



Guía docente de la asignatura

Asignatura	PROGRAMACIÓN		
Materia	INFORMÁTICA		
Módulo	MATERIAS INSTRUMENTALES		
Titulación	GRADO EN INGENIERIA DE TECNOLOGÍAS ESPECÍFICAS DE TELECOMUNICACIÓN		
Plan	564	Código	45004
Periodo de impartición	1 ^{er} . CUATRIMESTRE	Tipo/Carácter	FORMACIÓN BÁSICA
Nivel/Ciclo	GRADO	Curso	1º
Créditos ECTS	6		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	JOSÉ FERNANDO DÍEZ HIGUERA DAVID GONZÁLEZ ORTEGA		
Datos de contacto (E-mail, teléfono...)	JOSÉ FERNANDO DÍEZ HIGUERA DESPACHO: 2D079 TELÉFONO: 983423000 ext. 5562 E-MAIL: josdie@tel.uva.es DAVID GONZÁLEZ ORTEGA DESPACHO: 2D022 TELÉFONO: 983423000 ext. 5552 E-MAIL: davgon@tel.uva.es		
Horario de tutorías	Ver Tutorías en http://www.uva.es/export/sites/uva/2.docencia/2.01.grados/2.01.02.ofertaformativagrados/2.01.02.01.alfabetica/Grado-en-Ingenieria-de-Tecnologias-de-Telecomunicacion/		
Departamento	TEORÍA DE LA SEÑAL Y COMUNICACIONES E INGENIERÍA TELEMÁTICA		



1. Situación / Sentido de la Asignatura

1.1 Contextualización

La asignatura *Programación* pertenece a los planes de estudios de los Grados que se imparten en la Escuela Técnica Superior de Ingenieros de Telecomunicación, siendo la primera asignatura de naturaleza informática a la que se enfrentan los alumnos de dichos Grados.

Esta asignatura es una experiencia de aprendizaje diseñada para enseñar a los alumnos cómo desarrollar programas que puedan resolver problemas, convertir datos, almacenar y recuperar información, ayudar a las personas a comunicarse o hacer cualquier cosa que el programador pueda imaginar. En el fondo, esto se hace cuando el programador escribe las instrucciones para el ordenador en un "lenguaje de programación". En otras palabras, los programadores de ordenadores actúan como traductores entre personas y ordenadores, escribiendo las especificaciones de un programa deseado en un lenguaje que el ordenador pueda entender.

Por último, no hay que perder de vista que, una vez cursada esta asignatura, el alumno debe disponer de una herramienta que tendrá que utilizar en otras asignaturas del Grado y, posiblemente, en su vida profesional.

1.2 Relación con otras materias

Esta asignatura está especialmente relacionada con la asignatura "Fundamentos de Ordenadores y Sistemas Operativos", la cual incluye programación en lenguaje de bajo nivel (ensamblador) y en lenguaje de alto nivel con llamadas al sistema operativo, y con la asignatura "Ingeniería de Sistemas Software", en la que se aplican las técnicas y procedimientos de una metodología de desarrollo software concreta al análisis y diseño de un sistema software en el ámbito de los servicios de telecomunicación.

1.3 Prerrequisitos

El alumno que curse esta asignatura ha de poseer unos conocimientos básicos de informática a nivel usuario. En lo referente a programación, y siendo la primera asignatura de los Grados que aborda dicha materia, se parte de la base de que el alumno no tiene conocimientos previos de la misma.



2. Competencias

2.1 Generales

- GB1. Capacidad de razonamiento, análisis y síntesis.
- GB2. Capacidad para relacionar conceptos y adquirir una visión integrada, evitando enfoques fragmentarios.
- GB3. Capacidad de toma de decisiones en la resolución de problemas básicos de ingeniería de telecomunicación, así como identificación y formulación de los mismos.
- GB4. Capacidad para trabajar en grupo, participando de forma activa, colaborando con sus compañeros y trabajando de forma orientada al resultado conjunto, y en un entorno multilingüe.
- GB5. Conocimiento de materias básicas, científicas y tecnologías, que le capacite para el aprendizaje de nuevos métodos y tecnologías.
- GBE3. Capacidad para resolver problemas con iniciativa, creatividad y razonamiento crítico.
- GC1. Capacidad de organización, planificación y gestión del tiempo.
- GC2. Capacidad para comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
- GC3. Trabajar en cualquier contexto, individual o en grupo, de aprendizaje o profesional, local o internacional, desde el respeto a los derechos fundamentales, de igualdad de sexo, raza o religión y los principios de accesibilidad universal, así como la cultura de paz.

