



DATOS IDENTIFICATIVOS

Programación I

Asignatura	Programación I			
Código	V05G300V01205			
Titulación	Grado en Ingeniería de Tecnologías de Telecomunicación			
Descriptores	Creditos ECTS	Seleccione	Curso	Cuatrimestre
	6	OB	1	2c
Lengua	Castellano			
Impartición				
Departamento	Ingeniería telemática			
Coordinador/a	Pazos Arias, José Juan			
Profesorado	García Palomares, Ubaldo Manuel Pazos Arias, José Juan Ramos Cabrer, Manuel Santos Suárez, José Manuel			
Correo-e	jose@det.uvigo.es			
Web	http://faitic.uvigo.es			
Descripción	El objetivo principal de esta asignatura es el desarrollo de capacidades de programación en un lenguaje de general alto nivel. El paradigma de programación que se sigue es el de programación estructurada.			

Competencias de titulación

Código	
A4	CG4 Capacidad para resolver problemas con iniciativa, para la toma de decisiones, la creatividad, y para comunicar y transmitir conocimientos, habilidades y destrezas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero Técnico de Telecomunicación.
A9	CG9 Capacidad para trabajar en un grupo multidisciplinar y en un entorno multilingüe y de comunicar, tanto por escrito como de forma oral, conocimientos, procedimientos, resultados e ideas relacionadas con las telecomunicaciones y la electrónica.
A15	CE6/T1 Capacidad para aprender de manera autónoma nuevos conocimientos y técnicas adecuados para la concepción, el desarrollo o la explotación de sistemas y servicios de telecomunicación.
A21	CE12/T7 Conocimiento y utilización de los fundamentos de la programación en redes, sistemas y servicios de telecomunicación.

Competencias de materia

Resultados previstos en la materia	Resultados de Formación y Aprendizaje
Capacidad de expresar la solución de un problema sencillo mediante algoritmos, utilizando programación estructurada.	A4 A21
Capacidad de identificar los datos necesarios para resolver un problema y asociarlos a los tipos correspondientes en función de sus características (tamaño, rango, operadores que actúan sobre ellos).	A4 A21
Codificar algoritmos sencillos a partir de los tres tipos básicos de instrucciones: asignación, selección e iteración.	A21
Declarar y definir funciones haciendo un uso adecuado del paso de parámetros.	A21
Manejar las operaciones de entrada/salida y operar con ficheros.	A21
Definir y utilizar tipos de datos estructurados.	A21
Definir y gestionar estructuras de datos dinámicas (listas).	A21
Crear y utilizar módulos y funciones de biblioteca dentro de un programa.	A15 A21
Deducir el resultado de una secuencia de sentencias básicas, conocidos los datos de entrada.	A21
Manejar herramientas básicas para el desarrollo de programas: editor de textos, compilador, enlazador, depurador y herramientas para la documentación.	A15

Desarrollar proyectos software de pequeña envergadura siguiendo todas las fases: análisis de requisitos, diseño, construcción, evaluación y documentación.	A4 A9 A15 A21
--	------------------------

