

**FICHA IDENTIFICATIVA****Datos de la Asignatura**

<b>Código</b>	34656
<b>Nombre</b>	Programación
<b>Ciclo</b>	Grado
<b>Créditos ECTS</b>	6.0
<b>Curso académico</b>	2022 - 2023

**Titulación(es)**

<b>Titulación</b>	<b>Centro</b>	<b>Curso</b>	<b>Periodo</b>
1400 - Grado en Ingeniería Informática	Escuela Técnica Superior de Ingeniería	1	Segundo cuatrimestre

**Materias**

<b>Titulación</b>	<b>Materia</b>	<b>Carácter</b>
1400 - Grado en Ingeniería Informática	5 - Informática	Formación Básica

**Coordinación**

<b>Nombre</b>	<b>Departamento</b>
ALBERT BLANCO, JESUS V.	240 - Informática

**RESUMEN**

La asignatura “Programación” es una asignatura del primer curso del Grado de Ingeniería Informática, que cubre una parte de la materia básica *Informática*.

En esta asignatura se profundiza en los conocimientos y habilidades de la programación en C++ vistos en la asignatura “Informática”, de la que puede considerarse una continuación. Las líneas básicas de la asignatura se articulan alrededor de la Programación Orientada a Objetos y los Tipos Abstractos de Datos y sus diversas interrelaciones. También se estudiará con cierto detalle el análisis del coste temporal de los algoritmos, lo que permitirá al alumno decidir el algoritmo más apropiado para cada problema concreto.

El profesorado de esta asignatura es miembro del Grupo Consolidado de Innovación Docente en Metodologías Docentes Colaborativas, Cooperativas y Competitivas, y participa en la propuesta de Red de Innovación Docente con referencia UV-SFPIE\_FO13-147196.



## CONOCIMIENTOS PREVIOS

### Relación con otras asignaturas de la misma titulación

No se han especificado restricciones de matrícula con otras asignaturas del plan de estudios.

### Otros tipos de requisitos

Es muy conveniente que los alumnos hayan cursado la asignatura Informática.

Los conocimientos y habilidades previas que se requieren en esta asignatura son los siguientes:

- Analizar problemas sencillos, diseñar y preparar algoritmos para resolverlos mediante la utilización del ordenador.
- Tipos de datos, variables, constantes, estructuras de control y estructuras de datos básicas que tienen los lenguajes de programación procedurales para desarrollar programas.

## COMPETENCIAS (RD 1393/2007) // RESULTADOS DEL APRENDIZAJE (RD 822/2021)

### 1400 - Grado en Ingeniería Informática

- G8 - Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
- G9 - Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
- B3 - 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.
- B1 - 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.
- B4 - Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

## RESULTADOS DE APRENDIZAJE (RD 1393/2007) // SIN CONTENIDO (RD 822/2021)

La asignatura, de acuerdo con la memoria de verificación, proporciona los siguientes resultados de aprendizaje:



- Realizar operaciones básicas sobre ficheros
- Describir algorítmicamente soluciones a problemas
- Capacidad para utilizar un lenguaje de programación para describir el algoritmo que resuelve un problema
- Describir los tipos de datos básicos, numéricos y no numéricos
- Diseñar programas de ordenador sencillos con uno o varios bucles
- Diseñar programas de ordenador sencillos estructurados mediante funciones
- Diseñar programas de ordenador sencillos utilizando estructuras condicionales
- Documentar adecuadamente los programas construidos
- Describir la representación interna de los datos no numéricos.
- Trabajar en equipo para realizar los diseños y configuraciones necesarias, repartiendo la carga de trabajo para afrontar problemas complejos.

Adicionalmente, se adquirirán las siguientes destrezas:

- Calcular el coste temporal teórico de un algoritmo. Expresar el coste usando notación asintótica.
- Usar clases, herencia y sobrecarga de operadores en la implementación de programas.
- Decidir el tipo abstracto de datos más adecuado para un problema concreto, distinguiendo entre vectores, pilas, colas y listas.
- Usar la implementación más adecuada para un TAD concreto, distinguiendo especialmente entre implementaciones estáticas y dinámicas.
- Proponer soluciones a problemas de programación usando una metodología de programación orientada a objetos con C++.