2.2 Específicas

- B2. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.



3. Objetivos

Al finalizar la asignatura el alumno deberá ser capaz de:

- Conocer los conceptos relacionados con la programación.
- Aplicar las técnicas y procedimientos de una metodología de programación de un sistema software.
- Codificar y probar dicho sistema, aplicando técnicas de programación orientada a procesos y a datos.
- Codificar, poner a punto y ejecutar programas sencillos en lenguaje C.
- Diseñar algoritmos sencillos basados en los esquemas de recorrido y búsqueda.
- Autoevaluar el trabajo realizado e identificar los propios errores y aspectos a mejorar.
- Aprender de forma autónoma:
 - Localizar y asimilar una determinada información a partir de su referencia.
 - Autoevaluarse o evaluar a otros a partir de unos criterios dados (Evaluación).
 - Identificar los propios errores.
 - Buscar información relevante para una tarea especificada.



4. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	20	Estudio y trabajo autónomo individual	90
Clases prácticas de aula (A)	0		
Laboratorios (L)	40		
Prácticas externas, clínicas o de campo	0		
Seminarios (S)	0		
Tutorías grupales (TG)			
Evaluación			
Total presencial	60	Total no presencial	90



5. Niveles de Aprendizaje

En el presente curso, la asignatura se articula en tres niveles de aprendizaje, a semejanza de los cursos de idiomas, ya que el objetivo principal es el aprendizaje de un lenguaje (de programación): *Nivel Básico*, *Nivel Intermedio* y *Nivel Avanzado*. Con este esquema se pretende que sea el alumno el que decida los conocimientos y habilidades que quiere aprender sobre programación de ordenadores y/o la calificación que aspira conseguir para su expediente. En este sentido, las calificaciones que se pueden obtener superando las pruebas de cada uno de los niveles, y la duración de cada nivel, se muestran en la siguiente tabla:

Nivel	Duración	Calificación
Básico	8 semanas	5.0
Intermedio	4 semanas	(5.0 - 8.0]
Avanzado	3 semanas	(8.0 - 10.0]

- Al finalizar las semanas correspondientes al nivel básico, se publicarán las calificaciones obtenidas en el nivel básico. Aquellos alumnos que superen este nivel **ya tienen un 5.0 como nota de la asignatura**. A partir de ese momento, será potestad de cada alumno decidir si quiere continuar con el nivel intermedio. En caso contrario, la nota definitiva será de 5.0 puntos (aprobado).
- Al finalizar las semanas correspondientes al nivel intermedio, se publicarán las calificaciones de aquellos alumnos que hayan decidido cursar el nivel intermedio. Aquellos alumnos que superen este nivel **podrán optar a una calificación entre 5.1 y 8.0**, en función de los resultados de la evaluación de este nivel. A partir de ese momento, será potestad de cada alumno decidir si quiere continuar con el nivel avanzado. En caso contrario, la nota definitiva será la obtenida en las pruebas de este nivel, y siempre en el intervalo (5.0, 8.0].
- Por último, al finalizar las semanas correspondientes al nivel avanzado, se publicarán las calificaciones de aquellos alumnos que hayan decidido cursar el nivel avanzado. Aquellos alumnos que superen este nivel **podrán optar a una calificación entre 8.1 y 10.0**, en función de los resultados de la evaluación de este nivel.

5. Bloques temáticos

En consonancia con el apartado anterior, los conocimientos y habilidades que el alumno debe adquirir, se articulan en tres bloques temáticos, correspondientes a los tres niveles de aprendizaje.

Los objetivos de aprendizaje describen en detalle todo lo que el alumno va a aprender durante este curso. Es importante que el alumno los tenga presentes desde el primer momento, aunque el profesor le irá recordando los objetivos que están implicados en las diferentes actividades del curso.

