



CEU

*Universidad
San Pablo*

Guía docente

PROGRAMACIÓN I

CURSO 1 SEMESTRE 1

GRADO EN INGENIERÍA DE SISTEMAS DE INFORMACIÓN

MODALIDAD PRESENCIAL

CURSO 2021-2022

ESCUELA POLITÉCNICA SUPERIOR

1. IDENTIFICACIÓN DE LA ASIGNATURA

1.1. ASIGNATURA

Nombre: Programación I		
Código: c104 17435		
Curso(s) en el que se imparte: 1	Semestre(s) en el que se imparte: 1	
Carácter: Básico	ECTS: 6	Horas ECTS: 25
Idioma: Español	Modalidad: Presencial	
Grado en que se imparte la asignatura: Ingeniería de Sistemas de Información		
Facultad en la que se imparte la titulación: Escuela Politécnica Superior		

1.2. ORGANIZACIÓN DE LA ASIGNATURA

Departamento: Tecnologías de la Información
Área de conocimiento: Ciencia de la Computación e Inteligencia Artificial

2. PROFESORADO DE LA ASIGNATURA

2.1. IDENTIFICACIÓN DEL PROFESORADO

Responsable de Asignatura	DATOS DE CONTACTO
Nombre:	Mariano Fernández López
Tlfno (ext):	14828
Email:	mfernandez.eps@ceu.es
Despacho:	D.2.2.1
Perfil Docente e Investigador	Formación académica: <ul style="list-style-type: none">■ Doctor en Informática (con premio extraordinario de tesis),■ Máster en Ingeniería del Software y■ Máster en Ingeniería del Conocimiento por la Universidad Politécnica de Madrid Categoría: Profesor Titular
Líneas de Investigación:	Ontología computacional (2 sexenios de investigación)

2.2. ACCIÓN TUTORIAL

Para todas las consultas relativas a la asignatura, los alumnos pueden contactar con el/los profesores a través del e-mail, del teléfono y en el despacho a las horas de tutoría que se harán públicas, en el portal del alumno.

3. DESCRIPCIÓN DE LA ASIGNATURA

Una vez terminada esta asignatura, el estudiante será capaz de implementar un sistema de altas, bajas y modificaciones de datos en un lenguaje orientado a objetos. Los dos mecanismos más importantes que debe acabar dominando son la herencia y el polimorfismo. Es una

asignatura fundamental para el resto de las asignaturas de programación y todas aquellas que están relacionadas con el desarrollo de software en general.

4. COMPETENCIAS

4.1. COMPETENCIAS

Código	Competencias Básicas y Generales
CG1	Cognitivos, como solución de problemas, pensamiento analítico y sistémico, emitir juicios documentados, uso eficiente de información, dirigir observaciones, investigaciones, inventar y crear cosas nuevas, analizar datos, presentar datos, expresión oral y escrita.
CB1	Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.
CB2	Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.
CB3	Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.
CB4	Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

Código	Competencias Específicas
BAS4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
BAS5	Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

4.2. RESULTADOS DE APRENDIZAJE

Código	Resultados de Aprendizaje
RA1-PR1	Ser capaz de compilar desde la consola así como de manejar un gestor de dependencias y un gestor de repositorios.
RA2-PR1	Ser capaz de modelizar y programar mediante clases y objetos.
RA3-PR1	Ser capaz de aplicar los mecanismos de herencia y polimorfismo.
RA4-PR1	Ser capaz de aplicar los principios de máxima cohesión y mínimo acoplamiento.