## DESCRIPCIÓN DE CONTENIDOS

### 1. Introducción al estudio de los algoritmos y su complejidad

- Definición de complejidad y su medida.
- Recuperación de información: Búsqueda.
- El problema de la ordenación. Métodos de ordenación interna

### 2. Tipos abstractos de datos

- Tipos de datos.
- Estructuras de datos.
- Tipos Abstractos de datos.



### 3. Programación orientada a objetos

- Clases.
- Sobrecarga.
- Herencia.
- Introducción a las plantillas. Standard Template Library (STL).

### 4. Pilas

- Fundamentos y definición del TAD Pila.
- Representación estática.
- Representación dinámica. Tipo de dato puntero.
- Representación mediante STL.
- Aplicaciones.

### 5. Colas

- Fundamentos y definición del TAD Cola.
- Representación estática y dinámica.
- Representación mediante STL.
- Aplicaciones.

### 6. Listas

- Definición del tipo Lista con punto de interés.
- Representación estática y dinámica.
- Mejoras en la representación de listas enlazadas.
- Iteradores.
- Representación mediante STL.
- Aplicaciones.

**VOLUMEN DE TRABAJO**

ACTIVIDAD	Horas	% Presencial
Clases de teoría	30,00	100
Prácticas en laboratorio	20,00	100
Prácticas en aula	10,00	100
Elaboración de trabajos en grupo	14,00	0
Elaboración de trabajos individuales	15,00	0
Preparación de actividades de evaluación	15,00	0
Preparación de clases de teoría	26,00	0
Preparación de clases prácticas y de problemas	20,00	0
<b>TOTAL</b>	<b>150,00</b>	

**METODOLOGÍA DOCENTE**

En las actividades teóricas de carácter presencial se desarrollarán los temas de la asignatura proporcionando una visión global e integradora, analizando con mayor detalle los aspectos clave y de mayor complejidad, fomentando, en todo momento, la participación del alumnado. Estas actividades se complementan con actividades prácticas con el objetivo de aplicar los conceptos básicos y ampliarlos con el conocimiento y la experiencia que se vayan adquiriendo durante la realización de los trabajos propuestos. Comprenden los siguientes tipos de actividades presenciales:

- Clases de problemas y cuestiones en aula
- Sesiones de discusión y resolución de problemas y ejercicios previamente trabajados por el alumnado
- Prácticas de laboratorio
- Realización de cuestionarios individuales de evaluación en el aula con la presencia del profesorado.

Además de las actividades presenciales, los estudiantes deberán realizar tareas personales (fuera del aula) sobre: búsqueda bibliográfica dirigida, cuestiones y problemas, así como la preparación de clases y exámenes. Estas tareas se realizarán principalmente de manera individual, con objeto de potenciar el trabajo autónomo, pero adicionalmente se incluirán trabajos que requieran la participación de pequeños grupos de estudiantes (2-4) para fomentar la capacidad de integración en grupos de trabajo.

Se utilizará la plataforma de e-learning (Aula Virtual) de la Universitat de València como soporte de comunicación con el alumnado. A través de ella se tendrá acceso al material didáctico utilizado en clase, así como los problemas y ejercicios a resolver.



## EVALUACIÓN

La asignatura se evaluará mediante:

SE1 - Prueba objetiva, consistente en uno o varios exámenes que constan tanto de cuestiones teórico-prácticas como de problemas.

SE2 - Evaluación de las actividades prácticas a partir de la elaboración de trabajos/memorias y/o exposiciones orales.

SE3 - Evaluación continua de cada alumno, basada en la participación y grado de implicación del alumno en el proceso de enseñanza-aprendizaje, teniendo en cuenta la asistencia regular a las actividades presenciales previstas y la resolución de cuestiones y problemas propuestos periódicamente.

En cada una de estas pruebas se tendrán en cuenta las siguientes consideraciones:

1) SE1: Se realizarán diversas pruebas individuales a lo largo del curso, que constarán tanto de cuestiones teórico-prácticas como de problemas (evaluación de competencias CB02, CB04, CG01, CG06, CT03, CT05, CE02, CE06). Habrá dos tipos de pruebas con el siguiente peso:

SE1a (70%): Examen de la asignatura al finalizar la docencia.