5.1. Características comunes a todos los bloques

a. Métodos docentes



- Clase expositiva participativa.
- Resolución de problemas.
- Desarrollo del Proyecto en el laboratorio.
- Aprendizaje basado en proyecto.

b. Plan de trabajo

El Plan de Trabajo de la asignatura, donde se describe la planificación detallada, incluyendo fechas de las prácticas por grupo, se entregará al comienzo de la asignatura.

c. Evaluación

- Pruebas de conocimientos básicos.
- Evaluación de la versión final del Proyecto correspondiente a cada nivel.
- Coevaluación de la versión del Proyecto correspondiente a cada nivel.

d. Bibliografía básica

- Apuntes de la asignatura.
- Ver Anexo I (bibliografía)

e. Bibliografía complementaria

- Ver **Anexo I** (bibliografía)

i. Recursos necesarios

- Aula con proyector multimedia y pizarra para sesiones de discusión.
- Laboratorio de prácticas, con un ordenador por alumno, para las sesiones de laboratorio. Cada ordenador debe contar con los entornos de desarrollo para el lenguaje C que se van a utilizar.
- Plataforma educativa para publicar material didáctico, guías de ejercicios, soluciones, tareas, etc.
- Foro de discusión en línea para atender preguntas y discutir aspectos relacionados con el aprendizaje de la asignatura.
- Acceso al material bibliográfico recomendado.



5.3. Desglose por bloques

Bloque 1: Nivel Básico

Carga de trabajo en créditos ECTS: 3.2

a. Contextualización y justificación

En este primer bloque se introducen los conceptos básicos sobre los diferentes métodos para el desarrollo de aplicaciones informáticas, se presenta C, el lenguaje de programación que se utilizará para la implantación de los programas desarrollados, y se proporcionan los elementos básicos de la programación estructurada. El bloque se completa con casos de estudio y con la aplicación al proyecto integrador de los métodos aprendidos.

b. Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación con los **tipos de datos** elementales y sus **operaciones** (enteros, caracteres y reales):
 - Escribir los tipos de datos elementales y las operaciones que actúan sobre ellos.
 - Escribir la declaración de datos de cualquiera de los tipos elementales.
 - Indicar el código ASCII de cualquier carácter, con la ayuda de la tabla correspondiente.
- En relación con las **sentencias básicas de asignación y de entrada/salida**:
 - Describir el funcionamiento de las sentencias básicas.
 - Predecir el resultado de una secuencia de sentencias básicas.
 - Codificar una tarea convenientemente especificada, utilizando la secuencia de sentencias básicas adecuada.
- En relación con los **archivos**:
 - Explicar el concepto de archivo, para qué sirve, y cuáles son las operaciones típicas sobre archivos de texto (crear, abrir, leer, escribir, preguntar por fin de archivo y cerrar).
 - Escribir las sentencias necesarias para realizar las operaciones básicas con un archivo de texto.
 - Escribir las sentencias necesarias para determinar el tipo de error que se ha producido al realizar una operación con un archivo de texto.
- En relación con las **sentencias de control**:
 - Conocer y usar los operadores que se utilizan para construir expresiones lógicas.
 - Usar las sentencias de selección `if, if...else` para elegir entre varias acciones alternativas.
 - Utilizar la sentencia de selección múltiple `switch` para escoger entre muchas alternativas de acción.
 - Utilizar las etiquetas `case` para identificar las acciones alternativas en las sentencias `switch`.



- Ejecutar sentencias repetidamente con la sentencia repetitiva `for`.
- Usar la sentencia de repetición `while` para ejecutar sentencias repetidamente.
- Usar la sentencia de repetición `do while` para ejecutar sentencias repetidamente.
- En relación con las funciones:
 - Definir el concepto de función y su utilidad.
 - Definir los conceptos: cabecera de función, parámetros formales, variables locales, resultado de la función, llamada a función, parámetros reales, paso de parámetros.
 - Describir la diferencia entre paso de parámetros por valor o por referencia.
 - Codificar convenientemente una llamada a función, pasando correctamente los parámetros.
 - Codificar en forma de función una tarea convenientemente especificada, estableciendo adecuadamente los parámetros necesarios.
 - Proponer una organización en bloques (funciones) de una aplicación determinada.
- En relación con el **entorno de desarrollo**:
 - Definir los conceptos de compilación, construcción y ejecución.
 - Realizar las operaciones necesarias para crear/abrir un proyecto y añadir y eliminar elementos a un proyecto.
 - Realizar las operaciones necesarias para editar, compilar, montar y ejecutar un programa, y localizar las carpetas donde están los archivos generados en cada uno de los pasos.
 - Diseñar archivos *makefile*.
 - Interpretar adecuadamente los mensajes de error de compilación y corregir el error de compilación correspondiente.
 - Describir las funcionalidades básicas del depurador.
 - Realizar correctamente las operaciones básicas del depurador (insertar un punto de parada, ejecutar paso a paso y visualizar valores de variables).
 - Identificar y subsanar los errores de ejecución de un programa, utilizando adecuadamente el depurador.