Contenidos

Tema	
Tema 1: El ordenador y los lenguajes de programación	<ol style="list-style-type: none"> 1. El ordenador 2. Conceptos de programación, programa y paradigma de programación 3. Etapas del proceso de desarrollo software 4. Lenguajes de programación en cuanto a su nivel de abstracción 5. Código fuente y código objeto 6. Compiladores e interpretes 7. El concepto de algoritmo 8. Diagramas de flujo, pseudo código y lenguaje natural para la representación de algoritmos 9. Estructura general de un programa en C
Tema 2: Elementos básicos (enteros, caracteres, reales y punteros)	<ol style="list-style-type: none"> 1. Conceptos clave de C: tipo de datos, identificador, variable, constante, operador y expresión 2. Tipos de datos básicos, sus características en cuanto a tamaño, rango y almacenamiento en memoria, y los operadores que actúan sobre ellos 3. Identificadores 4. Operadores aritméticos 5. Operadores lógicos 6. Operadores relacionales 7. Conversiones de tipos 8. Operaciones de declaración y asignación 9. Concepto de puntero 10. Operadores dirección e indirección
Tema 3: Instrucciones de control (asignación, condicionales, iterativas y de entrada/salida)	<ol style="list-style-type: none"> 1. Tipos básicos de instrucciones de control (secuencia, selección y repetición) 2. Instrucciones de selección (if-else, switch, operador ?) 3. Instrucciones de repetición (for, while, do-while), break, continue 4. Operaciones básicas de entrada/salida por teclado y pantalla (printf, scanf) 5. Directivas de compilación
Tema 4: Funciones	<ol style="list-style-type: none"> 1. Programación modular o procedural: reducción de problemas 2. Principios de la programación estructurada 3. Concepto de función: cabecera de función, prototipo, definición, invocación 4. Funciones sin parámetros 5. Variables globales, locales y estáticas 6. Funciones con paso de parámetros por valor 7. Funciones con paso de parámetros por referencia 8. Paso de argumentos a través de línea de comandos (argc, argv[]) 9. Recursividad
Tema 5: Tipos de datos estructurados	<ol style="list-style-type: none"> 1. Estructuras de datos (array, struct, union) 2. Arrays unidimensionales y bidimensionales 3. Cadenas de caracteres 4. Declaración y utilización de estructuras de datos 5. Declaración de tipos: typedef 6. Estructuras anidadas 7. Funciones de biblioteca que permiten gestionar cadenas de caracteres
Tema 6: Ficheros	<ol style="list-style-type: none"> 1. Concepto de fichero y flujo 2. Flujos estándar 3. Tipos de ficheros: texto y datos 4. Operaciones básicas con ficheros: apertura y cierre, escritura, lectura 5. Modos de acceso 6. Macros NULL y EOF 7. Funciones de biblioteca de entrada/salida en ficheros 8. Escritura y lectura con formato

1. Introducción a la gestión de memoria dinámica
2. Funciones de biblioteca para gestión de memoria dinámica
3. Listas y tipos de listas enlazadas más comunes: simple, doble, circular, circular doble y árbol binario
4. Esquemas de creación, inserción, recorrido, búsqueda y eliminación en listas enlazadas de forma simple
5. Estructura de los nodos en listas enlazadas de forma simple
6. Paso de ficheros a lista y viceversa

Planificación			
	Horas en clase	Horas fuera de clase	Horas totales
Actividades introductorias	2	2	4
Sesión magistral	25	25	50
Prácticas de laboratorio	11	11	22
Proyectos	11	33	44
Tutoría en grupo	0	3	3
Pruebas de tipo test	0	4	4
Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	3	6	9
Resolución de problemas y/o ejercicios	4	10	14

*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

Metodologías	
	Descripción
Actividades introductorias	Introducción a la asignatura tanto en su componente teórica como práctica.
Sesión magistral	Presentación por parte del profesorado del temario de la asignatura. Estas sesiones incluirán la realización de trabajos y la realización de programas por parte de los alumnos.
Prácticas de laboratorio	En el laboratorio se planteará el desarrollo de prácticas guiadas y la realización de programas. Estas prácticas tienen lugar a lo largo de la primera parte del cuatrimestre. En algunas de ellas se pedirá la entrega de informes para su evaluación.
Proyectos	En la segunda parte del laboratorio se propone al alumnado la realización de un proyecto. Este proyecto se realiza a lo largo de la segunda parte del cuatrimestre e incluirá actividades individuales y en grupo.
Tutoría en grupo	Estas tutorías se plantean para la revisión de los trabajos propuestos en la "sesión magistral" y "prácticas de laboratorio".

Atención personalizada	
Metodologías	Descripción
Sesión magistral	Los estudiantes tendrán ocasión de acudir a tutorías personalizadas en el despacho del profesor en el horario que los profesores establecerán a tal efecto a principio de curso y que se publicará en la página de la asignatura. Se prevén tutorías personalizadas para la resolución de dudas sobre los conceptos introducidos en la sesión magistral y las actividades realizadas en las prácticas de laboratorio y en el proyecto.
Prácticas de laboratorio	Los estudiantes tendrán ocasión de acudir a tutorías personalizadas en el despacho del profesor en el horario que los profesores establecerán a tal efecto a principio de curso y que se publicará en la página de la asignatura. Se prevén tutorías personalizadas para la resolución de dudas sobre los conceptos introducidos en la sesión magistral y las actividades realizadas en las prácticas de laboratorio y en el proyecto.
Proyectos	Los estudiantes tendrán ocasión de acudir a tutorías personalizadas en el despacho del profesor en el horario que los profesores establecerán a tal efecto a principio de curso y que se publicará en la página de la asignatura. Se prevén tutorías personalizadas para la resolución de dudas sobre los conceptos introducidos en la sesión magistral y las actividades realizadas en las prácticas de laboratorio y en el proyecto.

