



# Programación para Sistemas

## Guía de Aprendizaje – Información al estudiante

### 1. Datos Descriptivos

<b>Asignatura</b>	Programación para Sistemas
<b>Materia</b>	Programación
<b>Departamento responsable</b>	Lenguajes y Sistemas Informáticos e Ingeniería de Software
<b>Créditos ECTS</b>	3
<b>Carácter</b>	Obligatoria
<b>Titulación</b>	Grado en Ingeniería Informática (Plan 2009)
<b>Curso</b>	2º
<b>Especialidad</b>	No aplica

<b>Curso académico</b>	2010-2011
<b>Semestre en que se imparte</b>	Tercero
<b>Semestre principal</b>	Tercero
<b>Idioma en que se imparte</b>	Español
<b>Página Web</b>	<a href="http://web3.fi.upm.es/AulaVirtual/course/view.php?id=135">http://web3.fi.upm.es/AulaVirtual/course/view.php?id=135</a>



## 2. Profesorado

NOMBRE Y APELLIDO	DESPACHO	Correo electrónico
José Crespo del Arco (Coord.)	2311	jcrespo@fi.upm.es
M. Del Socorro Bernardos Galindo	5206	sbernardos@fi.upm.es
Jorge Dávila Muro	5205	jdavila@fi.upm.es
Ángel Herranz Nieva	2309	aherranz@fi.upm.es
Juan Luis Pérez Camaño	5002	jlperez@fi.upm.es
José Antonio Pérez Ruy-Díaz	5207	ruy@fi.upm.es
Julio Setién Villarán	5208	jsetien@fi.upm.es

## 3. Conocimientos previos requeridos para poder seguir con normalidad la asignatura

<b>Asignaturas superadas</b>	<ul style="list-style-type: none"><li>• Debe haber superado la asignatura Programación I</li></ul>
<b>Otros resultados de aprendizaje necesarios</b>	<ul style="list-style-type: none"><li>• Desarrollar programas sencillos en un lenguaje de programación de propósito general.</li><li>• Capacidad de comunicación oral y escrita en español.</li></ul>



## 4. Objetivos de Aprendizaje

<b>COMPETENCIAS ASIGNADAS A LA ASIGNATURA Y SU NIVEL DE ADQUISICIÓN</b>		
<b>Código</b>	<b>Competencia</b>	<b>Nivel</b>
CE-4	Capacidad para describir una solución de forma abstracta	2
CE-8	Poseer destrezas fundamentales de la programación que permitan la implementación de los algoritmos y las estructuras de datos en el software	3
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	2

LEYENDA: Nivel de adquisición 1: Conocimiento  
Nivel de adquisición 2: Comprensión  
Nivel de adquisición 3: Aplicación  
Nivel de adquisición 4: Análisis y síntesis



**POLITÉCNICA**



UNIVERSIDAD POLITÉCNICA DE MADRID  
**FACULTAD DE INFORMÁTICA**  
Campus de Montegancedo  
Boadilla del Monte. 28660 Madrid

<b>RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA</b>			
<b>Código</b>	<b>Resultado de aprendizaje</b>	<b>Competencias asociadas</b>	<b>Nivel de adquisición</b>
RA1	Programar aplicaciones C que le permiten la comunicación con el sistema	CE-4, CE-8, CE-9	3
RA2	Programar scripts que automaticen determinadas tareas o faciliten llevar a cabo pruebas funcionales de programa	CE-4, CE-8	2



## 5. Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relacionado con RA
I1	Manejar el entorno Unix	RA1
I2	Diseñar y codificar programas en Lenguaje C	RA1
I3	Depurar y validar programas	RA1
I4	Diseñar y codificar scripts para un entorno Unix	RA2
I5	Automatizar tareas	RA2

EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Tarea 1 (práctica) Shell (bash) y programación con scripts	Semana 7	Entrega remota	22
Tarea 2.1 (práctica) Programación C (parte 1)	Semana 10	Entrega remota	11
Tarea 2.2 (práctica) Programación C (parte 2)	Semana 13	Entrega remota	17
Tarea 2.3 (práctica) Programación C (parte 3)	Semana 16	Entrega remota	17
Examen	Semana 17	Aula o Sala Informática	33
			<b>Total: 100%</b>

Nota: las fechas son aproximadas y orientativas.

Nota: este cuadro se refiere al sistema general de evaluación continua.



## CRITERIOS DE CALIFICACIÓN

La asignatura de Programación para Sistemas se configura como una asignatura con sistema general de evaluación continua en la que las prácticas constituyen la parte principal de la asignatura. La asignatura consta de prácticas informáticas y de un examen escrito.