c. Contenidos

- Herramientas de desarrollo.
- Conceptos básicos del lenguaje C.
- Funciones de interacción con el usuario y de acceso a archivos de texto.
- Entornos de desarrollo. Depuradores.
- Sentencias de control.



- Vectores numéricos y cadenas de caracteres.

Bloque 2: Nivel Intermedio

Carga de trabajo en créditos ECTS:

1.6

a. Contextualización y justificación

El bloque 2 se centra en la introducción de los conceptos de puntero, paso de parámetros por dirección a una función, tipos de datos estructurados, registros, archivos binarios, vectores multidimensionales y vectores de registros. El bloque se completa con casos de estudio y con la aplicación al proyecto integrador de los métodos aprendidos.

b. Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación a los **punteros**:
 - Comprender el concepto de puntero y sus diferencias con una variable simple.
 - Escribir el código necesario para acceder al dato apuntado por un puntero.
 - Escribir el código necesario para recorrer un vector o cadena de caracteres utilizando un puntero.
 - Implantar el paso de parámetros por dirección a una función utilizando punteros.
- En relación con los **tipos de datos estructurados**:
 - Describir las estructuras de datos fundamentales, y las operaciones típicas sobre ellas.
 - Escribir la declaración de una estructura de datos convenientemente especificada.
 - Escribir el código necesario para acceder a un elemento o conjunto de elementos de una estructura de datos.
 - Elegir la estructura de datos más adecuada para una aplicación determinada.
- En relación con los **archivos binarios**:
 - Recordar el concepto de archivo, para qué sirve, y cuáles son las operaciones típicas sobre archivos (crear, abrir, leer, escribir, preguntar por fin de archivo y cerrar).
 - Escribir las sentencias necesarias para realizar las operaciones básicas con archivos binarios.
 - Escribir las sentencias necesarias para determinar el tipo de error que se ha producido al realizar una operación con un archivo binario.

c. Contenidos

- Punteros.
- Paso de parámetros por dirección.
- Archivos binarios



- Vectores multidimensionales.
- Registros y Vectores de Registros.

Bloque 3: Nivel Avanzado

Carga de trabajo en créditos ECTS:

1.2

a. Contextualización y justificación

El bloque 3 se centra en las estructuras de datos dinámicas, en la asignación dinámica de memoria, y en la introducción a bases de datos. El bloque se completa con casos de estudio y con la aplicación al proyecto integrador de los nuevos conceptos aprendidos.

b. Objetivos de aprendizaje

Al final de este bloque el alumno será capaz de:

- En relación con la **asignación dinámica de memoria**:
 - Asignar y liberar memoria dinámicamente para diferentes tipos de datos.
 - Crear y redimensionar vectores dinámicos.
- En relación con los **vectores dinámicos**:
 - Escribir la declaración de un vector dinámico.
 - Escribir el código necesario para asignar memoria a un vector dinámico.
 - Escribir el código necesario para reasignar memoria a un vector dinámico.
 - Escribir el código necesario para liberar la memoria asignada un vector dinámico.
- En relación a las **bases de datos**:
 - Entender el concepto de base de datos, y cómo se estructura.
 - Escribir las sentencias necesarias para crear, abrir y cerrar una base de datos sencilla (*SQLite*).
 - Escribir las sentencias necesarias para realizar las operaciones básicas sobre una base de datos (crear tablas, insertar datos, seleccionar datos).

c. Contenidos

- Asignación dinámica.
- Vectores dinámicos.
- Introducción a Bases de datos.



