



Universidad
de Alcalá

GUÍA DOCENTE

PARADIGMAS AVANZADOS DE PROGRAMACIÓN

Grado en Ingeniería Informática

Curso Académico 2019/2020

Curso 3º – Cuatrimestre 2º

GUÍA DOCENTE

Nombre de la asignatura:	Paradigmas Avanzados de Programación
Código:	781004
Titulación en la que se imparte:	Grado en Ingeniería Informática
Departamento:	Departamento Ciencias de la Computación
Carácter:	Optativa de tecnología específica ⁽¹⁾
Créditos ECTS:	6
Curso y cuatrimestre:	Tercer Curso / Segundo Cuatrimestre
Profesorado:	Consultar en la página web del departamento
Horario de Tutoría:	El horario de Tutorías se indicará el primer día de clase
Idioma en el que se imparte:	Español

1a. PRESENTACIÓN

La asignatura de Paradigmas Avanzados de Programación desarrolla contenidos de programación orientados a alcanzar un perfeccionamiento en las técnicas de programación conocidas por los alumnos mediante la utilización de las tecnologías más novedosas.

La asignatura promueve la comprensión de los conceptos básicos, busca la capacitación para el análisis de problemas donde se requieran un alto grado de cálculo computacional.

Se trata de una asignatura con un grado de aplicabilidad práctica importante, lo que permite trabajar con ejemplos reales sobre aplicaciones diversas.

Los principales temas a abordar en esta asignatura son: programación paralela, programación funcional, programación en la nube.

Para un buen aprovechamiento de la asignatura, se requieren conocimientos y competencias de las materias programación impartidas en el primer y segundo curso del grado.

¹ Las asignaturas optativas de tecnología específica dan lugar a la mención y en este Grado son obligatorias de cursar.

1b. PRESENTACIÓN

The subject Advanced Programming Expansion develops content aimed at achieving and improvement in programming techniques known by students using the last technologies.

The course promotes understanding of the basics, the training seeks to analyze problems where a high degree of computational calculation is required.

This is a subject with a significant degree of practical applicability, which allows working with real examples of various applications.

The main topics to be addressed in this course are: parallel programming, functional programming, programming in the cloud.

For a good use of the subject, knowledge and skills of programming subjects taught in the first and second degree course they are required.

2. COMPETENCIAS

Competencias generales:

CG4 Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5 de la resolución BOE-A-2009-12977.

CG8 Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

Competencias específicas:

CC1 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.

CC3 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.

Resultados de Aprendizaje:

- RA1 Evaluar, justificar y resolver cómo afectan las capacidades y limitaciones de los lenguajes de programación más comunes a la creación de sistemas informáticos.
- RA2 Identificar, aplicar y evaluar críticamente la evolución de los lenguajes de programación, los diferentes paradigmas disponibles hoy día y sus principales características a las decisiones de diseño de sistemas de información.
- RA3 Comparar y contrastar entornos de ejecución interpretados y compilados, con sus ventajas y desventajas, así como la importancia de la abstracción en el contexto de las máquinas virtuales y comprender distintas realizaciones prácticas de dicho concepto.
- RA4 Identificar y aplicar la programación distribuida, concurrente y paralela, conociendo sus algoritmos fundamentales y las ventajas e inconvenientes de cada paradigma, para diseñar sistemas más eficaces o eficientes desde el punto de vista de rendimiento, usuario u otros
- RA5 Identificar, aplicar y evaluar críticamente los fundamentos de la programación funcional identificando las ventajas e inconvenientes de cada paradigma.

3. CONTENIDOS

Bloques de contenido (se pueden especificar los temas si se considera necesario)	Total de clases, créditos u horas (*)
Programación Paralela Conceptos básicos de la programación paralela. Modelos de sistemas paralelos, metodología y programación. Programación de hardware gráfico paralelo. Consideraciones y evaluación del rendimiento.	25 horas
Programación Funcional Características de este tipo de programación. Influencia sobre los lenguajes actuales. Lenguajes funcionales modernos y tendencias.	20 horas
Programación en la nube (Cloud Computing) Nuevos paradigmas. Software como servicio, Plataforma como servicio e Infraestructura como Servicio. Tipos de nubes. Desarrollo e implementación de aplicaciones en la nube.	15 horas

(*) Incluyen PEI y PEF

4. METODOLOGÍAS DE ENSEÑANZA-APRENDIZAJE.- ACTIVIDADES FORMATIVAS

4.1. Distribución de créditos (especificar en horas)

Número de horas presenciales:	56 horas + 4 horas de examen de evaluación
Número de horas del trabajo propio del estudiante:	90 horas
Total horas	150 horas

4.2. Estrategias metodológicas, materiales y recursos didácticos

En el proceso de enseñanza-aprendizaje de los contenidos anteriormente reseñados se emplearán las siguientes actividades formativas:

- Clases Teóricas presenciales.
- Clases Prácticas: resolución de problemas presenciales.
- Prácticas en Laboratorio presenciales.
- Tutorías: individuales y/o grupales.