El examen escrito se realizará en la fecha marcada por Jefatura de Estudios. La nota numérica final (NF) se calculará de acuerdo a la fórmula que combina el examen escrito con las tareas prácticas que se proponen en la asignatura:

$$NF = (2/3) P + (1/3) T$$

donde P es la nota de prácticas y T la de examen escrito, siempre y cuando ambas partes estén aprobadas (superiores o iguales a 5,0).

La nota de prácticas se calcula de la siguiente manera:

$$P = (1/3) Tarea1 + (2/3) Tarea2$$

donde Tarea1 es la parte práctica sobre shell (bash) y programación con scripts, y Tarea2 es la parte práctica sobre programación C. Se debe tener un aprobado en ambas partes Tarea1 y Tarea2 para aprobar la nota de prácticas.

Un aprobado en prácticas (en el conjunto de las prácticas) se guarda para futuras convocatorias.

### Sistema de evaluación mediante sólo prueba final

El Sistema de evaluación mediante sólo prueba final sólo se ofrecerá si así lo exige la Normativa Reguladora de los Sistemas de Evaluación en la UPM que esté vigente en el curso académico 2010-2011, y el procedimiento para optar por este sistema estará sujeto a lo que establezca en su caso Jefatura de Estudios de conformidad con lo que estipule dicha Normativa.

Los alumnos que lo soliciten en las condiciones establecidas en dicha normativa serán evaluados con prueba final fuera del sistema normal de evaluación continua. Como en el caso del sistema general de evaluación continua, un aprobado en prácticas se guarda para futuras convocatorias.



Evaluación en el periodo extraordinario

La evaluación en el periodo extraordinario tendrá un sistema de evaluación mediante sólo prueba final.

## 6. Contenidos y Actividades de Aprendizaje

CONTENIDOS ESPECÍFICOS		
Bloque / Tema / Capítulo	Apartado	Indicadores Relacionados
<b>Tema 1: Entorno Unix y Programación con scripts</b>	1.1 Introducción a Unix y al intérprete de mandatos. Lenguaje de Shell. Comandos y programas útiles.	I1
	1.2 Programación con scripts. Automatización de tareas.	I4, I5
<b>Tema 2: Programación C</b>	2.1 Programación estructurada y fundamentos de C. Tipos de datos, estructuras de control, Entrada / Salida, funciones	I2
	2.2 Herramientas de desarrollo: editor, compilador, enlazador, depurador, make	I2, I3
	2.3 Programación C avanzada. Arrays, strings, punteros	I2, I3

## 7. Breve descripción de las modalidades organizativas utilizadas y de los métodos de enseñanza empleados

Tabla 7. Modalidades organizativas de la enseñanza

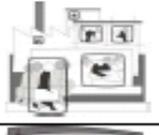
MODALIDADES ORGANIZATIVAS DE LA ENSEÑANZA		
Escenario	Modalidad	Finalidad
	Clases Teóricas	<i>Hablar a los estudiantes</i>
	Seminarios-Talleres	<i>Construir conocimiento a través de la interacción y la actividad de los estudiantes</i>
	Clases Prácticas	<i>Mostrar a los estudiantes cómo deben actuar</i>
	Prácticas Externas	<i>Completar la formación de los alumnos en un contexto profesional</i>
	Tutorías	<i>Atención personalizada a los estudiantes</i>
	Trabajo en grupo	<i>Hacer que los estudiantes aprendan entre ellos</i>
	Trabajo autónomo	<i>Desarrollar la capacidad de autoaprendizaje</i>



Tabla 5. Métodos de enseñanza

MÉTODOS DE ENSEÑANZA		
	Método	Finalidad
	Método Expositivo/Lección Magistral	Transmitir conocimientos y activar procesos cognitivos en el estudiante
	Estudio de Casos	Adquisición de aprendizajes mediante el análisis de casos reales o simulados
	Resolución de Ejercicios y Problemas	Ejercitar, ensayar y poner en práctica los conocimientos previos
	Aprendizaje Basado en Problemas (ABP)	Desarrollar aprendizajes activos a través de la resolución de problemas
	Aprendizaje orientado a Proyectos	Realización de un proyecto para la resolución de un problema, aplicando habilidades y conocimientos adquiridos
	Aprendizaje Cooperativo	Desarrollar aprendizajes activos y significativos de forma cooperativa
	Contrato de Aprendizaje	Desarrollar el aprendizaje autónomo

Se conoce como método expositivo "la presentación de un tema lógicamente estructurado con la finalidad de facilitar información organizada siguiendo criterios adecuados a la finalidad pretendida". Esta metodología -también conocida como lección (lecture)- se centra fundamentalmente en la exposición verbal por parte del profesor de los contenidos sobre la materia objeto de estudio. El término "lección magistral" se suele utilizar para denominar un tipo específico de lección impartida por un profesor en ocasiones especiales.