6. Temporalización (por bloques temáticos)

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 1: Nivel Básico	3.2	Semanas 1 a 8
Bloque 2: Nivel Intermedio	1.6	Semanas 9 a 12
Bloque 3: Nivel Avanzado	1.2	Semanas 13 a 15

7. Elementos de Evaluación

7.1. Cuestionarios de Conocimientos Básicos

Esta asignatura dispone de un banco de preguntas, residente en el Campus Virtual. A partir de este banco de preguntas se elaboran una serie de 8 cuestionarios de Conocimientos Básicos: 6 para el Nivel Básico, 1 para el Nivel Intermedio y 1 para el Nivel Avanzado. Los alumnos pueden acceder a y practicar con las preguntas de este banco de preguntas a través de los cuestionarios de aprendizaje disponibles durante todo el curso.

Cada cuestionario de Conocimientos Básicos consta de 10 preguntas, seleccionadas aleatoriamente del banco de preguntas. Para superar un cuestionario de conocimientos básicos, el alumno debe responder correctamente al menos a 8 de las 10 preguntas.

Para superar cada cuestionario, el alumno dispone de cuatro intentos durante el curso, más dos intentos adicionales, uno al final de la convocatoria ordinaria y otro en la convocatoria extraordinaria.

- **Nivel Básico:** para promediar con el resto de elementos de evaluación, el alumno debe responder correctamente al menos a 8 de las 10 preguntas de los 6 primeros cuestionarios.
- **Nivel Intermedio:** solamente incluye un cuestionario de Conocimientos Básicos. Para promediar con el resto de elementos de evaluación, el alumno debe responder correctamente al menos a 8 de las 10 preguntas de los 7 primeros cuestionarios: los 6 del nivel básico, más el correspondiente a este nivel.
- **Nivel Avanzado:** solamente incluye un cuestionario de Conocimientos Básicos. Para promediar con el resto de elementos de evaluación, el alumno debe responder correctamente al menos 8 de las 10 preguntas de los 8 cuestionarios: los 6 del nivel básico, más el del nivel intermedio y el correspondiente a este nivel.

7.2. Proyecto Integrador

A lo largo del curso, y durante las horas presenciales en laboratorio, el alumno desarrollará un proyecto informático que se irá actualizando con la incorporación de los conceptos aprendidos en cada semana. En la versión de cada nivel se incorporarán nuevos elementos del programación vistos en las semanas correspondientes.

En la última semana de cada nivel (semanas 8, 12 y 15, respectivamente), el alumno entregará la versión desarrollada en el buzón correspondiente del Campus Virtual, para que dicho envío sea coevaluado por al menos tres alumnos de forma anónima. Esta evaluación será revisada por el profesor previa petición de los alumnos que consideren, razonadamente, que su evaluación no es correcta, o en el caso de que se detecten serias discrepancias en las calificaciones de los evaluadores.

La nota final del proyecto será, en cada nivel, la obtenida en el proceso de coevaluación.



Convocatorias para cada nivel:

- El alumno que opte por el **nivel básico** y no apruebe el proyecto en la entrega de la semana 8, podrán completarlo y/o modificarlo en las siguientes sesiones de laboratorio, y volver a intentarlo en la entregas de las semanas 12 y 15, en la de la convocatoria ordinaria, y en la de la convocatoria extraordinaria. En total, 5 intentos.
- El alumno que opte por el **nivel intermedio** y no apruebe el proyecto en la entrega de la semana 12, podrán completarlo y/o modificarlo en las siguientes sesiones de laboratorio, y volver a intentarlo en la entregas de la semana 15, en la de la convocatoria ordinaria, y en la de la convocatoria extraordinaria. En total, 4 intentos.
- El alumno que opte por el **nivel avanzado** y no apruebe el proyecto en la entrega de la semana 15, podrán completarlo y/o modificarlo, y volver a intentarlo en la entregas de la convocatoria ordinaria, y en la de la convocatoria extraordinaria. En total, 3 intentos.

Para promediar con el resto de elementos de evaluación, el proyecto debe obtener una calificación mínima de 5 sobre 10.