Evaluación		
	Descripción	Calificación
Prácticas de laboratorio	Realización de trabajos y resolución de problemas planteados durante las sesiones de laboratorio. En estos trabajos se evaluarán las competencias CG4 y CE12/T7.	10

Proyectos	La evaluación del proyecto se realizará mediante tres pruebas. La primera se centrará principalmente en el diseño, la segunda en la funcionalidad y la tercera en la codificación y estructuración del proyecto. En este proyecto se evaluarán las competencias CG4, CG9, CE6/T1 y CE12/T7.	30
Pruebas de tipo test	(*)Realizarse un cuestionario tipo test a lo largo del curso. Nesta prueba evaluarán las competencias CE6/T1 e CE12/T7.	5
Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	Se realizará 1 ejercicio en el laboratorio de forma individual que consistirá en la realización de un programa en el ordenador. En estas pruebas se evaluarán las competencias CE6/T1 y CE12/T7.	10
Resolución de problemas y/o ejercicios	Ejercicios plantados durante el desarrollo de las sesiones magistrales. Prueba final sobre todos los contenidos de la materia. En estas pruebas se evaluarán las competencias CG4 y CE12/T7.	45

Otros comentarios sobre la Evaluación

A continuación se muestra la **planificación de la asignatura por temas** indicando el momento estimado de los **hitos de evaluación más importantes**:

	Teoría	Laboratorio	Pruebas Teoría	Pruebas Laboratorio
Semana 1	Tema 1			
Semana 2	Tema 2	Práctica 1		
Semana 3	Tema 2 / Tema 3	Práctica 2		Entrega Práctica 1 (EP1)
Semana 4	Tema 3	Práctica 2		
Semana 5	Tema 3	Práctica 3		Entrega Práctica 2 (EP2)
Semana 6	Tema 4	Práctica 3	Prueba Test 1 (CW1)	
Semana 7	Tema 4	PL1		Puntuable Prácticas (PL1)
Semana 8	Tema 4	Proyecto	Prueba Aula 1 (PA1)	
Semana 9	Tema 5	Proyecto		Entrega Diseño Proyecto (PR)
Semana 10	Tema 6	Proyecto	Prueba Test 2 (CW2)	
Semana 11	Tema 6	Proyecto		
Semana 12	Tema 7	Proyecto	Prueba Aula 2 (PA2)	
Semana 13	Tema 7	PL2		Puntuable Proyecto (PL2)
Semana 14	Tema 7	Proyecto		
Período de exámenes			Prueba final sobre todos los contenidos de la asignatura (PFT)	Entrega y defensa del proyecto desarrollado en el laboratorio (PR)

Seguendo las directrices propias de la titulación se ofrece a los alumnos que cursen esta materia dos sistemas de evaluación: **evaluación continua** y **evaluación al final del cuatrimestre**. Se entiende que el alumno sigue la **evaluación continua** si no opta, de forma explícita, por **la evaluación al final del cuatrimestre**. La decisión de optar por evaluación al final del cuatrimestre podrá ser tomada como muy tarde en la semana en la que se realiza en el laboratorio la prueba "Puntuable de prácticas (PL1)".

Para aprobar la asignatura siguiendo el sistema de **evaluación continua** habrá que obtener por lo menos una nota final igual o superior a 5. La nota final (que se calculará como la media armónica entre la suma de las pruebas prácticas y la suma de las pruebas teóricas) vendrá dada por la siguiente expresión:

$$N_f = (2 \cdot NP + N_T) / (NP + N_T)$$

$$NP \text{ (Max. 100\%)} = EP \text{ (Max. 20 \%)} + PL1 \text{ (Max. 20 \%)} + PL2 \text{ (Max. 20\%)} + PR \text{ (Max. 40 \%)}$$

$$N_T \text{ (Max. 100\%)} = CW \text{ (Max. 10 \%)} + PA1 \text{ (Max. 10 \%)} + PA2 \text{ (Max. 20 \%)} + PFT \text{ (Max. 60\%)}$$

Nótese, que la aplicación de la media armónica implica que para aprobar la asignatura es necesario obtener una nota mínima en cada una de las partes (NP y N_T). La nota mínima varía entre 3.3 y 5, dependiendo de lo que se obtenga en la otra parte. Es decir, lo necesario, en cada caso, para conseguir una nota final igual o superior a 5.