5. ACTIVIDADES FORMATIVAS

5.1. DISTRIBUCIÓN TRABAJO DEL ESTUDIANTE

Total horas de la asignatura	150
------------------------------	-----

Nombre	Horas presenciales
Clase magistral	12
Seminario	36
Taller, seminario de grupo o tutoría académica	18
Prácticas	12
Otras	6
TOTAL Horas Presenciales	84

Nombre	Horas No Presenciales
Trabajo Autónomo del Estudiante	66

5.2. DESCRIPCIÓN ACTIVIDADES FORMATIVAS

Actividad	Definición
Clase magistral	Actividad formativa orientada preferentemente a la competencia de adquisición de conocimiento (competencia 1 RD 861) y representativa de las materias más teóricas. Prioriza la transmisión de conocimientos por parte del profesor, exigiendo al alumno la preparación previa o el estudio posterior.
Seminario	Actividad formativa que potencia la participación del alumno en la interpretación razonada de los conocimientos y de las fuentes del área de estudio. Se orientada preferentemente a la competencia de aplicación de los conocimientos (competencia 2 RD 861), así como a la capacidad de reunir, interpretar y juzgar información y datos relevantes (competencia 3 RD 861). Es representativa de las materias o actividades de perfil mixto, teóricas - práctico.
Seminario de grupo Tutoría Académica Taller	Actividad formativa orientada preferentemente a la competencia de adquisición de habilidades para la transmisión de conocimientos (competencia 4 RD 861) y representativa de las materias de carácter más metodológico. Prepara al alumno para la comunicación escrita - oral y la transmisión de conocimientos.
Prácticas	Actividad formativa orientada preferentemente a la competencia de aplicación de los conocimientos (competencia 2 RD 861) y representativa de las materias o actividades prácticas (laboratorios, trabajos de campo, prácticas tuteladas, prácticas regladas, prácticas asistenciales, prácticum, etc.).

6. SISTEMAS Y CRITERIOS DE EVALUACIÓN

6.1. ASISTENCIA A CLASE

Para poder acogerse al sistema de evaluación continua es precisa la asistencia al 75 % de las clases de teoría (se realizarán controles de asistencia). Ya que el alumno puede faltar el 25 % del total de las clases, no se admitirán justificaciones de ausencia.

Si no se satisface el mínimo de asistencia, la calificación final en la evaluación continua será *no presentado*.

Aquellos alumnos que hayan superado la asignatura de Programación II durante el curso anterior estarán exentos de la obligación de asistencia. Tal exención no regirá necesariamente para aquéllos a quienes se les haya reconocido una asignatura equivalente a Programación II procedentes de otra titulación.

6.2. SISTEMAS Y CRITERIOS DE EVALUACIÓN

CONVOCATORIA ORDINARIA (Evaluación Continua)	
Nombre	Peso
Tests en el campus virtual	10 %
Examen de capacitación	40 %
Práctica	50 %

CONVOCATORIA EXTRAORDINARIA	
Nombre	Peso
Examen extraordinario	100 %

6.3. DESCRIPCIÓN SISTEMAS DE EVALUACIÓN

Sistemas de Evaluación	Definición
Tests en el campus virtual	A lo largo del curso se publicará una serie de baterías de tests. Cada una de ellas estará disponible para los alumnos durante un tiempo limitado y tendrán un número de intentos acotado. Todas las baterías tendrán la misma ponderación en la nota final. Es imprescindible obtener al menos un 5 sobre 10 de media en las baterías de tests para aprobar la asignatura en convocatoria ordinaria.
Examen de capacitación	Examen escrito que abarcará toda la materia con aproximadamente un 10 % de tipo test o prueba objetiva, un 30 % de preguntas de razonamiento, y un 60 % de problemas, ejercicios, supuestos, etc. Es imprescindible obtener un 5 sobre 10 en este examen para poder aprobar la asignatura. Si los medios lo permiten, es posible que el examen de capacitación se realice en parte o en su totalidad con ordenador.
Práctica	Desarrollo de una aplicación mediante programación orientada a objetos, utilizando los mecanismos de herencia y polimorfismo. Es imprescindible aprobar la práctica para poder aprobar la asignatura. La nota final de la práctica estará desglosada de la siguiente manera: <ul style="list-style-type: none">■ Documentación: 25 %,■ Uso de herramientas de soporte al desarrollo: compilación desde consola, herramienta de gestión de dependencias, herramienta de gestión de repositorios, etc.: 15%.■ Producto funcionando: 60 %.
Examen extraordinario	El alumno que no supere la asignatura en la convocatoria ordinaria tendrá la opción de presentarse a la convocatoria extraordinaria. Esta convocatoria constará de una prueba presencial única que determinará la calificación final de la asignatura, sin tener en cuenta su rendimiento académico en la convocatoria ordinaria. Si los medios lo permiten, es posible que el examen extraordinario se realice en parte o en su totalidad con ordenador.

