para alcanzar los objetivos de resolución de problemas. • Usar las herramientas de un entorno de desarrollo de programación para crear y desarrollar aplicaciones • Programar aplicaciones de forma robusta, correcta y eficiente teniendo en cuenta restricciones de tiempo y coste, y eligiendo el paradigma y los lenguajes de programación más adecuados. 				
Requisitos previos				
Conocer los conceptos de programación basada en tipos abstractos.				
Actividades formativas				
Actividad	ECTS	metodologías	competencias específicas	competencias transversales

Clases de aula	1,8	Sesión magistral	4,5,7,12,13,14,19,22,25,28,26,27,28,30,32,33	I1, I5, I8, P1, S1, S3, S7
Prácticas de laboratorio	2,5	Prácticas de laboratorio	1,3,4,5,7,12,13,14,19,22,25,26,27,28,30,32,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Clases prácticas	0,4	Resolución de problemas y/o ejercicios	1,3,4,5,7,12,13,14,19,22,25,26,27,28,30,32,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Evaluación	0,3	Prácticas de laboratorio	1,3,4,5,7,12,13,14,19,22,25,26,27,28,30,32,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Trabajos y proyectos	0,1	Trabajos tutelados	1,3,4,5,7,12,13,14,19,22,25,26,27,28,30,32,33	I5, I6, I8, I9, P1, P5, S1, S3, S5, S7, S9
Evaluación	0,9	Otras (Actividad de recuperación para los alumnos que no hayan superado la asignatura en la primera opción)	Todas las de la asignatura	Todas las de la asignatura

Procedimientos de evaluación

Procedimiento de evaluación ASISTENTES	Descripción (aspectos/criterios)
Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	Prueba práctica en la que se resuelven problemas concretos aplicando los conocimientos adquiridos en la asignatura. Se realiza a través de las TIC, empleando el lenguaje Java y de manera grupal.
Resolución de problemas y/o ejercicios	Prueba en la que se formulan problemas/ejercicios relacionados con los contenidos de la asignatura y que el alumno debe resolver de forma individual
Trabajos y proyectos	Entrega de actividades propuestas tanto en las sesiones magistrales como en las prácticas de laboratorio, teniendo que entregar como mínimo el 90% de las actividades
Procedimiento de evaluación NO ASISTENTES	Descripción (aspectos/criterios)
Resolución de problemas y/o ejercicios	Examen de teoría/problemas que recogerá los contenidos correspondientes a la materia impartida durante el semestre.

Pruebas prácticas

Examen práctico sobre un computador, que podrá ser sustituido por examen escrito en función de la viabilidad de realizar dicho examen sobre los ordenadores

Breve descripción de contenidos

BLOQUE I.-Árboles

BLOQUE II.- Mapas y diccionarios

BLOQUE III.-Grafos

BLOQUE IV.- Esquemas algorítmicos

Prácticas de laboratorio

Utilizando una herramienta de desarrollo de software se resolverán problemas relacionados con la teoría.

Denominación de la asignatura		Nº créditos ECTS= 6		
LÓGICA PARA LA COMPUTACIÓN		carácter = OBLIGATORIA		
		Materia = PROGRAMACIÓN		
Ubicación dentro del plan de estudios y duración				
Esta asignatura es obligatoria en el primer semestre de tercer curso. Es la última asignatura dentro de la materia de programación y en ella se hace una condensación de los distintos conceptos y competencias adquiridos en las asignaturas previas.				
Resultados de aprendizaje				
<ul style="list-style-type: none"> • Conocer y comprender los fundamentos y conceptos principales de la programación declarativa y la programación imperativa • Desarrollar programas prototípicos para problemas concretos que requieran el manejo de características propias de cada paradigma • Capacidad de elegir un lenguaje de programación a partir de unos requisitos operativos dados 				
Requisitos previos				
<ul style="list-style-type: none"> • Saber programar en un lenguaje de programación. • Conocer estructuras de datos y abstracciones. • Es recomendable que se curse tras iniciarse en la adquisición de las competencias más básicas como Estructuras de Datos y Programación. 				
Actividades formativas				
actividad	ECTS	Metodologías	competencias específicas	competencias transversales
Clases de aula	2,4	Sesión magistral	3,4,5,7,12,13,14,28,32	I1, I5, I8, S3
Prácticas de laboratorio	2,4	Prácticas de laboratorio	3,4,5,7,12,13,14,28,32	I1, I5, I8, S3
Evaluación	1,2	Otras (Actividad de recuperación para los alumnos que no hayan superado la asignatura en la primera opción)	Todas las de la asignatura	Todas las de la asignatura
Procedimientos de evaluación				
Procedimiento de evaluación ASISTENTES		Descripción (aspectos/criterios)		

Prácticas de laboratorio	Los alumnos deberán realizar una defensa de las prácticas realizadas, consistente en una prueba de funcionamiento y en la contestación de las preguntas realizadas por el profesor, con el objetivo de comprobar lo aprendido por los alumnos durante la realización del trabajo.
Pruebas de respuesta larga, de desarrollo	Al final del curso se realizará una prueba escrita en donde se examinará a los alumnos sobre los conocimientos adquiridos en las clases teóricas.

