



**POLITÉCNICA**



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

## **PROGRAMACIÓN I**

### **Guía de Aprendizaje – Información al estudiante**

#### **1. Datos Descriptivos**

<b>Asignatura</b>	<b>Programación I</b>
<b>Materia</b>	<b>Programación</b>
<b>Departamento responsable</b>	<b>DLSIIS</b>
<b>Créditos ECTS</b>	<b>6</b>
<b>Carácter</b>	<b>Obligatoria</b>
<b>Titulación</b>	<b>Graduado en Matemáticas e Informática</b>
<b>Curso</b>	<b>1º</b>
<b>Especialidad</b>	<b>N/A</b>

<b>Curso académico</b>	<b>2012-2013</b>
<b>Semestre en que se imparte</b>	<b>Septiembre a enero</b>
<b>Semestre principal</b>	<b>Primero</b>
<b>Idioma en que se imparte</b>	<b>Español</b>
<b>Página Web</b>	<b>Moodle</b>



**POLITÉCNICA**



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

La asignatura *Programación I* forma parte de los planes de estudios de Grado en Ingeniería Informática y Grado en Matemáticas e Informática que se imparten en la Facultad de Informática de la Universidad Politécnica de Madrid.

La asignatura tiene 6 créditos ECTS y requiere una dedicación total de unas 160 horas de trabajo. El curso tiene carácter semestral y se imparte en el primer semestre del curso académico (Septiembre a Febrero) con una duración de 16 semanas efectivas. Esto supone una dedicación por parte del estudiante de unas 10 horas de trabajo por semana. De estas 10 horas de trabajo semanal, normalmente de 3,5 a 5 horas serán horas de clase y de 6,5 a 5 serán horas de trabajo adicional, para realizar tareas individuales o en grupo. Se impartirá un grupo de repetición en el segundo semestre.

Este curso es una introducción a la computación y a la programación. Su objetivo principal es presentar a los estudiantes los principios de la resolución sistemática de problemas a través de la programación y las reglas básicas de la computación. El curso aborda la programación desde el punto de vista de los paradigmas funcional e imperativo. El lenguaje que se utilizará para transmitir los contenidos es Java. De esta manera, el alumno se empieza a familiarizar con un lenguaje de propósito general, de amplia utilización en el mundo laboral, actualizado a las tecnologías actuales y que será utilizado ampliamente a lo largo de sus estudios de grado. No obstante, sólo se presenta de este lenguaje la parte necesaria para cumplir el objetivo del curso. Por tanto, no es un curso de Java ni de programación orientada a objetos.



## 2. Profesorado

**COORDINADORA: Pilar Herrero**

<b>NOMBRE Y APELLIDO</b>	<b>DESPACHO</b>	<b>Correo electrónico</b>
José Manuel Burgos	D-2312	<a href="mailto:jmburgos@fi.upm.es">jmburgos@fi.upm.es</a>
Javier Galve	D-2307	<a href="mailto:jgalve@fi.upm.es">jgalve@fi.upm.es</a>
Julio García	D-2306	<a href="mailto:juliog@fi.upm.es">juliog@fi.upm.es</a>
Francisco Gisbert	D-2306	<a href="mailto:fgisbert@fi.upm.es">fgisbert@fi.upm.es</a>
Pilar Herrero	D-2304	<a href="mailto:pherrero@fi.upm.es">pherrero@fi.upm.es</a>

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

<b>Asignaturas superadas</b>	N/A
<b>Otros resultados de aprendizaje necesarios</b>	<b>Conocimientos básicos de aritmética y álgebra. Capacidad de comunicación oral y escrita en castellano.</b>

Para seguir este curso no es necesario tener ninguna experiencia previa en programación. Además de los conocimientos enunciados en la tabla anterior, para seguirlo con éxito son deseables curiosidad, auto-disciplina y habilidades para interrelacionarse con los compañeros, necesarias para trabajar en grupo.



