



UNIVERSIDAD PONTIFICIA DE SALAMANCA

Programación

Grado en ADMINISTRACIÓN Y DIRECCIÓN DE
EMPRESAS TECNOLÓGICAS – 2º curso

Modalidad Presencial

Sumario

Sumario	2
Datos básicos	3
Breve descripción de la asignatura	4
Requisitos previos	4
Objetivos	4
Competencias	4
Contenidos	5
Metodología	6
Criterios de evaluación	7
Recursos de aprendizaje y apoyo tutorial	8
Breve CV del profesor responsable	8
ADENDA GUÍA DOCENTE <i>ONLINE</i>	10
Metodología en el entorno <i>online</i>	10
Criterios de evaluación en el entorno <i>online</i>	10
Recursos de aprendizaje y apoyo tutorial <i>online</i>	11

Programación

Datos básicos

Módulo: Programación

Carácter: Obligatoria

Nº de créditos: 6 ECTS

Unidad Temporal: 2º Curso – 1º Semestre

Calendario: De septiembre a diciembre de 2020

Horario: Los horarios podrán consultarse en la sección *Horarios* de [la página web de la titulación](#)

Idioma en el que se imparte: Español

Profesores responsables de la asignatura: Berjón Gallinas, Roberto
Hernández de la Iglesia, Daniel

E-mail: Roberto Berjón Gallinas: rberjonga@upsa.es
Rubén Martín García: rmartinga@upsa.es

Horario de tutorías: El horario podrá consultarse en sección correspondiente a la asignatura dentro de la plataforma Moodle

Breve descripción de la asignatura

En esta asignatura se establecen las bases de la Programación Orientada a Objetos (OOP) En particular, el alumno aprenderá a programar en Java.

Requisitos previos

Haber cursado Fundamentos de Programación I

Objetivos

- Habilidad para definir la estructura modular y de datos para llevar a cabo las aplicaciones informáticas que cumplan con las especificaciones funcionales y restricciones del lenguaje de programación.
- Conocer y apreciar la importancia de un estilo de programación limpio, claro y ordenado.
- Capacidad para depurar, verificar y validar programas.
- Conocer un lenguaje de programación estructurada y saber aplicarlo en el desarrollo de software.
- Habilidad para utilizar lenguajes orientados a objetos para el desarrollo de sistemas software.

Competencias

Competencias Básicas y Generales	CG1 Aprendizaje autónomo 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 CB5 Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía
Transversales	CT1 Capacidad de análisis y síntesis CT2 Capacidad de organización y planificación CT5 Resolución de problemas CT6 Capacidad de toma de decisiones CT7 Trabajo en equipo en un entorno cambiante

Específicas	<p>CE39 Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.</p> <p>CE40 Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.</p> <p>CE41 Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.</p> <p>CE43 Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.</p>
-------------	---

Contenidos

CONTENIDOS DE LA ENSEÑANZA TEÓRICA

1. Conceptos básicos

Tipos básicos
 Variables y constantes
 Operadores
 Cadenas
 Entrada-Salida
 Flujo de control
 Arrays

2. Objetos y clases

Conceptos
 Clases propias
 Atributos finales
 Atributos, constantes y métodos estáticos
 Bloques de inicialización
 Paquetes

3. Herencia

Introducción
 Constructores y herencia
 Jerarquía de herencia
 Herencia, sustitución y refundición
 Sobreescritura de métodos
 Evitar la herencia
 Visibilidad
 La clase Object

Clases y métodos abstractos

4. Enumeraciones

5. Interfaces

Definición
 Implementación
 Herencia
 Métodos default
 Métodos static
 Comparadores: Comparable, Comparator

6. Java Collections Framework

Collection, Set, SortedSet, Queue, List
 Iteradores
 Map, SortedMap
 Collections

7. Excepciones

Introducción
 Definición
 Bloques try, catch, finally

CONTENIDOS DE LA ENSEÑANZA PRÁCTICA

En la parte práctica se trabajará sobre los mismos aspectos de la parte teórica en los que el alumno, realizando una serie de programas, comprobará que ha comprendido los conocimientos teóricos y es capaz de llevarlos a la práctica.