La evaluación continua consta de las tareas que se detallan en esta guía y no son recuperables, es decir, si un alumno no puede cumplirlas en el plazo estipulado el profesor no tiene la obligación de repetírselas. La calificación obtenida en las tareas evaluables será válida tan solo para el curso académico en el que se realicen.

La **evaluación al final del cuatrimestre** consistirá en la entrega de un proyecto y en la realización de ejercicios de programación tanto en papel como con el ordenador.

La calificación será de no presentado en los siguientes casos:

1. En caso de no optar por la evaluación al final del cuatrimestre sólo si no presenta ninguna entrega después de la prueba práctica PL1.
2. En caso de si optar por la evaluación al final del cuatrimestre sólo si no realiza las pruebas indicadas.

En las convocatorias extraordinarias se plantearán dos tipos de evaluación:

- La primera para aquellos alumnos que hayan seguido la evaluación **continua** durante el curso. En este caso la evaluación consistirá en la realización de la prueba final (parte teórica) y/o de la entrega de una ampliación del programa proyecto y de la realización de un ejercicio en el laboratorio (parte práctica). Las dos pruebas anteriores tendrán que hacerse ambas o sólo una de ellas en las siguientes circunstancias:
 - la prueba final correspondiente a la parte teórica si en la convocatoria ordinaria no ha superado la misma (N_T menor que 50%);
 - la entrega de la ampliación del programa proyecto y el ejercicio de laboratorio correspondiente a la parte práctica si en la convocatoria ordinaria no se superó la misma (N_P menor que 50%);
 - ambas si se cumplen las dos anteriores o el alumno así lo desea.
- La segunda para aquellos alumnos que no hayan seguido la evaluación continua, o **que la siguieron pero quieren abandonarla**. Esta evaluación consistirá en la entrega de un proyecto y la realización de un examen final con ejercicios de prácticos tanto en papel como con el ordenador.

El alumno que pueda optar de forma voluntaria por una u otra opción, podrá hacerlo hasta el momento de entrega o realización de las pruebas correspondientes.

La nota obtenida en la evaluación continua no se guarda de un curso para el siguiente.

En caso de detección de plagio en alguno de los trabajos/pruebas realizadas la calificación será de suspenso (0) y los profesores comunicarán a la dirección de la escuela el asunto para que tome las medidas que considere oportunas. En el caso de que el plagio se detecte en alguno de los trabajos/pruebas de evaluación continua no se permitirá que el alumno siga este procedimiento de evaluación.

Fuentes de información

Osvaldo Cairo Battistuti, **Fundamentos de Programación**, 2006,

José Rafael García-Bermejo Giner, **Programación Estructurada en C**, 2008,

Brian W. Kernighan & Dennis M. Ritchie, **El Lenguaje de Programación C**, 1995,

James L. Antonakos & Kenneth C. Mansfield Jr., **Programación Estructurada en C**, 2004,

Jorge A. Villalobos S. & Rubby Casallas G., **Fundamentos de Programación: Aprendizaje Activo Basado en Casos**, 2006,

Manuel Caeiro Rodríguez, Enrique Costa Montenegro, Ubaldo García Palomares, Cristina López Bravo, J, **Practicar Programación en C**, 2014,

Recursos Web

- <http://www.Cprogramming.com>
- José R. García-Bermejo Giner: http://maxus.fis.usal.es/FICHAS_C.WEB/11xx_PAGS/11xx.html

Recomendaciones

Asignaturas que continúan el temario

Asignaturas que se recomienda haber cursado previamente

Informática: Arquitectura de ordenadores/V05G300V01103

Otros comentarios

La asignatura Programación II es una continuación de esta asignatura en el segundo curso.