Además, en función de la naturaleza de las distintas partes de la materia objeto de estudio, se podrán utilizar, entre otras, las siguientes actividades formativas:

- Elaboración de trabajos con responsabilidad individual pero con gestión de la información como equipo.
- Puesta en común de la información, problemas y dudas que aparezcan en la realización de los trabajos.
- Organización y realización de jornadas públicas con presentaciones orales y discusión de resultados.
- Utilización de Plataforma de Aula Virtual.

Actividades presenciales:

1. En el aula: exposición y discusión de los conocimientos básicos de la asignatura y planteamiento y resolución teórica de ejercicios y supuestos relacionados (orientados a la enseñanza de las competencias específicas de la asignatura).
2. En el laboratorio: planteamiento y desarrollo de ejercicios prácticos que permitan solventar problemas y analizar hipótesis y contribuyan al desarrollo de la capacidad de análisis de resultados, razonamiento crítico y comprensión de los métodos de resolución planteados.

Actividades no presenciales:

1. Análisis y asimilación de los contenidos de la materia, resolución de problemas, consulta bibliográfica, preparación de trabajos individuales y grupales, realización de exámenes presenciales y autoevaluaciones. Orientadas especialmente al desarrollo de métodos para la autoorganización y planificación del trabajo individual y en equipo.
2. Tutorías: asesoramiento individual y en grupos durante el proceso de enseñanza-aprendizaje, bien en forma presencial o a distancia.

Durante todo el proceso de aprendizaje en la asignatura, el alumno deberá hacer uso de distintas fuentes y recursos bibliográficos o electrónicos, de manera que se familiarice con los entornos de documentación que en un futuro utilizará profesionalmente

5. EVALUACIÓN: Procedimientos, criterios de evaluación y de calificación

Preferentemente se ofrecerá a los alumnos un sistema de evaluación continua que tenga características de evaluación formativa, de manera que sirva de realimentación en el proceso de enseñanza-aprendizaje por parte del alumno. Para ello se establecen los siguientes

Procedimientos de Evaluación

1. **Convocatoria Ordinaria** : La evaluación en la convocatoria ordinaria debe estar inspirada en los criterios de evaluación continua (Normativa de Regulación de los Procesos de Enseñanza Aprendizaje, NRPEA, art 3), atendiendo siempre a la adquisición de las competencias especificadas en la asignatura
 - a. *Evaluación Continua*: Consistente en la realización y superación de las prácticas de laboratorio, trabajos de la asignatura y exámenes intermedios y final. La superación de las prácticas y del trabajo de la asignatura se realizará a lo largo del cuatrimestre
 - b. *Evaluación Final*: Consistirá en la realización y superación de las prácticas de laboratorio de propuestas para esta convocatoria y un examen final.
2. **Convocatoria Extraordinaria**: en la convocatoria extraordinaria, los alumnos que no hayan superado la convocatoria ordinaria realizarán una única prueba que incluirá pruebas teóricas, problemas, entrega y defensa de las prácticas y/o trabajos propuestos para esta convocatoria. Los alumnos deberán ponerse en contacto con los profesores de la asignatura para que éstos les informen de las prácticas y/o trabajos que deberán presentar. Esta prueba constituirá el 100% de la nota de la asignatura. Se plantean dos situaciones:

- En caso de que el estudiante haya realizado el trabajo de la asignatura, si así lo decide, se le asignará la calificación obtenida en la convocatoria ordinaria. El resto de la evaluación se basará en un examen final.

Criterios de Evaluación

Los Criterios de Evaluación deben atender al grado de adquisición de las competencias por parte del estudiante. Para ello se definen los siguientes criterios:

CE1: El alumno muestra capacidad de aplicar conocimientos y metodologías de programación paralela y evaluar el rendimiento de este paradigma frente otros.

CE2: El alumno muestra que puede realizar, desarrollar e implementar soluciones para la programación de hardware gráfico paralelo.

CE3: El alumno muestra capacidad e iniciativa a la hora de resolver problemas prácticos asociados al desarrollo de la programación paralela.

CE4: El alumno conoce y puede aplicar los fundamentos y las características de la programación funcional.

CE5: El alumno muestra capacidad e iniciativa a la hora de resolver problemas prácticos asociados al desarrollo, mediante la programación funcional.

CE6: El alumno muestra capacidad de aplicar conocimientos y metodologías de programación en la nube.

CE7: El alumno ha adquirido conocimientos técnicos para realizar un adecuado desarrollo e implementación de una aplicación en la nube (Cloud Computing).

Instrumentos de Calificación.

Esta sección indica los instrumentos de evaluación que serán aplicados a cada uno de los criterios de Evaluación.