Metodología

METODOLOGÍA	HORAS	HORAS TRABAJO PRESENCIAL	DE	HORAS TRABAJO PRESENCIAL	DE NO
Sesiones Teóricas	30	80 (53%)			
Sesiones Prácticas	30				
Tutorías	10				
Exámenes	10				
Estudio y Trabajo Autónomo	30			70 (47%)	
Elaboración de Trabajos	35				
Bibliografía	5				
TOTAL	150	80		70	

Sesiones teóricas

Clases magistrales en aula de ordenadores en las que expondrán el temario con ayuda de medios audiovisuales y acompañados de ejemplos prácticos aclaratorios.

Sesiones prácticas

Por cada tema y después de haber asentado sus bases teóricas se aplican de forma práctica los conocimientos adquiridos. Se trata de sesiones centradas en la participación del alumno, en las que deberá realizar las tareas prácticas propuestas por cada tema, en el aula de ordenadores.

Criterios de evaluación

CONVOCATORIA ORDINARIA

Para aquellos alumnos que hayan realizado matrícula ordinaria en la asignatura (lo que les da derecho a docencia y dos convocatorias en enero y extraordinaria de junio/julio) la evaluación consta de una evaluación continua y un examen final. En la evaluación continua (que tiene una ponderación del 70% de la nota final) habrá dos exámenes parciales EP_1 y EP_2 (que englobarán, respectivamente, los temas 1-3 y 4-7) así como dos trabajos TR_1 y TR_2 (que podrán hacerse de forma individual o en grupo). El examen final englobará todo el contenido de la asignatura y tendrá una ponderación del 30% de la nota final.

La nota final se calculará a través de la siguiente fórmula:

$$notaFinal = (0,45 \times Parciales) + (0,25 \times Trabajos) + (0,3 \times ExamenFinal)$$

donde:

$$Parciales = \frac{EP_1 + EP_2}{2} \qquad Trabajos = \frac{TR_1 + TR_2}{2}$$

Para poder compensar entre sí, se exige una calificación mínima de 4 tanto en Parciales como en ExamenFinal. De no ser así, la nota final se obtendrá a través de la siguiente fórmula:

$$notaFinal = (0,45 \times \text{Mínimo}(Parciales, 5)) + (0,25 \times \text{Mínimo}(Trabajos, 5)) + (0,3 \times \text{Mínimo}(ExamenFinal, 5))$$

Los exámenes parciales EP_1 y EP_2 podrán recuperarse el mismo día en el que se realiza el examen final. Las calificaciones aprobadas (*calificación* ≥ 5) se guardarán hasta la convocatoria de junio/julio del presente curso académico.

CONVOCATORIA EXTRAORDINARIA

Para aquellos alumnos que hayan realizado matrícula extraordinaria en la asignatura (lo que únicamente da derecho al examen correspondiente a la convocatoria para la que realizaron la matrícula) la evaluación consistirá en un examen práctico final que englobará todos los contenidos de la misma.

Recursos de aprendizaje y apoyo tutorial

BIBLIOGRAFÍA



Core Java. Vol. I, Fundamentals / Cay S. Horstmann, Gary Cornell. . --8th ed. . -- Upper Saddle River ; Madrid [etc.] : Sun Microsystems Press, cop. 2008. ISBN: 978-0-13-235476-9

<http://koha.upsa.es/cgi-bin/koha/opac-detail.pl?biblionumber=200360>



Core Java 2. Vol. II, Advanced features / Cay S. Horstmann, Gary Cornell.. --8th ed. . -- Upper Saddle River [etc.] : Sun Microsystems, cop. 2008. ISBN: 978-0-13-235479-0; 0-13-235479-9

<http://koha.upsa.es/cgi-bin/koha/opac-detail.pl?biblionumber=186135>



Java generics and collections / Maurice Naftalin and Philip Wadler. . --1st ed. . -- Beijing [etc.] : O'Reilly, 2006

<http://koha.upsa.es/cgi-bin/koha/opac-detail.pl?biblionumber=161985>

OTROS RECURSOS

Apache NetBeans

- <https://netbeans.apache.org/>

Oracle Java Tutorials.