## 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-7	Conocer los cimientos esenciales y fundacionales de la informática, subrayando los aspectos esenciales de la disciplina que permanecen inalterables ante el cambio tecnológico.	3
CE-8	Formalización y especificación de problemas reales cuya solución requiere el uso de la informática.	2
CE-9	Capacidad de elegir y usar los métodos analíticos y de modelización relevantes, y de describir una solución de forma abstracta.	2
CE-13	Poseer destrezas fundamentales de la programación que permitan la implementación de los algoritmos y las estructuras de datos en el software.	2
CE-43	Capacidad para trabajar de forma efectiva como individuo, organizando y planificando su propio trabajo, de forma independiente o como miembro de un equipo.	3

**LEYENDA:** Nivel de adquisición 1: Conocimiento  
Nivel de adquisición 2: Comprensión  
Nivel de adquisición 3: Aplicación

<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	Desarrollar programas sencillos en un lenguaje de programación de propósito general.	CE-7, CE-8, CE-43	2
RA2	Aplicar las pautas explícitas de un proceso de modelización basado en la clasificación del problema para ir del enunciado del problema al programa.	CE-9, CE-43	2
RA3	Usar un entorno de programación para editar, compilar y ejecutar programas	CE-13, CE-43	2



## 5. Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relaciona-do con RA
I1	Utilizar cualquier tipo de dato elemental para resolver un cómputo matemático o lógico.	RA1
I2	Resolver un problema de solución directa mediante una función.	RA1
I3	Resolver un problema de análisis de casos mediante una función que contiene frases de selección.	RA1
I4	Resolver un problema de manejo de enumerados mediante una función.	RA1
I5	Resolver un problema de manejo de agregados mediante una función.	RA1
I7	Utilizar adecuadamente la frase de iteración while y/o la for para hacer un recorrido iterativo de una colección.	RA1
I8	Utilizar un procedimiento equivalente a una función.	RA1
I10	Utilizar un procedimiento para leer datos de la entrada estándar.	RA1
I11	Comprender los algoritmos de ordenación de la burbuja, inserción y selección.	RA1
I12	Extraer información del enunciado del problema y convertirlo en piezas útiles para el programa.	RA2
I13	Clasificar un problema a partir de su enunciado.	RA2
I14	Aplicar correctamente el esquema adecuado al tipo de problema.	RA2
I15	Preparar un juego de pruebas significativo para el programa y los subprogramas que lo componen.	RA2
I16	Manejar las funcionalidades básicas de un entorno de programación.	RA3



a) Sistema de evaluación continua que seguirán de manera generalizada los alumnos

<b>EVALUACIÓN</b>			
<b>Breve descripción de las actividades evaluables</b>	<b>Momento</b>	<b>Lugar</b>	<b>Peso en la calif.</b>
<b>Prueba 1</b>	<b>Semana 4</b>	<b>Aula</b>	<b>10%</b>
<b>Prueba 2</b>	<b>Semana 8</b>	<b>Aula</b>	<b>20%</b>
<b>Prueba 3</b>	<b>Semana 12</b>	<b>Aula</b>	<b>30%</b>
<b>Prueba Final</b>	<b>Semana 17</b>	<b>Aula</b>	<b>40%</b>
<b>  0.10 * P1 + 0.20 * P2 + 0.30 * P3 + 0.40 * PF si PF &gt;= 5</b>			
<b>NOTA FINAL =  </b>			
<b>  S eoc</b>			

En el sistema de evaluación continua, se realizan cuatro pruebas de evaluación en fechas aproximadas a las semanas indicadas en la tabla anterior, Las fechas serán publicadas con la debida antelación para cada grupo. Las pruebas son individuales y se realizan siempre en horas de clase.

Las pruebas consisten en la valoración de trabajos similares a los propuestos en la marcha del curso en forma tanto de ejercicios como de prácticas hechas en clase y/o en casa. El formato de las tres primeras pruebas será diseñado específicamente por cada profesor en su grupo según las características de su estilo de enseñanza. La prueba final será una prueba escrita de formato común a todos los grupos.

Las tres primeras pruebas son pruebas de evaluación principalmente formativa. Sirven para comprobar el progreso del alumno durante el curso y poder establecer las medidas necesarias para corregir su marcha. La última prueba es calificativa y tiene carácter de prueba de conocimientos mínimos. Establece si el alumno tiene los conocimientos necesarios para poder aprobar la asignatura.

La nota final de la asignatura se obtiene a partir de las calificaciones de las pruebas, según la fórmula ponderada expresada arriba. Para poder hacer la ponderación, es necesario obtener al menos un cinco en la última prueba; en caso contrario, la calificación final es de suspenso.