SOBRE EL NO PRESENTADO EN LA CONVOCATORIA ORDINARIA

En los siguientes casos la calificación del alumno en la convocatoria ordinaria será *no presentado*:

1. No presentarse en el examen de capacitación.
2. No entregar prácticas suficientes para tener opciones a aprobar.
3. No satisfacer los criterios de asistencia a clase.

SOBRE EL CÁLCULO DE LA NOTA FINAL EN CONVOCATORIA ORDINARIA

En caso de aprobar tanto el examen de capacitación como la práctica, la calificación final será la media de ambos. En caso de que se suspenda alguno de los dos, la nota final será la mínima de las dos.

ORTOGRAFÍA

Cada falta de ortografía podrá penalizarse con 0,5 puntos.

7. PROGRAMA DE LA ASIGNATURA

7.1. PROGRAMA DE LA ASIGNATURA

PROGRAMA TEÓRICO

1. FUNDAMENTOS DE PROGRAMACIÓN ORIENTACIÓN A OBJETOS

- Las nociones de objeto, mensaje, clase, método, atributo, parámetro y variable.
- Encapsulación.
- Tecnología de soporte a la programación.
- Alcance de los atributos y métodos.
- Composición de objetos.
- La clase String.
- Estructuras de control.
- Arrays.
- Listas de objetos.
- Resolución de problemas matemáticos básicos con Java.
- Clases y métodos para el tratamiento del tiempo (fechas, horas, etc.).
- Métodos y atributos de clase.
- Herencia, clases abstractas, interfaces y polimorfismo.
- Excepciones.

2. ENTRADA-SALIDA DE DATOS

- Entrada por teclado.
- Lectura/escritura de ficheros.
- Acceso a páginas Web remotas.
- Generación de hojas de cálculo.

3. DOCUMENTACIÓN DE SOFTWARE

- Javadoc.
- UML.

PROGRAMA DE PRÁCTICAS (Véase la sección [6.3](#))

8. BIBLIOGRAFÍA DE LA ASIGNATURA

8.1. BIBLIOGRAFÍA BÁSICA

Sestoft P. *Java Precisely*. MIT Press

8.2. BIBLIOGRAFÍA COMPLEMENTARIA

- Eckel B. *Piensa en Java*. Prentice-Hall
- Deitel P., Deitel H. *Cómo programar en Java*. Pearson

8.3. RECURSOS WEB DE UTILIDAD

- MIT Open Course: Introduction to Computer Science and Programming <http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-00-introduction-to-computer-science-and-programming-fall-2008/>
- <https://netbeans.org/>
- <http://www.javahispano.org/>
- <http://www.oracle.com/technetwork/java/index.html> (especificación de Java)

9. NORMAS DE COMPORTAMIENTO

9.1. NORMAS

Las faltas en la Integridad Académica (ausencia de citación de fuentes, plagios de trabajos o uso indebido/prohibido de información durante los exámenes), así como firmar en la hoja de asistencia por un compañero que no está en clase, implicarán la pérdida de la evaluación continua, sin perjuicio de las acciones sancionadoras que estén establecidas (que pueden incluir la apertura de expediente).

10. MEDIDAS EXTRAORDINARIAS

En el caso de producirse alguna situación excepcional que impida la impartición de la docencia presencial en las condiciones adecuadas para ello, la Universidad adoptará las decisiones oportunas, y aplicará las medidas necesarias para garantizar la adquisición de las competencias y los resultados de aprendizaje de los estudiantes establecidos en esta Guía docente, según los mecanismos de coordinación docente del Sistema Interno de Garantía de calidad de cada título.