7.3. Evaluación

La calificación de este elemento vendrá dada por las notas obtenidas por el alumno en las actividades de coevaluación y autoevaluación de las versiones del proyecto.

b. Evaluación Continua

ELEMENTO DE EVALUACIÓN	PESO EN LA NOTA FINAL	OBSERVACIONES
Cuestionarios de Conocimientos básicos	30%	Para superar cada cuestionario de conocimientos básicos el alumno dispone de cuatro intentos durante el curso, más dos intentos adicionales en las convocatorias ordinaria y extraordinaria. Para promediar con el resto de elementos de evaluación, el alumno debe superar: <ul style="list-style-type: none">• los 6 cuestionarios de conocimientos básicos del nivel básico (para el nivel básico).• Los 6 cuestionarios del nivel básico y el del nivel intermedio (para el nivel intermedio).• Los 6 cuestionarios del nivel básico, el del nivel intermedio y el del nivel avanzado (para el nivel avanzado).
Proyecto integrador	60%	Para promediar con el resto de elementos de evaluación, el proyecto debe obtener una calificación mínima de 5 sobre 10 en el proceso de coevaluación.





Evaluación	10%	La calificación de este elemento vendrá dada por las notas obtenidas por el alumno en las actividades de coevaluación y autoevaluación de las versiones del proyecto. No hay nota mínima.
------------	-----	---

Si un alumno no alcanza los requisitos mínimos descritos en la tabla anterior, su calificación final en la asignatura será el mínimo entre el valor calculado según la ponderación descrita en la tabla y 4.0.

b. Evaluación en la Convocatoria Ordinaria

La Convocatoria Ordinaria se realiza en la fecha prevista en el calendario de exámenes. Esta convocatoria está indicada para:

1. Alumnos que no han aprobado alguno de los niveles en la Evaluación Continua.
2. Alumnos que, habiendo aprobado alguno de los niveles en la evaluación continua, deseen mejorar su calificación presentándose a un nivel superior al aprobado. En este caso, si un alumno no supera el nivel al que se presenta en esta convocatoria, mantiene la calificación obtenida en la Evaluación Continua.

En cualquier caso, el alumno se puede presentar a cualquiera de los tres niveles, pero solamente a uno de ellos. Solamente se tiene que examinar de los cuestionarios asociados al nivel del que se quiere examinar, y presentar el proyecto asociado a dicho nivel.

Esta tabla recoge los criterios de evaluación que se aplicarán en esta convocatoria.

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Cuestionarios de Conocimientos básicos	30%	Se conserva la nota de los cuestionarios superados en la evaluación continua. El alumno solamente tendrá que recuperar los conocimientos básicos que tenga pendientes. Para promediar con el resto de elementos de evaluación, el alumno debe superar, en función del nivel correspondiente: <ul style="list-style-type: none">• Los 6 cuestionarios de conocimientos básicos del nivel básico (para el nivel básico).• Los 6 cuestionarios del nivel básico y el del nivel intermedio (para el nivel intermedio).• Los 6 cuestionarios del nivel básico, el del nivel intermedio y el del nivel avanzado (para el nivel avanzado).
Proyecto integrador	60%	Se conserva la nota de la evaluación continua, siempre que dicha nota supere la calificación





		de 5 sobre 10. En caso contrario, el alumno tendrá que entregar la versión correspondiente al nivel del que se quiere evaluar. Para promediar con el resto de elementos de evaluación, el proyecto debe obtener una calificación mínima de 5 sobre 10 en el proceso de coevaluación.
Evaluación	10%	Si el alumno ha aprobado el proyecto, se conserva la calificación obtenida como evaluador en la convocatoria en la que aprobó el proyecto. En caso contrario, la calificación de este elemento vendrá dada por las notas obtenidas por el alumno en las actividades de coevaluación y autoevaluación de las versiones del proyecto.

Si un alumno no alcanza los requisitos mínimos descritos en la tabla anterior, su calificación final en la asignatura será el mínimo entre el valor calculado según la ponderación descrita en la tabla y 4.

c. Convocatoria Extraordinaria

La Convocatoria Extraordinaria se realiza en la fecha prevista en el calendario de exámenes. Esta convocatoria está indicada solamente para alumnos que no han aprobado alguno de los niveles en la Evaluación Continua o en la Convocatoria Ordinaria.