- b) Sistema de evaluación de “sólo prueba final” que seguirán aquellos alumnos que así lo soliciten (de acuerdo con las normas y pautas establecidas por Jefatura de Estudios) al comienzo de la asignatura.

<b>EVALUACION</b>			
<b>Breve descripción de las actividades evaluables</b>	<b>Momento</b>	<b>Lugar</b>	<b>Peso en la calif.</b>
<b>Examen</b>	<b>Marcado por Jefatura de Estudios</b>	<b>Marcado por Jefatura de Estudios</b>	<b>100</b>
			<b>Total:</b>

- c) Prueba de evaluación extraordinaria (en julio) a la que optarán los alumnos que, habiendo seguido el sistema de evaluación continua, no hayan superado la asignatura a lo largo del periodo académico.

<b>EVALUACION SUMATIVA</b>			
<b>Breve descripción de las actividades evaluables</b>	<b>Momento</b>	<b>Lugar</b>	<b>Peso en la calif.</b>
<b>Examen</b>	<b>Marcado por Jefatura de Estudios</b>	<b>Marcado por Jefatura de Estudios</b>	<b>100</b>
			<b>Total:</b>

## **6.Contenidos y Actividades de Aprendizaje**

### **Organización del Curso**



El curso está organizado en Unidades de Estudio (UEs). Cada UE es una unidad de contenidos del temario y se corresponde con una o dos semanas del calendario de impartición del curso. Cada semana se compone de dos sesiones. La primera está más centrada en la teoría, con explicaciones del profesor intercaladas con ejercicios a trabajar por el alumno en clase. La segunda es una sesión de aula informática orientada a dar al alumno algo de experiencia práctica con herramientas reales y a explicar algunos de los principios de la teoría con ejemplos de su aplicación. Adicionalmente se proponen trabajos prácticos para realizar en casa durante los días entre sesiones.

<b>CONTENIDOS ESPECÍFICOS</b>		
<b>Bloque</b>	<b>Contenido</b>	<b>Indicadores Relacionados</b>
<b>UE#01</b>	<b>Familiarización con el Entorno y con el Lenguaje de Programación.</b>	<b>I16</b>
<b>UE#02</b>	<b>Datos Elementales. Salida de Datos. Estado Explícito. Asignación. Bloque.</b>	<b>I1</b>
<b>UE#03</b>	<b>Problemas de Solución Directa. Funciones.</b>	<b>I2, I12-I15</b>
<b>UE#04</b>	<b>Problemas de Análisis de Casos. Selección.</b>	<b>I3, I4, I12-I15</b>
<b>UE#05</b>	<b>Agregados.</b>	<b>I5, I12-I15</b>
<b>UE#06</b>	<b>Problemas de Recorrido. Bucles</b>	<b>I7, I12-I15</b>
<b>UE#07</b>	<b>Acciones. Entrada de Datos.</b>	<b>I8, I9, I10, I12-I15</b>
<b>UE#08</b>	<b>Colecciones Indexadas. Arrays</b>	<b>I6, I7, I12-I15</b>



## 7. Recursos didácticos

RECURSOS DIDÁCTICOS	
<b>BIBLIOGRAFÍA</b>	<i>Program Design and Development, Rick Mercer, 2011</i>
<b>RECURSOS WEB</b>	Sitio Moodle de la asignatura ( <a href="http://web3.fi.upm.es/AulaVirtual/">http://web3.fi.upm.es/AulaVirtual/</a> )
<b>EQUIPAMIENTO</b>	Aula de Clase
	Aula Informática

### Software

- Kit de Desarrollo de Java JDK
- Entorno de Desarrollo Integrado (IDE) DrJava

### Bibliografía Complementaria:

- *Java How to Program, 8/e, Late Objects Version.* P. Deitel, H. Deitel. Deitel, 2010.

## 8. Cronograma de trabajo de la asignatura

Semana	Actividades en Aula	Trabajo Individual / En Grupo	Actividades de Evaluación
Semanas 1-3	4	6	
Semana 4	2	6	2
Semanas 5-7	4	6	
Semana 8	2	6	2
Semanas 9-11	4	6	
Semana 12	2	6	2
Semanas 13-15	4	6	
Semana 16	4	8	
Semana 17			2

Nota: Para cada actividad se especifica la dedicación en horas que implica para el alumno