Análisis intensivo y completo de un hecho, problema o suceso real con la finalidad de conocerlo, interpretarlo, resolverlo, generar hipótesis, contrastar datos, reflexionar, completar conocimientos, diagnosticarlo y, en ocasiones, entrenarse en los posibles procedimientos alternativos de solución.

Situaciones en las que se solicita a los estudiantes que desarrollen las soluciones adecuadas o correctas mediante la ejercitación de rutinas, la aplicación de fórmulas o algoritmos, la aplicación de procedimientos de transformación de la información disponible y la interpretación de los resultados. Se suele utilizar como complemento de la lección magistral.

Método de enseñanza-aprendizaje cuyo punto de partida es un problema que, diseñado por el profesor, el estudiante ha de resolver para desarrollar determinadas competencias previamente definidas.

Método de enseñanza-aprendizaje en el que los estudiantes llevan a cabo la realización de un proyecto en un tiempo determinado para resolver un problema o abordar una tarea mediante la planificación, diseño y realización de una serie de actividades, y todo ello a partir del desarrollo y aplicación de aprendizajes adquiridos y del uso efectivo de recursos.

Enfoque interactivo de organización del trabajo en el aula en el cual los alumnos son responsables de su aprendizaje y del de sus compañeros en una estrategia de corresponsabilidad para alcanzar metas e incentivos grupales. Es tanto un método, a utilizar entre otros, como un enfoque global de la enseñanza, una filosofía.

Un acuerdo establecido entre el profesor y el estudiante para la consecución de unos aprendizajes a través de una propuesta de trabajo autónomo, con una supervisión por parte del profesor y durante un periodo determinado. En el contrato de aprendizaje es básico un acuerdo formalizado, una relación de contraprestación recíproca, una implicación personal y un marco temporal de ejecución.



## **BREVE DESCRIPCIÓN DE LAS MODALIDADES ORGANIZATIVAS UTILIZADAS Y METODOS DE ENSEÑANZA EMPLEADOS**

<b>CLASES DE TEORIA</b>	Clases de exposición por parte del profesor de material teórico y práctico (con ejemplos de código, etc.) en aula tradicional.
<b>CLASES DE PROBLEMAS</b>	Estas clases tendrán un contenido fundamentalmente práctico y se desarrollarán preferentemente en aulas informáticas para facilitar que los alumnos puedan resolver casos prácticos durante parte de las mismas.
<b>PRÁCTICAS</b>	En la asignatura, de naturaleza en gran parte práctica, los alumnos deberán realizar prácticas. En las aulas informáticas los alumnos podrán contar con el software necesario para su realización.
<b>TRABAJOS AUTONOMOS</b>	En la asignatura los alumnos deberán estudiar y afianzar los conceptos de los contenidos tratados.
<b>TRABAJOS EN GRUPO</b>	En la asignatura las prácticas se realizarán como trabajo en grupo.
<b>TUTORÍAS</b>	Los alumnos podrán recibir asistencia sobre cuestiones de la asignatura en el horario de tutorías de los profesores de la asignatura.



## 8. Recursos didácticos

<b>RECURSOS DIDÁCTICOS</b>	
<b>BIBLIOGRAFÍA</b>	Material didáctico y de apoyo proporcionado por los profesores de la asignatura.
	“Programación estructurada en C”, J.L. Antonakos, K.C. Mansfield Jr. Prentice-Hall 1997.
	“The C programming language”, B. Kernighan, D.Ritchie. Segunda edición. Prentice-Hall 1988.
	“Learning the bash shell”, C. Newham, B. Rosenblatt. O’Reilly 2005.
	“The UNIX programming environment”, B. Kernighan, R. Pike, Second Edition. Prentice-Hall 1988.
<b>RECURSOS WEB</b>	“The GNU Bash Reference Manual (revised for version 3.2)”, Chet Ramey and Brian Fox <a href="http://www.network-theory.co.uk/docs/bashref/">http://www.network-theory.co.uk/docs/bashref/</a>
	Página web de la asignatura: <a href="http://web3.fi.upm.es/AulaVirtual/course/view.php?id=135">http://web3.fi.upm.es/AulaVirtual/course/view.php?id=135</a>
<b>EQUIPAMIENTO</b>	Sitio Moodle de la asignatura: <a href="http://web3.fi.upm.es/AulaVirtual/course/view.php?id=135">http://web3.fi.upm.es/AulaVirtual/course/view.php?id=135</a>
	Laboratorio: sala de ordenadores asignada por Jefatura de Estudios
	Aula asignada por Jefatura de Estudios
	Software: entorno Unix/Linux, gcc, gdb/ddd, make, bash