SE1b (30%): Controles intermedios realizados durante el periodo docente.

2) SE2: Evaluación de las actividades prácticas realizadas tanto en los laboratorios, como en ejercicios escritos (evaluación de competencias CB02, CB04, CG01, CG06, CT03, CT05, CE02, CE06). Estas pruebas se realizarán en grupos de 2 personas y contemplan las siguientes actividades:

SE2a (70%): Evaluación de prácticas de laboratorio a partir de la documentación (y con los plazos) exigida en cada una de ellas.

SE2b (30%): Realización de ejercicios prácticos por escrito en controles intermedios realizados durante el periodo docente.

3) SE3: Evaluación continua de cada estudiante para medir su grado de participación e implicación en las actividades presenciales. Se considerarán los siguientes aspectos (evaluación de competencias CB02, CB04, CG01, CT03): Resolución de ejercicios propuestos durante el periodo docente; Resolución pública de cuestiones y problemas discutidos en clase; Participación activa en las actividades propuestas.

La nota final de la asignatura se calculará como la media ponderada de los apartados SE1 (60%) y SE2 (40%). El criterio SE3 tendrá la consideración de puntuación extra sobre la nota anterior, pero solo si esta fuera mayor o igual que 4,5. Además, el incremento estará limitado a un máximo del 10% de la calificación obtenida a partir de SE1 y SE2.



Consideraciones particulares sobre la evaluación:

1) Apartados no recuperables: Los criterios que evalúan el seguimiento de la asignatura durante el periodo lectivo no son recuperables posteriormente. Esto son: SE1b, SE2b y SE3. El criterio SE2a será recuperable, solo en la 2ª convocatoria, mediante un examen práctico individual realizado en laboratorio en condiciones equivalentes a las de una práctica de laboratorio, pero con una limitación de tiempo y de acceso a materiales de apoyo.

2) Apartados que requieren nota mínima: Se requiere obtener una nota mínima de 3 (sobre 10) en cada uno de los siguientes apartados de evaluación para poder aprobar la asignatura: SE1a y SE2a.

3) Estarán eximidos de la realización de la prueba SE1a (examen final) aquellos estudiantes que hayan realizado todos los controles periódicos de la asignatura (SE1b, SE2b) y cuya nota media ponderada en estos controles (SE1b (70%), SE2b (30%)) sea mayor o igual que 5. Adicionalmente, será necesario haber obtenido una calificación superior o igual a 3 en todos los controles (tanto en los individuales como en los de parejas). En estos casos, se asignará como calificación del apartado SE1a la nota media ponderada de los controles periódicos intermedios.

En cualquier caso, la evaluación de la asignatura se hará de acuerdo con el Reglamento de evaluación y calificación de la Universitat de València para los títulos de grado y master aprobado por Consejo de Gobierno de 30 de mayo de 2017 (ACGUV 108/2017)

## REFERENCIAS

### Básicas

- TADs Estructuras de datos y resolución de problemas con C++ (2ª Ed.)  
L.R. Nyhoff. Prentice Hall, 2005
- Resolución de problemas con C++ (5ª Ed.)  
W. Savitch. Prentice Hall, 2007
- Cómo programar en C++ (6ª Ed.)  
H.M. Deitel, P.J. Deitel, P.J. . Prentice Hall, 2009

### Complementarias

- C++ plus data structures  
N. Dale, C. Weems, T. Richards. Burlington, MA: Jones & Bartlett Learning, 2016  
<https://ebookcentral.proquest.com/lib/univalencia/detail.action?docID=4714314>



- 
- C++ Cómo programar, Novena Edición  
H.M. Deitel, P.J. Deitel, P.J. Prentice Hall, 2014  
[http://www.ingebook.com/ib/NPcd/IB\\_BooksVis?cod\\_primaria=1000187&codigo\\_libro=6053](http://www.ingebook.com/ib/NPcd/IB_BooksVis?cod_primaria=1000187&codigo_libro=6053)
- 