El alumno se puede presentar a cualquiera de los tres niveles, pero solamente a uno de ellos. Solamente se tiene que examinar de los cuestionarios asociados al nivel del que se quiere examinar, y presentar el proyecto asociado a dicho nivel.

Esta tabla recoge los criterios de evaluación que se aplicarán en esta convocatoria.

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Cuestionarios de Conocimientos básicos	30%	Se conserva la nota de los cuestionarios superados en la convocatoria ordinaria o en la evaluación continua. El alumno solamente tendrá que recuperar los conocimientos básicos que tenga pendientes. Para promediar con el resto de elementos de evaluación, el alumno debe superar, en fun-



		<p>ción del nivel correspondiente:</p> <ul style="list-style-type: none">• Los 6 cuestionarios de conocimientos básicos del nivel básico (para el nivel básico).• Los 6 cuestionarios del nivel básico y el del nivel intermedio (para el nivel intermedio).• Los 6 cuestionarios del nivel básico, el del nivel intermedio y el del nivel avanzado (para el nivel avanzado).
Proyecto integrador	60%	<p>Se conserva la nota de la convocatoria ordinaria, siempre que dicha nota supere la calificación final de 5 sobre 10.</p> <p>En caso contrario, el alumno tendrá que entregar la versión correspondiente al nivel del que se quiere evaluar.</p> <p>Para promediar con el resto de elementos de evaluación, el proyecto debe obtener una calificación mínima de 5 sobre 10 en el proceso de coevaluación.</p>
Evaluación	10%	<p>Si el alumno ha aprobado el proyecto, se conserva la calificación obtenida como evaluador en la convocatoria en la que aprobó el proyecto.</p> <p>En caso contrario, la calificación de este elemento vendrá dada por las notas obtenidas por el alumno en las actividades de coevaluación y autoevaluación de las versiones del proyecto.</p>

Si un alumno no alcanza los requisitos mínimos descritos en la tabla anterior, su calificación final en la asignatura será el mínimo entre el valor calculado según la ponderación descrita en la tabla y 4.

8. Consideraciones finales

Anexo I: Bibliografía recomendada

La bibliografía que se propone a continuación tiene como objeto servir de soporte al aprendizaje de los conceptos y del lenguaje de programación de la asignatura, por lo que aprenderse alguno (¡o todos!) los libros no tiene sentido por sí mismo.

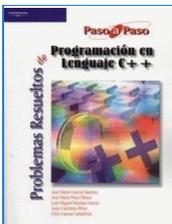
Sin embargo, recomendamos el uso de alguno de los libros mencionados más abajo (en el orden en el que



aparecen). Existen ejemplares en la biblioteca y todos ellos tratan el mismo tema, con diferentes enfoques, ejemplos y orden de exposición.

Texto Básico:

	Autor	Deitel, Harvey M.
	Título	Cómo programar en C/C++
	Ed.	México [etc.] : Prentice-Hall, 2000

	Autor	Félix García Carballeira [et al.]
	Título	Problemas resueltos de programación en lenguaje C/C++
	Ed.	Colección Paso a Paso. Australia [etc.] : Thomson, 2002

Manual de bolsillo:

	Autor	Alejandro Sierra Urrecho, Manuel Alfonseca Moreno
	Título	Programación en C/C++
	Ed.	Anaya Multimedia, 2005

Textos Complementarios

Fundamentos de Programación	
	• Kernighan, B. W. & Ritchie, D. M. (1992) "El lenguaje de programación C". Ed. Méxi-



	<p>co. Prentice-Hall (2ª edición).</p> <ul style="list-style-type: none">• Joyanes Aguilar, L. (1992). "Fundamentos de programación. Algoritmos y estructura de datos". Ed. McGraw-Hill.• Perry, G. (1998). "Aprendiendo Principios de Programación en 24 horas". Ed. Prentice-Hall.
Programación Estructurada	
	<ul style="list-style-type: none">• Wirth, N. (1985). "Algoritmos + Estructuras de Datos = Programas". Ed. del Castillo.• Quero Catalinas, E. & López Herranz (1997). "Programación en Lenguajes Estructurados". Ed. Paraninfo.