Procedimientos de evaluación para NO ASISTENTES: El procedimiento de evaluación para no asistentes será el mismo que para asistentes.

Breve descripción de contenidos

- BLOQUE I.- Paradigma Imperativo
- BLOQUE II.- Paradigma Lógico
- BLOQUE III.- Paradigma Funcional

<p>Denominación de la asignatura</p> <p>TEORÍA DE AUTÓMATAS Y LENGUAJES FORMALES</p>	<p>Nº créditos ECTS= 6 carácter = OBLIGATORIA Materia = PROGRAMACIÓN</p>
<p>Duración y ubicación temporal dentro del plan de estudios</p> <p>La asignatura está ubicada en el 2º semestre del 3º curso del plan de estudios. Su ubicación se razona por mover la carga teórica de los estudios en el ámbito de Ciencia de la Computación al momento en el cual principales capacidades, sobre todo la programación práctica, ya están adquiridos por los estudiantes</p> <p>En esta asignatura se incluyen competencias básicas e imprescindibles para el futuro ejercicio profesional del Ingeniero/a Técnico/a en Informática, para entender, colaborar, avisar, y decidir en el complejo ámbito del diseño de software, entre otros la comprensión del funcionamiento de compiladores, transductores, y el proceso de especificación formal de información.</p>	
<p>Resultados de aprendizaje</p> <ul style="list-style-type: none"> • Conocer los elementos básicos de la teoría de lenguajes formales y, sus propiedades y como se combinan para generar los diferentes tipos de autómatas y lenguajes • Conocer la jerarquía de Chomsky de lenguajes formales y saber relacionar sus categorías con la clase de autómata que la reconoce • Conocer la definición y propiedades fundamentales de las máquinas de estado finito y los autómatas con pila • Capacidad para implementar las diferentes técnicas de construcción de autómatas para el análisis de lenguajes formales en los niveles léxico y sintáctico • Capacidad para usar herramientas de generación de analizadores léxicos y sintácticos basadas en algoritmos de construcción de autómatas 	
<p>Requisitos previos</p> <ul style="list-style-type: none"> • Conocimientos de estructuras de datos y algoritmos básicos y de su análisis • Conocimiento de entornos y herramientas de desarrollo de software • Conocimiento de conceptos básicos de la programación • Conocimientos principales de la matemática • Capacidad de buscar información por medios propios 	

Actividades formativas

actividad	ECTS	Metodologías	competencias específicas	competencias transversales
Clases de aula	2,4	Sesión magistral	3,4,5,7,12,13,14,28	I1, I5, I7, I8, S3
Prácticas de laboratorio	2,4	Prácticas de laboratorio	3,4,5,7,12,13,14,28	I1, I5, I7, I8, S3
Evaluación	1,2	Otras (Actividad de recuperación para los alumnos que no hayan superado la asignatura en la primera opción)	Todas las de la asignatura	Todas las de la asignatura

Procedimientos de evaluación

Procedimiento de evaluación ASISTENTES	Descripción (aspectos/criterios)
Prácticas de laboratorio	Los alumnos deberán realizar una defensa de las prácticas realizadas, consistente en una prueba de funcionamiento y en la contestación de las preguntas realizadas por el profesor, con el objetivo de comprobar lo aprendido por los alumnos durante la realización del trabajo. La nota final dependerá de la calidad del trabajo realizado y de la defensa realizada por los alumnos
Pruebas de respuesta larga, de desarrollo	Al final del curso se realizará una prueba en donde se examinará a los alumnos sobre los conocimientos adquiridos en las clases teóricas

Procedimientos de evaluación para NO ASISTENTES: Los procedimientos de evaluación para no asistentes serán los mismos que para asistentes.

Breve descripción de contenidos**BLOQUE I: AUTÓMATAS Y LENGUAJES FORMALES**

Tema 1.- Conceptos fundamentales: Alfabetos, gramáticas, lenguajes, derivaciones. Lema Fundamental. Jerarquía de Chomsky.

Tema 2.- Lenguajes Regulares: Gramáticas regulares. Expresiones regulares. Propiedades. Autómatas finitos.

Tema 3.- Lenguajes independientes del contexto: Gramáticas independientes del contexto. Árboles de derivación. Ambigüedad. Propiedades. Autómatas de pila

BLOQUE II: PROCESADORES DEL LENGUAJE

Tema 4.- Análisis léxico: Técnicas de generación de autómatas finitos.

Tema 5.- Análisis sintáctico mixto: Familia de técnicas LR.