### 9. Cronograma de trabajo de la asignatura

Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
Semana 1 5,25 horas	•	• Tema 1, ejercicios prácticos 1,5 horas	• Estudio 2,75 horas	• Formación grupo de prácticas 1 hora	•	•
Semana 2 5,75 horas	•	• Tema 1, ejercicios prácticos y realización de Tarea 1 1,5 horas	• Estudio y realización de Tarea 1 2 horas	• Realización de Tarea 1 2 horas	•	• Entrega informe datos grupo 0.25 horas
Semana 3 5 horas	•	• Tema 1, ejercicios prácticos y realización de Tarea 1 1,5 horas	• Estudio y realización de Tarea 1 1,5 horas	• Realización de Tarea 1 2 horas	•	•
Semana 4 5 horas	•	• Tema 1, ejercicios prácticos y realización de Tarea 1 1,5 horas	• Estudio y realización de Tarea 1 1,5 horas	• Realización de Tarea 1 2 horas	•	•



Semana 5 4,75 horas	•	• Tema 1, ejercicios prácticos y realización de Tarea 1 1,5 horas	• Estudio y realización de Tarea 1 1,25 horas	• Realización de Tarea 1 2 horas	•	•
Semana 6 4,75 horas	•	• Tema 1, ejercicios prácticos y realización de Tarea 1 1,5 horas	• Estudio y realización de Tarea 1 1,25 horas	• Realización de Tarea 1 2 horas	•	•
Semana 7 4,5 horas	• Tema 2: clase teoría 1	•	• Estudio y realización de Tarea 2.1 1,25 horas	• Realización de Tarea 2.1 2 horas	•	• Entrega práctica 0,25 horas
Semana 8 5,25 horas	•	• Tema 2, ejercicios prácticos y realización de Tarea 2.1 2 horas	• Estudio y realización de Tarea 2.1 1,25 horas	• Realización de Tarea 2.1 2 horas	•	•
Semana 9 4,25 horas	• Tema 2: clase teoría 1	•	• Estudio y realización de Tarea 2.1 1,25 horas	• Realización de Tarea 2.1 2 horas	•	•



Semana 10 5,5 horas	•	• Tema 2, ejercicios prácticos y realización de Tarea 2.1 2 horas	• Estudio y realización de Tarea 2.2 1,25 horas	• Realización de Tarea 2.2 2 horas	•	• Entrega práctica 0,25 horas
Semana 11 4,75 horas	•	• Tema 2, ejercicios prácticos y realización de Tarea 2.2 1,5 horas	• Estudio y realización de Tarea 2.2 1,25 horas	• Realización de Tarea 2.2 2 horas	•	•
Semana 12 4,25 horas	• Tema 3: clase teoría 1	•	• Estudio y realización de Tarea 2.2 1,25 horas	• Realización de Tarea 2.2 2 horas	•	•
Semana 13 5,5 horas	•	• Tema 3, ejercicios prácticos y realización de Tarea 2.2 2 horas	• Estudio y realización de Tarea 2.3 1,25 horas	• Realización de Tarea 2.3 2 horas	•	• Entrega práctica 0,25 horas
Semana 14 4,25 horas	• Tema 3: clase teoría 1	•	• Estudio y realización de Tarea 2.3 1,25 horas	• Realización de Tarea 2.3 2 horas	•	•



Semana 15 5,25 horas	•	• Tema 3, ejercicios prácticos y realización de Tarea 2.3 2 horas	• Estudio y realización de Tarea 2.3 1,25 horas	• Realización de Tarea 2.3 2 horas	•	•
Semana 16 y examen final 7 horas	•	• Tema 3, ejercicios prácticos y realización de Tarea 2.3 1.5 horas	• Estudio y realización de Tarea 2.3 1,25 horas	• Realización de Tarea 2.3 2 horas	• Examen final 2 horas	• Entrega práctica 0,25 horas
TOTAL	• 4	• 20	• 22,75	• 31	• 2	• 1,25

Nota: Para cada actividad se especifica la dedicación en horas que implica para el alumno. Esta distribución de esfuerzos debe entenderse para el "estudiante medio", por lo que si bien puede servir de orientación, no debe tomarse en ningún caso en sentido estricto a la hora de planificar su trabajo. Cada alumno deberá hacer su propia planificación para alcanzar los resultados de aprendizaje descritos en esta Guía y ajustar dicha planificación en un proceso iterativo en función de los resultados intermedios que vaya obteniendo.

Nota: en "Otros" se han recogido actividades como la realización de entregas de informes y prácticas.



**POLITÉCNICA**



UNIVERSIDAD POLITÉCNICA DE MADRID  
**FACULTAD DE INFORMÁTICA**  
Campus de Montegancedo  
Boadilla del Monte. 28660 Madrid