- Learning the Java language: <https://docs.oracle.com/javase/tutorial/java/index.html>
- Collections: <https://docs.oracle.com/javase/tutorial/collections/index.html>
- Date-Time API: <https://docs.oracle.com/javase/tutorial/datetime/index.html>
- Java SE 8 Programmer I Exam: <https://docs.oracle.com/javase/tutorial/extra/certification/javase-8-programmer1.html>
- Java SE 8 Programmer II Exam: <https://docs.oracle.com/javase/tutorial/extra/certification/javase-8-programmer2.html>
- Java SE 8 Upgrade Exam: <https://docs.oracle.com/javase/tutorial/extra/certification/javase-8-upgrade.html>

PLATAFORMA MOODLE

En el campus virtual el alumno encontrará los recursos básicos de la asignatura organizados por temas. Principalmente incluyen las presentaciones de cada tema, información complementaria, así como los enunciados de los ejercicios planteados y de las prácticas a realizar.

TUTORÍAS

Existe un horario de atención tutorial de 4 horas a la semana distribuidas en horario de mañana y tarde para facilitar la asistencia de los alumnos que necesiten este apoyo tutorial.

Igualmente, el apoyo tutorial es permanente a través de los medios electrónicos como la plataforma Moodle y el correo electrónico.

Breve CV del profesor responsable

Roberto Berjón Gallinas es licenciado en Informática y Doctor por la Universidad de Deusto. Actualmente es profesor Encargado de Cátedra en la Facultad de Informática de la Universidad Pontificia de Salamanca. Tiene un Sexenio de Investigación vivo reconocido por la CNEAI y posee

las acreditaciones de profesor de Universidad Privada y profesor Contratado Doctor por la ACSUCYL. Es autor de numerosas publicaciones científicas indexadas en los principales rankings de referencia (JCR y SCOPUS), ha dirigido y participado en numerosos proyectos de investigación contando con varios registros de propiedad intelectual derivados de los mismos en el ámbito de las aplicaciones móviles y los sistemas de información.

Daniel Hernández de la Iglesia es Doctor en Ingeniería Informática, Ingeniero técnico en Informática de Sistemas y Grado en Ingeniería Informática por la Universidad de Salamanca. Cuenta con un máster oficial en Sistemas Inteligentes y en los últimos años ha estado ligado a diferentes grupos de investigación del ámbito de la inteligencia artificial donde ha participado en decenas de proyectos de investigación nacionales e internacionales. Es autor de diferentes capítulos de libros y ha presentado más de veinte trabajos de investigación en distintos congresos internacionales. Además, cuenta con numerosas publicaciones científicas en revistas de impacto internacionales indexadas en el ranking de referencia JCR. Ha sido galardonado con el primer premio del Concurso de Datos Abiertos organizado por la Junta de Castilla y León en el año 2013, y con el primer premio de innovación al mejor proyecto de investigación entregado por la junta de Castilla y León en el año 2016.

ADENDA GUÍA DOCENTE *ONLINE*

Vigente durante el curso 2020-21 en caso de confinamiento generado por la crisis del COVID-19

Programación

Metodología en el entorno *online*

METODOLOGÍA (ACTIVIDADES FORMATIVAS)		TOTAL HORAS	HORAS ACTIVIDADES SÍNCRONAS DEL ALUMNO	HORAS ACTIVIDADES ASÍNCRONAS DEL ALUMNO
CLASES SÍNCRONAS	Clases teóricas síncronas	30	70 (47%)	
	Clases prácticas síncronas	30		
PRUEBAS SÍNCRONAS	Pruebas escritas síncronas	10		
RETOS ACADÉMICOS	Enunciado, entrega y retroalimentación de tareas	80		80 (53%)
TOTAL		150	70	80

Criterios de evaluación en el entorno *online*

CONVOCATORIA ORDINARIA

Para aquellos alumnos que hayan realizado matrícula ordinaria en la asignatura (lo que les da derecho a docencia y dos convocatorias en enero y extraordinaria de junio/julio) la evaluación consta de una evaluación continua y un examen final. En la evaluación continua (que tiene una ponderación del 70% de la nota final) habrá dos exámenes parciales EP₁ y EP₂ (que englobarán, respectivamente, los temas 1-3 y 4-7) así como dos trabajos TR₁ y TR₂ (que podrán hacerse de forma individual o en grupo). El examen final englobará todo el contenido de la asignatura y tendrá una ponderación del 30% de la nota final.