1. Pruebas de Evaluación Intermedia (PEI 1): Consistente en la resolución de problemas prácticos y teóricos de los Conocimientos aplicados a la programación paralela.
2. Pruebas de Evaluación Intermedia (PEI 2): Consistente en la resolución de problemas prácticos y teóricos de programación funcional, programación en la nube.
3. Pruebas de Laboratorio PL: Las pruebas de laboratorio de la asignatura consisten en el diseño e implementación de una aplicación práctica para cada uno de los paradigmas de la asignatura, (PL1, PL2, PL3), en cada una de ellas se presentarán informes periódicos así como un informe final y la defensa de la práctica.

4. Trabajos de la asignatura TA: Los trabajos de la asignatura consiste en el análisis de una aplicación práctica de cada uno de los tres paradigmas de la asignatura (TA1 y TA2).
5. Prueba de Evaluación Final (PEF) consistente en la resolución de problemas de prácticos y teóricos de organización de empresas y gestión de proyectos.

Criterios de Calificación

Esta sección cuantifica los criterios de evaluación para la superación de la asignatura.

Convocatoria Ordinaria: Evaluación Continua

En la convocatoria ordinaria – evaluación continua la relación entre los criterios, instrumentos y calificación es la siguiente.

Competencia		Resultado Aprendizaje	Criterio de Evaluación	Instrumento de Evaluación	Peso en la calificación
CG4, CG8	CC1 - CC3	RA1, RA2, RA4	CE1,CE2,CE3	PEI 1	13%
				PL1	18%
				TAL1	2%
CG4, CG8	CC1 - CC3	RA1,RA2, RA3, RA5	CE4, CE5, CE6, CE7	PEI 2	13%
				PL2	17%
				TA2	2%
				PL3	15%
CG4, CG8	CC1 - CC3	RA1,RA2,RA3, RA4,RA5	CE1,CE2,CE3, CE4,CE5,CE6 CE7	PEF	20%

Como criterio general, aquellos alumnos en convocatoria ordinaria que no se presenten a la evaluación de todos los instrumentos de Evaluación se considerarán No Presentados.

Convocatoria Ordinaria: Evaluación Final

Competencia		Resultado Aprendizaje	Criterio de Evaluación	Instrumento de Evaluación	Peso en la calificación
CG4, CG8	CC1 - CC3	RA1-RA5	CE1-CE7	PEF	55%
	CC1 - CC3	RA1-RA5	CE1-CE7	PL1, PL2, PL3	45%

Convocatoria Extraordinaria

Competencia		Resultado Aprendizaje	Criterio de Evaluación	Instrumento de Evaluación	Peso en la calificación
CG4, CG8	CC1 - CC3	RA1-RA5	CE1-CE7	PEF	55%
	CC1 - CC3	RA1-RA5	CE1-CE7	PL1, PL2, PL3	45%

6.BIBLIOGRAFÍA

Bibliografía Básica

- Apuntes de la asignatura, disponibles a través de la plataforma virtual.

Bibliografía Complementaria

Programación Paralela

- Kirk & Hwu, "Programming massively Parallel Processors", Elsevier, 2010
- Sanders & Kandrot, "CUDA by Example", Addison-Wesley, 2010.
- Mattson et al., "Patterns for Parallel Programming", Addison Wesley, 2005
- NVIDIA, "NVidia CUDA Programming Guide", NVidia (Libro de Referencia), 2011

Programación Funcional

- Odersky, Spoon & Venners, "Programming in Scala – A Comprehensive Step-by-step Guide", Artima 2008
- Wampler & Payne, "Programming Scala", O'Reilly, 2009
- Subramaniam, "Programming Scala. Tackle Multi-Core Complexity on the Java Virtual Machine", Pragmatic Bookshelf, 2009
- Lindholm & Yellin, "The Java™ Virtual Machine

Computación en la nube

- Roche & Douglas, "Beginning Java Google App Engine", Apress, 2009
- Dewsbury, "Google Web Toolkit Applications", Prentice-Hall, 2008
- Chen-Becker, Danciu & Weir, "The definite Guide to Lift", Apress, 2009
- Chandrasekaran, "Essentials of Cloud Computing", Chapman & Hall/CRC, 2015
- Roche & Douglas, "Beginning Java Google App Engine", Apress, 2009.
- Dewsbury "Google Web Toolkit Applications", Prentice-Hall, 2008.
- Chen-Becker, Danciu, & Weir, "The definite Guide to Lift A Scala-based Web Framework", Apress, 2009
- Tulloch "Introducing Windows Azure for IT professionals", Microsoft Press, 2013

- Collier & Shahan, “Microsoft Azure Essentials: Fundamentals of Azure, Second Edition”, Microsoft Press, 2016
- Amazon Web Services, Getting Started with AWS. 2016
- AWS Whitepapers Architecting for the AWS Cloud: Best Practices, 2018
- Geewax, “Google Cloud Platform in Action” Manning Publications, 2018