La nota final se calculará a través de la siguiente fórmula:

$$\text{notaFinal} = (0,45 \times \text{Parciales}) + (0,25 \times \text{Trabajos}) + (0,3 \times \text{ExamenFinal})$$

donde:

$$\text{Parciales} = \frac{EP_1 + EP_2}{2} \qquad \text{Trabajos} = \frac{TR_1 + TR_2}{2}$$

Para poder compensar entre sí, se exige una calificación mínima de 4 tanto en Parciales como en ExamenFinal. De no ser así, la nota final se obtendrá a través de la siguiente fórmula:

$$\text{notaFinal} = (0,45 \times \text{Mínimo}(\text{Parciales}, 5)) + (0,25 \times \text{Mínimo}(\text{Trabajos}, 5)) + (0,3 \times \text{Mínimo}(\text{ExamenFinal}, 5))$$

Los exámenes parciales EP₁ y EP₂ podrán recuperarse el mismo día en el que se realiza el examen final. Las calificaciones aprobadas (*calificación* ≥ 5) se guardarán hasta la convocatoria de junio/julio del presente curso académico.

EVALUACIÓN		PUNTOS (sobre 10)
PRUEBAS Y PRESENTACIONES SÍNCRONAS	Pruebas escritas síncronas	7,5
RETOS ACADÉMICOS	Enunciado, entrega y retroalimentación de tareas	2,5
TOTAL		10

• PRUEBAS Y RETOS ACADÉMICOS DE EVALUACIÓN *ONLINE*

- Trabajo 1 (TR₁): el alumno realizará una aplicación en la que ponga de manifiesto los conocimientos adquiridos respecto a los temas 1-3 de la asignatura.
- Trabajo 2 (TR₂): el alumno realizará una aplicación que haga uso de los contenidos de la segunda parte de la asignatura.

CONVOCATORIA EXTRAORDINARIA

Para aquellos alumnos que hayan realizado matrícula extraordinaria en la asignatura (lo que únicamente da derecho al examen correspondiente a la convocatoria para la que realizaron la matrícula) la evaluación consistirá en un examen práctico final que englobará todos los contenidos de la misma.

Recursos de aprendizaje y apoyo tutorial *online*

RECURSOS

Apache NetBeans

- <https://netbeans.apache.org/>

Oracle Java Tutorials.

- Learning the Java language: <https://docs.oracle.com/javase/tutorial/java/index.html>
- Collections: <https://docs.oracle.com/javase/tutorial/collections/index.html>
- Date-Time API: <https://docs.oracle.com/javase/tutorial/datetime/index.html>

- Java SE 8 Programmer I Exam: <https://docs.oracle.com/javase/tutorial/extra/certification/javase-8-programmer1.html>
- Java SE 8 Programmer II Exam: <https://docs.oracle.com/javase/tutorial/extra/certification/javase-8-programmer2.html>
- Java SE 8 Upgrade Exam: <https://docs.oracle.com/javase/tutorial/extra/certification/javase-8-upgrade.html>

PLATAFORMA MOODLE

En el campus virtual el alumno encontrará los recursos básicos de la asignatura organizados por temas. Principalmente incluyen demostraciones de cada tema, información complementaria, así como los enunciados de los ejercicios planteados y de las prácticas a realizar.

TUTORÍAS

Existe un horario de atención tutorial de 4 horas a la semana para facilitar la asistencia de los alumnos que necesiten este apoyo tutorial. Será preciso establecer cita previa a través email. Igualmente, el apoyo tutorial es permanente a través de los medios electrónicos el correo electrónico.